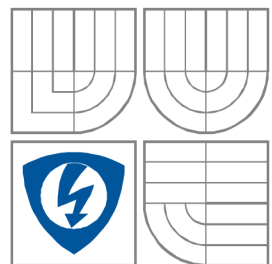




VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ
ÚSTAV BIOMEDICÍNSKÉHO INŽENÝRSTVÍ

FACULTY OF ELECTRICAL ENGINEERING AND
COMMUNICATION
DEPARTMENT OF BIOMEDICAL ENGINEERING

STATISTICKÁ DATA V NEMOCNIČNÍM INFORMAČNÍM SYSTÉMU

STATISTIC DATA IN THE HOSPITAL INFORMATION SYSTEM

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

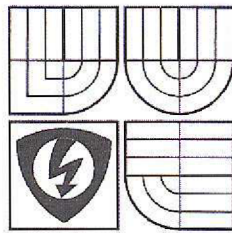
AUTOR PRÁCE
AUTHOR

TOMÁŠ VELIČKA

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. PETR FEDRA

BRNO, 2009



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav biomedicínského inženýrství

Diplomová práce

magisterský navazující studijní obor
Biomedicínské a ekologické inženýrství

Student: Velička Tomáš, Bc.

Ročník: 2

ID: 84715

Akademický rok: 2008/09

NÁZEV TÉMATU:

Statistická data v nemocničním informačním systému

POKYNY PRO VYPRACOVÁNÍ:

Prostudujte platnou legislativu České republiky o národních zdravotních registrech a povinných hlášeních do těchto registrů. Seznamte se s nemocničním informačním systémem Clinicom a možností grafického prezentace uložených dat pomocí CSP (Caché Server Pages) ve webovém prohlížeči.

Práce musí obsahovat:

- návrh a realizaci výpočtu důležitých statistických ukazatelů z tabulek pomocí SQL dotazů,
- návrh a realizaci uživatelského webového rozhraní s vypočtenými ukazateli, například: průměrná doba hospitalizace, počet nových onkologických diagnóz, spotřeba léků a další,
- analýzu zabezpečení přístupu k těmto informacím v síti Internet,
- zhodnocení funkce celého systému.

DOPORUČENÁ LITERATURA:

- [1] KURSTEN, Wolfgang, et al. Caché: Databáze postrelačního typu a tvorba aplikací. Brno: CP Books, 2005. ISBN 80-251-0491-5.
[2] KASAL, Pavel. Internet a medicína. Praha: Grada, 2001. ISBN 80-247-0119-7.

Termín zadání: 30.10.2008

Termín odevzdání: 29.5.2009

Vedoucí práce: Ing. Petr Fedra

Konzultanti diplomové práce:

prof. Ing. Jiří Jan, CSc.

předseda oborové rady



UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

LICENČNÍ SMLOUVA POSKYTOVANÁ K VÝKONU PRÁVA UŽÍT ŠKOLNÍ DÍLO

uzavřená mezi smluvními stranami:

1. Pan/paní

Jméno a příjmení: Tomáš Velička
Bytem: Ibsenova 3, Brno, 638 00
Narozen/a (datum a místo): 24. července 1985 v Brně

(dále jen „autor“)

a

2. Vysoké učení technické v Brně

Fakulta elektrotechniky a komunikačních technologií
se sídlem Údolní 53, Brno, 602 00
jejímž jménem jedná na základě písemného pověření děkanem fakulty:
prof. Ing. Jiří Jan, CSc, předseda rady oboru Biomedicínské a ekologické
inženýrství
(dále jen „nabyvatel“)

Čl. 1

Specifikace školního díla

1. Předmětem této smlouvy je vysokoškolská kvalifikační práce (VŠKP):

- disertační práce
 - diplomová práce
 - bakalářská práce
 - jiná práce, jejíž druh je specifikován jako
- (dále jen VŠKP nebo dílo)

Název VŠKP: Statistická data v nemocničním informačním systému

Vedoucí/ školitel VŠKP: Ing. Petr Fedra

Ústav: Ústav biomedicínského inženýrství

Datum obhajoby VŠKP: _____

VŠKP odevzdal autor nabyvateli*:

- v tištěné formě – počet exemplářů: 2
- v elektronické formě – počet exemplářů: 2

2. Autor prohlašuje, že vytvořil samostatnou vlastní tvůrčí činností dílo shora popsané a specifikované. Autor dále prohlašuje, že při zpracovávání díla se sám nedostal do rozporu s autorským zákonem a předpisy souvisejícími a že je dílo dílem původním.
3. Dílo je chráněno jako dílo dle autorského zákona v platném znění.
4. Autor potvrzuje, že listinná a elektronická verze díla je identická.

* hodící se zaškrtněte

Článek 2

Udělení licenčního oprávnění

1. Autor touto smlouvou poskytuje nabyvateli oprávnění (licenci) k výkonu práva uvedené dílo nevýdělečně užit, archivovat a zpřístupnit ke studijním, výukovým a výzkumným účelům včetně pořizování výpisů, opisů a rozmnoženin.
2. Licence je poskytována celosvětově, pro celou dobu trvání autorských a majetkových práv k dílu.
3. Autor souhlasí se zveřejněním díla v databázi přístupné v mezinárodní síti
 - ihned po uzavření této smlouvy
 - 1 rok po uzavření této smlouvy
 - 3 roky po uzavření této smlouvy
 - 5 let po uzavření této smlouvy
 - 10 let po uzavření této smlouvy(z důvodu utajení v něm obsažených informací)
4. Nevýdělečné zveřejňování díla nabyvatelem v souladu s ustanovením § 47b zákona č. 111/ 1998 Sb., v platném znění, nevyžaduje licenci a nabyvatel je k němu povinen a oprávněn ze zákona.

Článek 3

Závěrečná ustanovení

1. Smlouva je sepsána ve třech vyhotoveních s platností originálu, přičemž po jednom vyhotovení obdrží autor a nabyvatel, další vyhotovení je vloženo do VŠKP.
2. Vztahy mezi smluvními stranami vzniklé a neupravené touto smlouvou se řídí autorským zákonem, občanským zákoníkem, vysokoškolským zákonem, zákonem o archivnictví, v platném znění a popř. dalšími právními předpisy.
3. Licenční smlouva byla uzavřena na základě svobodné a pravé vůle smluvních stran, s plným porozuměním jejímu textu i důsledkům, nikoliv v tísní a za nápadně nevýhodných podmínek.
4. Licenční smlouva nabývá platnosti a účinnosti dnem jejího podpisu oběma smluvními stranami.

V Brně dne: 29. května 2009

.....
Nabyvatel

.....
Autor

ABSTRAKT

Abstrakt práce v češtině

Cílem práce byl výpočet některých statistických ukazatelů z dat získaných z Nemocničního Informačního Systému Clinicom. Data jsou v tomto informačním systému spravována databázovým systémem Caché. Pro práci byla využita data z databáze spravované VUT v Brně, ústavem Biomedicínkého inženýrství. Data byla získávána z databáze pomocí SQL dotazů. Takto získané statistické údaje bylo třeba prezentovat na internetu. Pro tuto prezentaci bylo navrženo a realizováno uživatelské rozhraní v programovacím jazyce CSP.

Výsledné statistické ukazatele byly rozděleny do 3 základních skupin: Hospitalizace, Povinná hlášení a Spotřeba materiálu. Kromě statistických údajů jsou v uživatelském rozhraní zobrazena i některá patientská data a grafy výsledků. Součástí práce je i návrh online formuláře pro hlášení nálezu novotvaru.

KLÍČOVÁ SLOVA

Klíčová slova v češtině

NIS Clinicom, SQL, Caché databáze, CSP, Národní zdravotní registry

ABSTRACT

Abstract in English

The aim of this work was the evaluation of some statistic indicators from the data obtained from the Health Informatic System Clinicom. The data used in this informatic system are administrated by the database system Caché. The data from the database which is administrated by VUT in Brno, departement of Biomedical engineering, have been also used. Data mentioned in this work were mined from the database by SQL queries. The statistic indicators obtained this way were needed to be presented on internet. The user interface for this presentation was designed and realized in programming language CSP.

The resulting statistic indicators were split up into 3 basic groups: Hospitalization, Mandatory reporting and Usage of material. Besides the statistic indicators some patients' data and the graphs of results are shown in the user interface. The part which has also been used in this work is the concept of the online form for reporting of a cryptologic diagnose.

KEYWORDS

Keywords in English

HIS Clinicom, SQL, Caché database, CSP, The National registers of health

VELIČKA, T. *Statistická data v nemocničním informačním systému*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 93 s.
Vedoucí diplomové práce Ing. Petr Fedra.

Prohlášení

Prohlašuji, že svou diplomovou práci na téma Statistická data v nemocničním informačním systému jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne 29. května 2009

.....
podpis autora

Poděkování

Děkuji vedoucímu diplomové práce Ing. Petru Fedrovi za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé diplomové práce.

V Brně dne 29. května 2009

.....
podpis autora

Obsah

Obsah.....	1
Úvod.....	10
1. Nemocniční informační systémy (NIS).....	11
1.1. Základní funkce NIS	11
1.2. Medicínská informatika.....	12
1.3. Pacientské zdravotní záznamy	13
1.4. Nemocniční informační systém Clinicom.....	13
1.4.1. Uživatelská rozhraní systému Clinicom.....	15
2. Mezinárodní klasifikace nemocí (MKN)	17
2.1. MKN-10	17
2.2. Třímístné položky	19
2.3. Čtyřmístné položky	19
2.4. Dva kódy pro stejný stav (Systém "křížků a hvězdiček")	19
3. Národní zdravotní registry	21
3.1. Národní onkologický registr.....	21
3.2. Hlášení zhoubného novotvaru	23
4. Zabezpečení dat na internetu.....	24
4.1. Firewall.....	24
4.2. Hesla.....	25
4.3. Kryptografie	26
4.3.1. Symetrické šifrování	26
4.3.2. Asymetrické šifrování	26
4.4. SSL.....	27
4.5. Zabezpečení využité v diplomové práci.....	28
5. Databázové systémy v NIS Clinicom.....	31
5.1. Caché.....	31
5.1.1. Caché Cube	31
5.1.2. Caché studio	32
5.1.3. SQL Manager	34
5.2. SQL	35
5.2.1. Využití SQL v diplomové práci	36
5.3. Objektové databáze	38
5.4. ODBC.....	38
5.4.1. Instalace softwaru.....	39
5.4.2. Načtení dat pomocí MS Excel.....	40
6. CSP.....	41
6.1. HTML.....	42
6.2. Větvení kódu (podmínky)	42
6.3. Cykly	43
6.4. Skripty	43
6.5. Databázové dotazy	43
6.6. Práce s proměnnými a daty	43
7. Popis vytvořené webové aplikace	45
7.1. Hospitalizace - Průměrná doba.....	46
7.2. Hospitalizace – Výpočet mortality.....	48
7.3. Povinná hlášení – Výskyt novotvarů.....	49
7.4. Povinná hlášení – Hlášení novotvarů	51

7.5. Povinná hlášení – Výskyt TBC	53
7.6. Spotřeba materiálu – Spotřeba léků	53
7.7. Spotřeba materiálu – Ostatní materiál	54
Závěr.....	55
Seznam použitých zdrojů	56
Abecední seznam použitých zkratk	58
Příloha č.1 - Kompletní zdrojový kód webové aplikace	59
Příloha č.2 - Protokol „Incidence a léčba zhoubného novotvaru“	92
Příloha č.3 - Protokol „Incidence zhoubného novotvaru“	93

Úvod

V dnešní době se výpočetní technika rozšířila již v podstatě do všech odvětví lidské společnosti. Počítače jsou užívány i ve všech státních resortech. V bezpečnostních složkách hrají velmi významnou úlohu. Lze s jejich pomocí monitorovat a lokalizovat různé objekty, uchovávat a snadno vyhledávat jakákoli data, atd. I ve zdravotnictví hrají počítače a hlavně jejich specializované zdravotnické informační systémy čím dál větší úlohu. Bez jejich pomoci by se například již velmi těžko zpracovávalo a archivovalo velké množství dat například ze zobrazovacích a monitorovacích systémů. Zdravotnická dokumentace se uchovává minimálně po dobu 5 let. Konkrétní délky dob archivace různých druhů dat (písemná, obrazová,..) jsou uvedeny ve vyhlášce č. 385/2006 Sb. Vzhledem k neustále se zvětšujícím datovým objemům z jednotlivých vyšetření by nároky na skladovací prostory těchto dat v papírové podobě byly těžko realizovatelné. Navíc by se v takovém množství dat těžko a zdlouhavě vyhledávalo.

Z těchto a dalších důvodů jsou hlavně ve větších zdravotnických zařízeních zaváděny takzvané Nemocniční informační systémy (NIS). České ministerstvo zdravotnictví reagovalo na možnost integrace počítačové techniky do zdravotnictví zavedením Národního zdravotnického informačního systému (NZIS). NZIS má za úkol dohlížet na využívání specializovaných informačních systémů v českém zdravotnictví. Spravuje například Národní zdravotní registry.

Na trhu už existuje mnoho komerčně dostupných informačních systémů vhodných pro správu a vedení zdravotnických zařízení (například nemocnic). Výrobci je celá řada, například firma SMS se svým systémem NIS Clinicom, na něž je tato diplomová práce zaměřena. Data nemocničních informačních systémů bývají většinou uložena v databázi. NIS Clinicom využívá databázi Caché od americké firmy InterSystems, která je v této práci využita.

Data, která můžeme z těchto specializovaných informačních systémů získat, se však netýkají jen správy pacientů, ale poskytují i výstupní informace důležité pro ekonomický chod zařízení. Právě získáváním takovýchto dat, jako je například průměrná délka hospitalizace, úmrtnost a výskyt určitých onemocnění, se zabývá tato práce.

Zjednodušení přenosu takto získaných dat poskytuje internet. Prezentací dat na internetu umožňujeme autorizovaným uživatelům přístup k těmto datům z celého světa. Na druhou stranu však zvyšujeme riziko zneužití, či krádeže osobních údajů neautorizovanými osobami. Ochrana dat na internetu se tedy musí neustále vyvíjet a zdokonalovat. Prezentace některých vypočítaných statistických ukazatelů a příslušných patientských dat na internetu a ochrana těchto dat je rovněž cílem této diplomové práce.

1. Nemocniční informační systémy (NIS)

Vedení záznamů o pacientech, léčivech a nemocích je staré již několik století. Dříve se však vedly pouze papírové záznamy. V těchto záznamech se obtížně vyhledávalo, vzhledem k jejich množství byly i veliké nároky na uskladňovací prostory. Náročná byla i duplikace a zálohování záznamů. Tyto problémy z větší míry řeší aplikace počítačů do zdravotnictví. První systémy orientované na samostatné dílčí oblasti nemocnice (kartotéky údajů, diagnózy, atd.) se začaly objevovat kolem roku 1950 při vzestupu počítačů v USA. V roce 1970 vznikla organizace IMIA (*International Medical Informatics Association*) ošetřující standardy a využití informačních systémů ve zdravotnictví.

Zdravotnická informatika je tedy soubor složený z aplikace informatiky, počítačových aplikací a zdravotní péče při řešení konkrétních úloh ve zdravotnictví. Jedná se o typickou interdisciplinární vědu. Je to soustava aplikací, technologií a postupů k získávání, uskladnění, nalezení a zpracování zdravotnických dat. Není to tedy izolovaný program s jediným účelem, ale kompletní správní systém zabývající se celou organizační strukturou nemocnice, nejen její lékařské části. Nástroje klinických informačních systémů nejsou jen počítače, ale i různé klinické směrnice, formální medicínské terminologie a informační a komunikační systémy.

Hlavní úkony NIS dle [1],[5]:

- Tvorba, uchování a sdílení zdravotnické dokumentace
- Administrativa činnosti nemocničního oddělení (zařízení)
- Podpora manažerského i odborného rozhodování
- Kontrola medicínských výrazů a diagnóz podle mezinárodních standardů

Klinický (nemocniční) informační systém pokrývá provoz klinických pracovišť řadou navzájem propojených produktů umožňujících vedení zdravotní dokumentace a podporujících provozní činnosti na jednotlivých klinických pracovištích. NIS zajišťují zadání potřebných administrativních údajů, pořizování výkaznických a statistických dat a podporují činnost lékařů a sester při dokumentaci zdravotního stavu pacienta.

Součástí řešení nemocničního informačního systému je patientská administrativa, která v sobě zahrnuje bloky funkcí vytvářejících prostředí pro efektivní počítačové zpracování základních administrativních činností, které jsou spojené s vedením patientské dokumentace. Tyto činnosti tvoří základní kostru informačního systému a jejich funkce jsou důležitým předpokladem pro navazující klinické procesy.

1.1. Základní funkce NIS

Záznam a komunikace

Jedná se o vstup, přenos a výstup dat. V dnešní době je třeba data šifrovat do tvaru nečitelného pro neoprávněné uživatele. Na této úrovni jsou data pouze krátkodobě uložena a přenesena na vzdálené místo při obdržení řídicího signálu.

Příklad použití: vizualizace biologických signálů jako je např. EKG v průběhu monitorování pacienta na jednotce intenzivní péče. Data jsou snímána a zobrazována na monitoru nebo vytištěna.

Ukládání a vyhledávání

Jedná se o předchozí stupeň doplněný o trvalé ukládání a vyhledávání dat v databázích. V současné době je možné ukládat poměrně levně obrovské množství dat a rychle je prohledávat. Nejrozsáhlejší medicínské databáze dosahují kapacity desítek až stovek TB.

Příklad použití: systém PACS (Picture Archiving and Communication Systems) pro archivaci obrazů,
pacientské zdravotní záznamy, viz. kapitola 1.3.

Vyhodnocení a automatizace

Týká se transformace dat před jejich uložením do paměti nebo přenesením na jiné místo. Praktické využití je získávání statistických a numerických analýz. Data jsou vyhodnocována na základě vstupních dat a vhodných algoritmů. Pro analýzu je často potřeba i spoluúčast lidského faktoru. Pokud známe vhodné matematické modely řešené úlohy, není spoluúčast člověka při vyhodnocení nutná.

Příklad použití: plánování terapie, kdy je lékař zodpovědný za výběr plánu a počítání slouží pouze jako podpora vyhodnocování medicínských obrazů (radiologie, nukleární medicína, atd.). Bez počítačů by toto vyhodnocování neexistovalo.

Rozpoznávání a rozhodování

V této oblasti je účast člověka nutná. Člověk musí počítači sdělit, jaký model zvolit a které znaky a příznaky jsou v něm relevantní. Přičemž výběr správných relevantních znaků a příznaků je pro rozhodování a rozpoznávání klíčový.

Příklad použití: diagnostická interpretace EKG běží zcela automaticky, ale konečné slovo při vyslovení závěru má lékař.

Terapie a řízení

Při řízení terapeutických procesů pomáhají počítače při výpočtu dávek v radioterapii, při regulaci tekutin na pooperačních odděleních, atd. Konečným cílem zpracování dat je dosažení konkrétního výsledku, například konkrétní dávky ozáření, nebo lepší náhled do průběhu choroby. Úspěšná řešení jsou jen v oblastech, které jsou plně teoreticky zvládnuty.

Příklad použití: radioterapie je upravována a kalibrována na základě počítačových výpočtů a výstupů.

Výzkum a vývoj

Jde o vývoj teorií popsaných matematickými modely a prověřovaných počítačovou simulací. Tyto modely jsou odvozovány buď na základě vědecky hodnověrných teorií nebo jen ze známých, leč zatím dostatečně nepopsaných jevů. V tomto odvětví je spolupráce člověka s počítačem nejužší. Počítačové modely přispívají k utřídění a potvrzení (či vyvrácení) teoretických předpokladů či domněnek. Je třeba pamatovat, že modely jsou jen simulace reálného stavu a je třeba je prakticky ověřit.

1.2. Medicínská informatika

Obor zabývající se aplikací počítačových informačních systémů do zdravotnictví se dle [2] nazývá medicínská informatika. Tento celkový soubor se dělí podle druhu aplikace na několik úzce zaměřených částí:

Lékařská informatika – zaměřuje se na zpracování informací pro potřeby lékařů při vedení elektronických zdravotních záznamů, při tvorbě klinických a epidemiologických studií a pro podporu diagnostického rozhodování

Zdravotnická informatika – zabývá se řízením a rozhodováním ve zdravotnictví a vytvářením zdravotnických informačních systémů na různých úrovních
Bioinformatika – aplikace informatiky do molekulární biologie

1.3. Pacientské zdravotní záznamy

Jedná se o základní prvek zdravotní dokumentace. Stále sice zůstává nutnost vést papírové záznamy, ale jejich využití je z důvodů uvedených v kapitole 1. značně neefektivní a je snaha o co největší integraci elektronických záznamů do zdravotnictví.

Automatizovaný zdravotní záznam:

Papírová dokumentace se nemění, ale některé informace o pacientovi jsou zpracovány počítačově a následně vytištěny.

Počítačový záznam

Vzniká skenováním papírové databáze do počítačové formy. Takto získané zprávy se nedají již dál zpracovávat.

Elektronický zdravotní záznam (*Electronic Patient Record*)

Je to strukturovaná databáze informací o jednotlivých pacientech, která obsahuje souhrn informací o zdravotním stavu jednotlivce a informací sloužících k jeho identifikaci. Dá se dále zpracovávat, jednoduše šířit mezi jednotlivými odděleními a získávat informace o zdravotním stavu rodinných příslušníků.

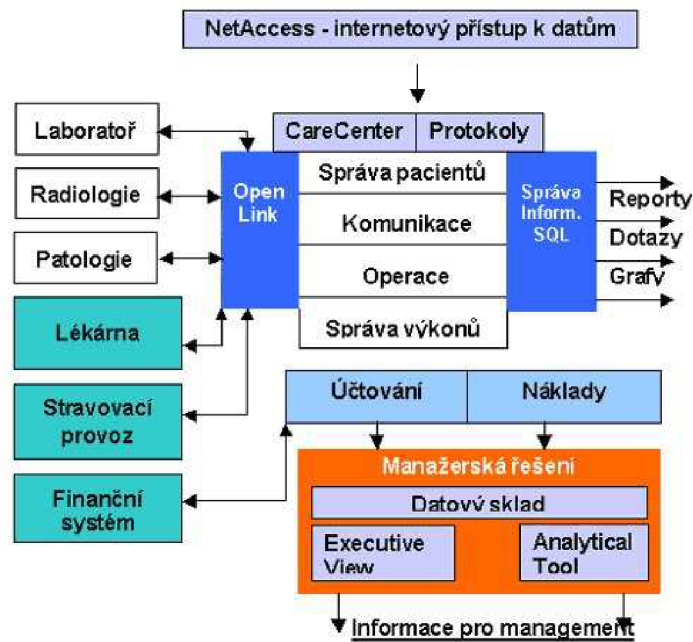
Elektronické zdravotní záznamy přispívají ke zlepšení péče o pacienta, ke snadnější administrativě, ke zkvalitnění procesů řízení a rozhodování ve zdravotnictví a slouží k vědeckým účelům.

1.4. Nemocniční informační systém Clinicom

NIS Clinicom je nemocniční informační systém s nadčasovým řešením a moderní postrelační databází Caché. Vychází ze zkušeností v USA a zemích EU. Byl vyvinut americkou firmou SMS a uzpůsoben pro české podmínky. V roce 2007 bylo na českém území 40 instalací NIS Clinicom. Stále však převažují jednodušší starší programy jako například Galileo.

NIS Clinicom je koncipován tak, aby byl v maximální míře postaven na proměnných položkách a číselnících. Pouhým přenastavením hodnot lze potom dosáhnout odlišného chování systému. To je velkým přínosem především v případě změn účtování lékařské péče, které mohou nastávat například v důsledku zdravotnické reformy.

Jedná se o systém s velkým stupněm možností uživatelského přizpůsobení. Umožňuje snadné propojení s jinými informačními systémy zdravotnického zařízení. Nastavení standardních komunikačních protokolů používaných ve zdravotnictví umožňuje snadnou komunikaci s dalšími subjekty a nabízí možnost přímého propojení do národních i nadnárodních zdravotnických sítí a on-line výměnu informací.



Obr. 1. Schéma fungování NIS Clinicom, přejato z [10]

NIS Clinicom stojí na čtyřech základních pilířích [8]:

Správa pacientů

Všechna data jsou zadávána pouze jedenkrát. Toto moderní pojetí přístupu k databázím vede například ke snížení chybovosti způsobené vícenásobným vložením a k časové úspoře (není třeba neustále vyplňovat stejnou informaci).

Správa výkonů

Zajištění rutinního chodu nezávislého na personálu. Uživatelé pouze vkládají uskutečněné výkony a o správný chod v souladu s legislativou a metodikou se stará systém. Změny legislativy sleduje pouze několik lidí odpovědných za nastavení kmenových souborů. Je možné výsledky vyšetření (výkonů) doplnit o obrazová data.

Komunikace

Vystavování žádanek, jejich odeslání, příjem výsledků s nastavitelnou úrovní rozlišení a vyúčtování.

Správa operací

Účinný nástroj pro sledování zákroků a operací za účelem získání přesných údajů o nákladech a výnosech.

Clinicom nepředstavuje jen administrativní nástroj lékaře pro usnadnění pořízení a archivace povinné dokumentace, ale je především navržen k sledování ekonomiky nemocnice. Data z NIS lze dále zpracovat a vyhodnotit prostřednictvím manažerského informačního systému DSS (Decision Support System).

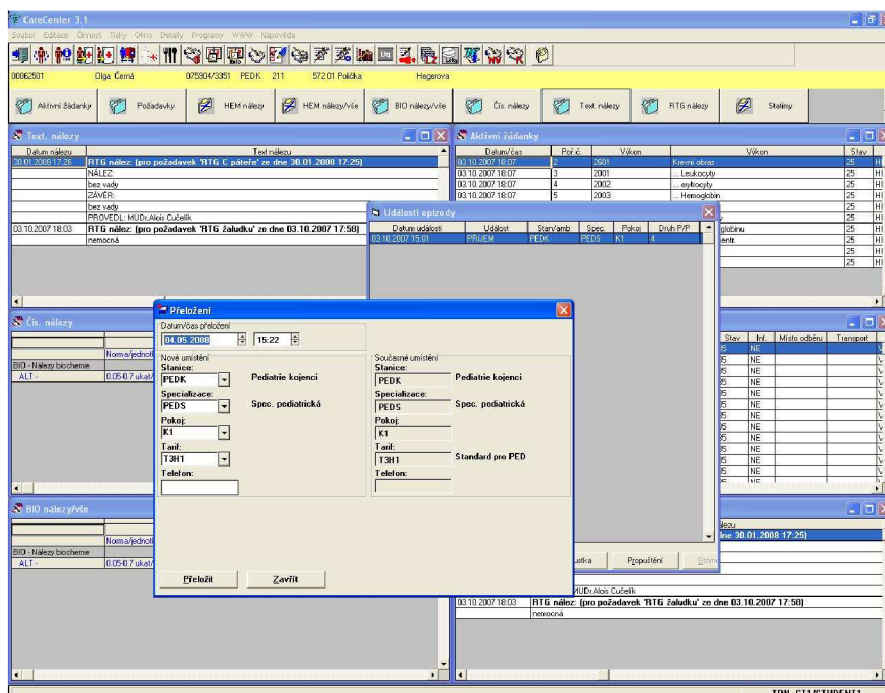
1.4.1. Uživatelská rozhraní systému Clinicom

Care Center

Care Center je grafické uživatelské rozhraní systému Clinicom, realizované ve standardním prostředí MS Windows. Toto prostředí poskytuje zdravotníkům jednoduchý přístup k datům. Care Center zprostředkovává okamžitý přístup k žádankám, nálezům, lékařské dokumentaci a k informacím o protokolech, a to s uspořádáním podle požadavků uživatele. Dále umožňuje uživateli jednoduchý pohled na informace o zvoleném pacientovi, ať už je třeba zobrazit nebo vytisknout plán péče, žádanky, nálezy, informace o vyšetření či textovou medicínskou dokumentaci.

Jediným přihlášením do systému má uživatel okamžitý přehled o pohybu a stavu pacienta od jeho přijetí, o ordinovaných a provedených vyšetřeních, výsledcích vyšetření až po ukončení léčby. Každý pacient je jednoznačně určen svým příjmovým číslem. Tím je zajištěna integrita pacientů. Vyhledávání pacientů je možné podle rodného čísla, příjmení, jména, atd.

Dostupnost urgentních výsledků je automaticky indikována, číselné výsledky vyšetření je možno zobrazit např. ve formě grafů a trendů. U výsledků jednotlivých vyšetření je možno přidat i obrazová data (např. RTG vyšetření).



Obr. 2. Ukázka rozhraní CareCenter

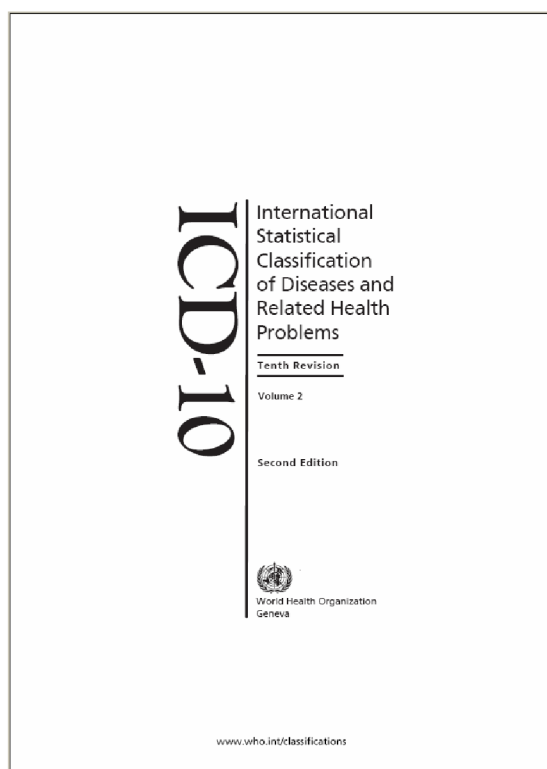
NetAccess

Představuje jednoduchý zabezpečený externí přístup k lékařským informacím prostřednictvím Internetu / Intranetu a umožňuje tak uživatelům přístup k datům NIS nejen z lokální sítě nemocnice, ale také z každého místa, které disponuje připojením na Internet.

NetAccess slouží k zadávání požadavků a zobrazení vybraných údajů. Pracuje se stejnou množinou dat jak Care Center a uživateli nabízí také velmi podobné grafické prostředí pod Windows.

Jedná se o aplikace klient – server, což přináší minimalizaci na zřízení a údržbu koncového zařízení. Veškeré informace a parametry jsou přenášeny v kódované podobě, navíc je zajištěno, že data nejsou ukládána do paměti počítače uživatele, což vede k tomu, že data nemůže nikdo později zobrazit.

2. Mezinárodní klasifikace nemocí (MKN)



Obr. 3. Úvodní přebal dokumentu ICD-10

Jedná se o metodu klasifikace nemocí kódováním jednotlivých diagnóz. Mezinárodně používaná zkratka je ICD (z anglického *International Classification of Diseases*). MKN se stala mezinárodním standardem diagnostické klasifikace pro epidemiologické a o zdravotní stav populace se zajímající organizace.

Soubor MKN-10 byl schválen na 41. Světovém zdravotnickém shromáždění v roce 1990. V praxi se používá od roku 1994, kdy ho oficiálně vydala WHO Member States (z anglického *World Health Organization*). Je to standard používaný v současné době. Myšlenka zdravotních registrů je stará už bezmála 200let. Původně šlo pouze o seznam příčin úmrtí. První vydání s názvem *Klasifikace příčin smrti* bylo přijato v roce 1893 na konferenci v Chicagu. Na konferenci v Paříži byla v roce 1946 přijata 6.revize mezinárodních seznamů pod názvem *Příručka mezinárodní statistické klasifikace nemocí, úrazů a příčin smrti*. Zároveň byly přijaty stanovy Světové zdravotnické organizace (WHO).

Některé tematické oblasti v klasifikaci byly považovány za nevhodně uspořádané a byl vyvíjen značný tlak na větší podrobnost a přizpůsobení klasifikace, aby se stala relevantnější pro zhodnocení lékařské péče zařazováním stavů do kapitol dělených podle postižené části těla, spíše než podle základní celkové nemoci. Proto vznikaly nové revize. Poslední revize používaná před zavedením revize MKN-10 byla MKN-9, zavedená v roce 1975 na konferenci v Ženevě.

2.1. MKN-10

MKN-10 dělí známé nemoci, chorobné stavy a události týkající se gravidity do 17 velkých skupin (I. – XVII.). Kapitola XIX. se týká poranění, otrav a některých jiných následků vnějších příčin. Zbývající kapitoly (XVIII, XX a XXI) doplňují tyto diagnózy.

Každá skupina má své další členění. Přehled základních skupin je na následujícím seznamu.

- I. Infekční a parazitární nemoci **A, B**
- II. Novotvary **C, D**
- III. Nemoci krve a imunity **D**
- IV. Nemoci endokrinní a metabolické **E**
- V. Nemoci duševní a poruchy chování **F**

- VI.** Nemoci nervové soustavy **G**
- VII., VIII.** Nemoci oka a očních adnex, nemoci ucha **H**
- IX.** Nemoci oběhové soustavy **I**
- X.** Nemoci dýchací soustavy **J**
- XI.** Nemoci trávicí soustavy **K**
- XII.** Nemoci kůže a podkožního vaziva **L**
- XIII.** Nemoci svalové a kosterní soustavy **M**
- XIV.** Nemoci močové a pohlavní soustavy **N**
- XV. – XVII.** Těhotenství, porod, šestinedělí, perinatální stavy, vrozené vady, deformace, malformace **O, P, Q**
- XVIII.** Příznaky, znaky a nálezy nezařazené jinde **R**
- XIX.** Poranění, otravy, následky působení vnějších příčin **S, T**
- XX.** Vnější příčiny nemocnosti **V, W, X, Y**
- XXI.** Faktory ovlivňující zdravotní stav **Z**

Klasifikace je tedy rozdělena do 21 kapitol. První znak kódu MKN je písmeno a každé písmeno je sdruženo s jednotlivou kapitolou až na písmeno D, které je užito jak pro kapitolu II. Novotvary, tak pro kapitolu III. Nemoci krve, krvetvorných orgánů a některé poruchy týkající se mechanismu imunity a písmeno H, které je užito jak v kapitole VII. Nemoci oka a očních adnex, tak v kapitole VIII. Nemoci ucha a bradavkového výběžku. Více než jedno písmeno na prvním místě svých kódů využívají čtyři kapitoly (Kapitoly I., II., XIX. a XX.). Za znakem následují 2 až 3 číslice pro zpřesnění diagnózy.

Pro ukázkou je uvedeno několik příkladů diagnózy zapsané pomocí standardu MKN-10:

C00.0 ZN- horní ret, zevní
 C00.1 ZN- dolní ret, zevní
 C00.2 ZN- ret zevní, NS
 C00.3 ZN- horní ret, vnitřní strana

C02.0 ZN- hřbetní strana jazyka
 C02.1 ZN- hrana jazyka
 C02.2 ZN- spodní strana jazyka
 C02.3 ZN- přední dvě třetiny jazyka
 C02.4 ZN- jazyková mandle
 C02.8 ZN- léze přesahující jazyk

ZN - zhoubný novotvar

Z ukázky je patrné, že první 2 číslice uvádějí oblast výskytu choroby (jazyk, plíce, ...) a poslední číslice tuto oblast ještě více zpřesňuje (přední strana, spodní strana, ...).

Každá kapitola obsahuje dostatek třímístných položek k pokrytí svého obsahu. Všechny kódy, které jsou k dispozici, nejsou využity, aby bylo poskytnuto místo pro budoucí revizi a rozšíření.

Kódy U00-U49 se budou používat pro prozatímní přidělení novým nemocem nejisté etiologie. Kódy U50-U99 mohou být použity ve výzkumech (například zkouší-li se možnosti alternativního podtřídění pro zvláštní projekt).

2.2. Třímístné položky

Uvnitř každé skupiny (bloku) jsou některé z třímístných položek vybrány pro jeden stav, protože tento stav je buď četný, závažný nebo ovlivnitelný zdravotní intervencí, zatímco jiné třímístné položky jsou pro skupiny nemocí s některou společnou charakteristikou. Obvykle je zde též kategorie pro "jiné" stavy, které by se měly zařadit, což připouští zahrnout mnoho odlišných vzácnějších, ale i "neurčených" stavů.

2.3. Čtyřmístné položky

Ačkoliv nejsou povinné pro hlášení na mezinárodní úrovni, je většina třímístných položek dále rozdělena pomocí čtvrtého místa. Znak po desetinné tečce připouští až deset podpoložek. Tam, kde třímístná položka není dále dělena, doporučuje se, aby se použilo písmeno "X" k vyplnění čtvrtého místa tak, aby kódy měly standardní délku pro zpracování dat. Je-li třímístná položka pro jednotlivou nemoc, čtyřmístné podpoložky se užívají jakýmkoliv co nejvhodnějším způsobem např. k rozlišení rozdílných lokalizací či variant. Je-li třímístná položka pro skupinu stavů, čtyřmístné podpoložky jsou užívány k identifikování jednotlivých nemocí.

Čtvrté místo .8 je obvykle hlavně pro "jiné" stavy patřící do třímístné položky a .9 se většinou používá k převodu názvu téhož smyslu jako má název třímístné položky bez připojení jakékoliv další informace.

2.4. Dva kódy pro stejný stav (Systém "křížků a hvězdiček")

MKN-9 zavedla systém, který v MKN-10 pokračuje. V tomto systému jsou dva kódy pro jeden stav, kde jedno zařazení je do hlavní nemoci a druhé udává postižení konkrétního orgánu nebo umístění, které je klinickým problémem samo o sobě.

Primární kód patří základnímu onemocnění a je označen křížkem (+), volitelný dodatkový kód pro popis stavu je označen hvězdičkou (*). Tato dohoda byla přijata proto, že kódování samotného základního onemocnění často bylo nedostatečné pro statistické účely ve vztahu k určitým specializacím (bylo žádoucí mít stav zakódován v kapitole určené projevem nemoci, pokud ten byl důvodem lékařské péče).

Protože systém křížků a hvězdiček umožňuje alternativní klasifikaci pro statistické účely, je stanoveno, že diagnóza s křížkem je primární a musí tak být užitá vždy. Pro užití kódu s hvězdičkou platí, že jde o dodatkové užití a smí být užit jako alternativní prezentace na vyžádání. Pro kódování nesmí být kód s hvězdičkou nikdy použit samostatně. Pro tradiční klasifikaci a prezentaci dat pro mortalitu, morbiditu a další aspekty zdravotní péče se používá kód s křížkem.

Příklad:

A18.1+ Tuberkulóza močové a pohlavní soustavy

Tuberkulóza:

- močového měchýře (N33.0*)
- hrdla (čípku) děložního (N74.0*)
- ledvin (N29.1*)
- mužských pohlavních orgánů (N51.-*)
- ureteru (N29.1*)

Tuberkulózní zánětlivé onemocnění ženských pánevních orgánů (N74.1*)

3. Národní zdravotní registry

Jsou naplňovány údaji o výskytu vybraných společensky závažných onemocnění a stavů. Údaje o těchto onemocněních jsou součástí zdravotnické dokumentace. Jsou součástí Národního zdravotnického informačního systému (NZIS).

V roce 2004 vstoupila v platnost novela zákona č. 20/1966 Sb., o péči o zdraví lidu, která vymezila osobní a další údaje, které lze bez souhlasu subjektu údajů v registrech zpracovávat.

V zákoně je rovněž stanoven okruh zdravotnických zařízení předávajících požadované osobní a další údaje do NZIS pro potřeby vedení registrů, způsob předávání, periodicitu a lhůty předání. Zdravotnická zařízení jsou povinna předávat údaje do registrů v elektronické nebo listinné podobě. Preferována je elektronická podoba předávání podle Datového standardu MZ ČR.

Přehled současných registrů v rámci NZIS:

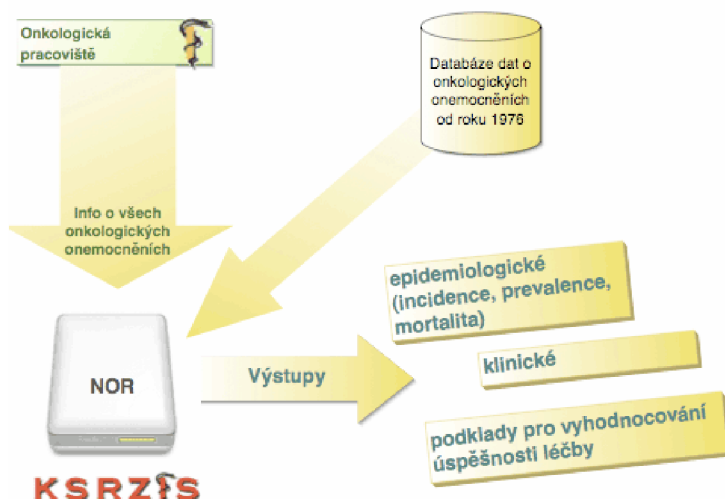
- Národní registr hospitalizovaných (NRHOSP)
- Národní registr rodiček (NRROD)
- Národní registr novorozenců (NRNAR)
- Národní registr vrozených vad (NRVV)
- Národní registr potratů (NRPOT)
- Registr lékařů, zubních lékařů a farmaceutů (RLZF)
- Národní registr uživatelů lékařsky indikovaných substitučních látek (NRUSL)
- Národní onkologický registr (NOR)
- Národní registr cévní chirurgie (NRCCH)
- Národní kardiochirurgický registr (NKCHR)
- Národní registr kloubních náhrad (NRKN)
- Národní registr kardiovaskulárních intervencí (NRKI)
- Národní registr nemocí z povolání (NRNP)
- Národní registr osob nesouhlasících s posmrtným odběrem tkání a orgánů
- Národní registr asistované reprodukce

Vzhledem k tomu, že při řešení diplomové práce bylo použito postupů a údajů z Národního onkologického registru, je mu věnována následující kapitola.

3.1. Národní onkologický registr

Registr novotvarů byl ustaven v roce 1976. Oproti předchozímu zjišťování výskytu novotvarů byla ustanovena evidence jednotlivých osob s diagnostikovaným zhoubným novotvarem včetně následného sledování každého pacienta prostřednictvím kontrolního hlášení. Od roku 1987 jsou v registru sledovány i novotvary neurčené povahy a neurčené morfologie. Od roku 1989 je registr doplňován každoročně o tzv. DCO případy, tj. onemocnění, která jsou hlášena pouze na základě Listu o prohlídce mrtvého. V roce 1991 byl Národní onkologický registr České republiky (*Czech Cancer Registry*) přijat za člena Mezinárodní asociace registrů rakoviny (*IACR*).

Národní onkologický registr - NOR



Obr. 4. Schéma fungování NOR, přejato z [7]

důsledků. Za sledování a doplňování registru jsou odpovědná příslušná zdravotnická zařízení, na kterých bylo onemocnění diagnostikováno a léčeno. Údaje o pacientech do registru zasílají speciálně určená a vyškolená zdravotnická pracoviště.

Onkologickému hlášení podléhají onemocnění uvedená v následující tabulce. Jedná se o skupiny diagnóz II. kapitoly Mezinárodní statistické klasifikace nemocí a přidružených zdravotních problémů - desáté revize (MKN-10).

Tabulka 1.: Onemocnění podléhající onkologickému hlášení

Onemocnění ze skupiny diagnóz:	Kód dle MKN-10
a) zhoubné nádory	C 00 – C97
b) onemocnění hodnocená jako novotvary in situ	D 00 – D 09
c) novotvary nejistého nebo neznámého chování	D 37 – D48
d) některé nemoci postihující lymforetikulární tkáň a retikulohistiocytární systém	D 76.0

Od 1. ledna 2006 nastaly v organizaci sběru dat do Národního onkologického registru (NOR) významné změny. Změnil se obsah datové struktury pro NOR, včetně podkladu pro sběr a pořizování dat, kterým je nově upravené hlášení „Incidence a léčba zhoubného novotvaru“. Současně dochází i k významné změně v definici subjektu, který hlášení novotvaru vyplňuje. Na vyplňování hlášení se nyní podílejí dvě zdravotnická pracoviště [9]:

- zdravotnická pracoviště, která stanoví diagnózu (vyplní a předá do 1 měsíce od stanovení diagnózy)
- zdravotnická pracoviště, zodpovědná za léčbu (vyplní a předá do 8 měsíců od stanovení diagnózy)

Úlohou NOR je registrace onkologických onemocnění, tj. shromažďování dat, jejich verifikace, ukládání, ochrana a zpracování. NOR poskytuje souhrnné údaje pro statistické přehledy, které jsou využívány jak na národní, tak i mezinárodní úrovni, dále pro epidemiologické studie a zdravotnický výzkum. Údaje registru slouží také k podpoře včasné diagnostiky a léčby novotvarů a přednádorových stavů, ke sledování trendů jejich výskytu, příčinných faktorů a společenských

3.2. Hlášení zhoubného novotvaru

Dle [9] se hlášení podává na formuláři „Incidence a léčba zhoubného novotvaru“. Povinnost hlásit nález zhoubného novotvaru má každé zdravotnické zařízení, které stanovilo diagnózu (1. část formuláře „Incidence a léčba zhoubného novotvaru“), a každé zdravotnické zařízení, které odpovídá za léčení (2. část formuláře „Incidence a léčba zhoubného novotvaru“) a dispenzarizaci onemocnění ze skupiny diagnóz zhoubné novotvary (viz. Tabulka 1). Hlášení „Incidence a léčba zhoubného novotvaru“ je třeba vyplnit a předat (z osobních údajů pacienta se přenáší pouze rodné číslo) na příslušné spádové pracoviště NOR do jednoho měsíce (1. část formuláře „Incidence a léčba zhoubného novotvaru“) a do osmi měsíců (2. část formuláře „Incidence a léčba zhoubného novotvaru“) od data stanovení diagnózy .

Zdravotnické pracoviště, které diagnostikovalo zhoubný novotvar (viz. [9])

Vyplní trojmo 1. část hlášení. Do jednoho měsíce od data stanovení diagnózy odešle originál na adresu spádového pracoviště Národního onkologického registru podle místa trvalého bydliště pacienta, u něhož byl nádor zjištěn a první kopii na adresu zdravotnického pracoviště zodpovědného za léčbu. Druhá kopie zůstane na pracovišti, které diagnostikovalo novotvar, jako součást zdravotnické dokumentace. Pokud byl zhoubný novotvar diagnostikován na základě provedené operace, byl zjištěn na základě pitvy nebo je hlášen na základě diagnózy uvedené na Listu o prohlídce mrtvého (DCO), vyplní pracoviště, které diagnostikovalo zhoubný novotvar i tomu odpovídající položky ve druhé části Hlášení.

Zdravotnické pracoviště zodpovědné za léčbu (viz. [9])

Vyplní dvojmo 2. část hlášení. Do osmi měsíců od data stanovení diagnózy (tj. 7 měsíců od vyplnění 1. části hlášení diagnostikujícím pracovištěm) odešle originál na adresu spádového pracoviště Národního onkologického registru podle místa trvalého bydliště pacienta a kopie zůstane na hlásícím pracovišti jako součást zdravotnické dokumentace. Toto pracoviště zodpovědné za léčbu může, v případě potřeby, změnit (kromě rodného čísla pacienta a pracoviště, které diagnostikovalo zhoubný novotvar) i údaje již dříve vyplněné pracovištěm, které diagnostikovalo zhoubný novotvar v 1. části hlášení.

4. Zabezpečení dat na internetu

Se stále rostoucím využitím internetu pro různé aplikace se na veřejně přístupné síti začíná objevovat množství osobních a citlivých dat uživatelů. Jedná se o tak důležitá data jako přístupová hesla do internetového bankovníctví, zdravotní údaje osob, kontaktní údaje a mnoho dalších. Nikdo z nás by jistě nebyl rád, aby se takto citlivá data vyskytovala na internetu „jen tak“ a lehce přístupná všem. Proto je třeba data nějakým způsobem chránit. Ochranu dat můžeme rozdělit na dvě oblasti:

- zabránění přístupu nepovolaných osob na klienta, server a komunikační cestu. Do této skupiny patří například využití firewallu, hesel a poučení uživatelů o zachování bezpečnosti dat.
- zajištění dat šifrou. Do této skupiny patří ku příkladu kryptografie, protokol SSL, a další.

Jelikož součástí této práce je i rozbor zabezpečení dat na internetu, popíšme si výše uvedené prostředky pro ochranu dat podrobněji.

4.1. Firewall

Je síťové zařízení sloužící k bezpečnému oddělení vnitřní sítě od internetu. Toto zabezpečení je nastaveno tak, aby uživatelé vnitřní sítě mohli volně přistupovat k informacím na internetu, ale aby nevyžádaná a případně nebezpečná data (např. počítačové viry) nemohla vstoupit z internetu do vnitřní sítě. Obecně nemusí firewall oddělovat jen vnitřní síť od internetu, ale může od sebe oddělovat jakékoli nezávislé sítě. Oddělení vnitřní sítě právě od internetu je ale v praxi nejběžnější úkol firewallu.

Nejjednodušším a nejstarším typem firewallu jsou **paketové filtry**. Princip ochrany spočívá v tom, že pravidla přesně uvádějí, z jaké adresy a portu na jakou adresu a port může být doručen procházející paket.

Stavové paketové filtry jsou již modernější variantou firewallů. Kontrola je prováděna stejně jako u paketových filtrů, navíc si ale stavové paketové filtry ukládají informace o povolených spojeních, které pak mohou využít při rozhodování, zda procházející pakety patří do již povoleného spojení a mohou být propuštěny, nebo zda musí znovu projít rozhodovacím procesem. Tyto filtry jsou schopny kontrolovat spojení až na úroveň korektnosti procházejících dat známých protokolů a aplikací.

Do dnešních verzí firewallů se integrují tzv. *in-line IDS (Intrusion Detection Systems – systémy pro detekci útoků)*. Tyto systémy pracují podobně jako antiviry a jsou schopny odhalit vzorce útoků i ve zdánlivě nesouvisejících pokusech o spojení, např. skenování adresního rozsahu, rozsahu portů, známé signatury útoků uvnitř povolených spojení apod.

Výhodou těchto systémů je vysoká úroveň bezpečnosti kontroly procházejících protokolů při zachování relativně snadné konfigurace a poměrně vysoká rychlost. Nevýhodou je zejména to, že z hlediska bezpečnosti designu je základním pravidlem bezpečnosti udržovat bezpečnostní systémy co nejjednodušší a nejmenší. Tyto typy firewallů integrují obrovské množství funkcionality a zvyšují tak pravděpodobnost, že v některé části jejich kódu bude zneužitelná chyba, která povede ke kompromitování celého systému.[20]

Firewall ukládá informace o své činnosti do souborů, ze kterých lze zpětně získat informace o dílčích akcích, popř. i vytvářet sumární přehledy. Aktivní firewally jsou kromě tohoto zápisu schopny způsobit jistou akci – alert. Tou může být například odeslání zprávy (SMS, e-mail, ...) pověřené osobě, zápis IP adresy útočníka na černou listinu, apod.

Jedny z používaných firewallů jsou: Raptor, Gauntlet, Norton Personal Firewall, atd.

4.2. Hesla

Obecně jsou hesla prostředkem k autentizaci uživatele. Uživatel je pokládán za oprávněného, pokud prokáže znalost hesla. Nutno říct, že z hlediska bezpečnosti je heslo použitelné pouze tehdy, není-li ostatním uživatelům známé. Proto musí oprávněný uživatel držet heslo v tajnosti.

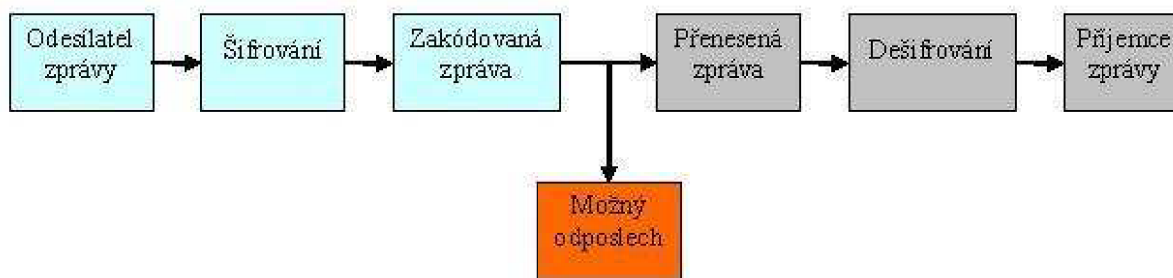
Hesla mohou mít obecně různou úroveň zabezpečení. V praxi jsou na jednodušších internetových stránkách hesla řešena poměrně jednoduchým způsobem. „Správné“ heslo je uloženo někde ve zdrojovém kódu stránky a heslo zadané uživatelem je pak s tímto porovnáváno. Porovnávání vykonává např. programovací jazyk php, nebo v této práci využitě csp. Jelikož překladač zdrojového kódu vypíše jen kód přeložený do html, nezobrazí se příkazy psané v jiných programovacích jazycích, ale pouze jejich výsledky.

U webových aplikací pracujících s citlivějšími daty, které vyžadují vyšší stupeň ochrany (např. internetové bankovníctví), jsou prostředky autentizace uživatele pomocí hesla složitější a sofistikovanější. Přístupové heslo na straně serveru nebývá např. uchováno v textové podobě, ale je znehodnoceno jednocestnou funkcí. Touto funkcí je algoritmus, který umožní technicky snadno určit výsledek, ale najít k výsledku původní text je technicky náročné. Toto opatření slouží proti zneužití chráněných informací správcem systému.

Někdy se v praxi využívají tzv. jednorázová hesla. Jedná se o seznam hesel vygenerovaný systémem a předaný uživateli (v tištěné podobě, na magnetické kartě, apod.). Na straně serveru mohou být hesla opět znehodnocena jednocestnou funkcí. Uživatel zadává jedno heslo po druhém, každé však jen jednou. Po vyčerpání seznamu je vygenerován seznam nový. Sofistikovanějšího využití jednorázových hesel je dosaženo zapojením systému GSM. V okamžiku, kdy se má klient autentizovat, uvede v aplikaci své přihlašovací jméno. Aplikace vyhledá na základě tohoto jména v databázi uživatelů příslušné číslo mobilního telefonu (které tam uživatel při zřizování účtu vloží) a odešle na něj vygenerované jednorázové heslo. Tento systém je využíván např. v internetovém bankovníctví při ověřování uživatele v případě převodu peněz z účtu po internetu.

4.3. Kryptografie

Kryptografie je obecně nauka o metodách utajování/šifrování smyslu zpráv převodem do podoby, která je čitelná jen se speciální znalostí daného šifrovacího mechanismu. Při přenosu zašifrovaných zpráv se obecně postupuje podle schématu uvedeného na Obr.5.



Obr. 5. Schéma postupu při šifrování dat.

Kvalita ochrany zprávy šifrováním je dána volbou šifrovací metody (Symetrické a Asymetrické šifrování), typem užitého algoritmu, jeho aplikací a délkou šifrovacího klíče.

4.3.1. Symetrické šifrování

Podstatou symetrického šifrování je použití stejného klíče na zašifrování i dešifrování. Tento klíč si musí osoby, které spolu chtějí takto komunikovat, sdělit nějakým důvěryhodným způsobem. Nesmí ovšem dopustit, aby se k němu dostala nepovolaná osoba. Celý přenos probíhá principiálně podle Obr. 6.



Obr. 6. Symetrické šifrování.

Podstatnou výhodou symetrických šifer je jejich nízká výpočetní náročnost. Navíc i při relativně krátkém šifrovacím klíči je i pro současnou techniku náročné klíč uhádnout. S prodlužováním klíče navíc bezpečnost šifrování roste. Na druhou stranu velkou nevýhodou je nutnost sdílení tajného klíče. Navíc nelze důvěryhodně určit, kdo odpověď odeslal a kdo ji převzal.

4.3.2. Asymetrické šifrování

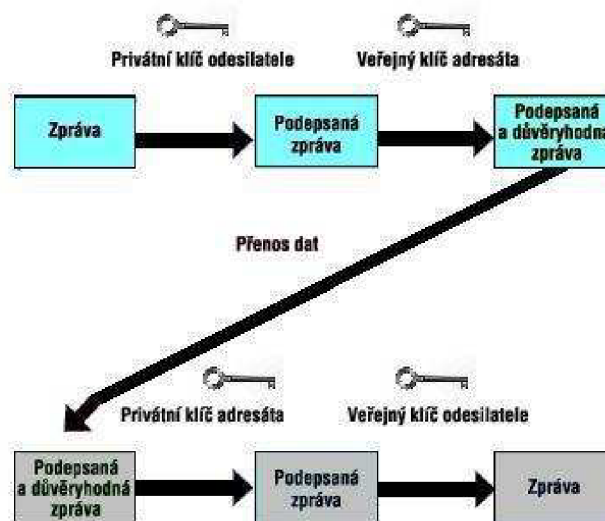
Tento typ šifrování využívá dvou šifrovacích klíčů, jednoho pro šifrování a druhého pro dešifrování. Jelikož operace šifrování a dešifrování jsou u některých šifer zaměnitelné, hovoříme o klíčích privátním a veřejném. Postup při šifrování vypadá následovně:

1. příjemce zprávy si vygeneruje dvojici klíčů: veřejný a soukromý
2. soukromý klíč si uloží na spolehlivé místo

3. distribuuje svůj veřejný klíč
4. odesílatel šifruje zprávu veřejným klíčem příjemce
5. příjemce dešifruje zprávu svým soukromým klíčem.

Odesílatel a příjemce nemusí sdílet tajný klíč, což je základní výhodou asymetrické kryptografie. Klíče jsou ale alespoň o řád delší než klíče u symetrického šifrování. Z tohoto důvodu je asymetrické šifrování pomalejší.

Výše podepsaným způsobem se zpráva zašifruje, ale příjemce nepozná, kdo ji poslal. Aby byla možná autorizace odesílatele, musí tento zašifrovat zprávu nejprve svým privátním klíčem. Tím je zpráva „podepsána.“ Celý algoritmus asymetrického šifrování i s „podepsáním“ zprávy je popsán na Obr.7.



Obr. 7. Asymetrické šifrování.

4.4. SSL

Zkratka SSL je odvozena z anglického *Secure Socket Layer*. Jedná se o protokol vložený mezi transportní (nejčastěji TCP/IP) a aplikační (např. HTTP) vrstvu. Pokud aplikační protokol HTTP provozujeme přes SSL, využíváme vlastně HTTPS, což je bezpečná komunikace s internetovými servery. Tento protokol vypadá uvnitř úplně stejně jako původní HTTP, jen před odesláním jsou data zašifrována a elektronicky podepsána. Kromě známého HTTP je možno šifrovat SSL technikou např. při stahování a odesílání emailů (POP3, SMTP, IMAP) nebo při přenosu dat pomocí FTP.

SSL spojení funguje na principu asymetrické šifry (viz. kapitola 4.3.2. Asymetrické šifrování). SSL spojení probíhá následovně: „

1. Klient pošle serveru požadavek na SSL spojení, spolu s různými doplňujícími informacemi (verze SSL, nastavení šifrování atd.).
2. Server pošle klientovi na jeho požadavek odpověď, která obsahuje stejný typ informací a hlavně certifikát serveru.
3. Podle přijatého certifikátu si klient ověří autentičnost serveru. Certifikát také obsahuje veřejný klíč serveru.

4. Na základě dosud obdržených informací vygeneruje klient základ šifrovacího klíče, kterým se bude šifrovat následná komunikace. Ten zašifruje veřejným klíčem serveru a pošle mu ho.
5. Server použije svůj soukromý klíč k rozšifrování základu šifrovacího klíče. Z tohoto základu vygenerují jak server, tak klient hlavní šifrovací klíč.
6. Klient a server si navzájem potvrdí, že od teď bude jejich komunikace šifrovaná tímto klíčem. Fáze handshake tímto končí.
7. Je ustaveno zabezpečené spojení šifrované vygenerovaným šifrovacím klíčem.
8. Aplikace od teď dál komunikují přes šifrované spojení. Například POST požadavek na server se do této doby neodešle.“

Citováno z [24].

4.5. Zabezpečení využité v diplomové práci

Na Obr.8 je ukázka autentizace uživatele zadáním hesla využité v této diplomové práci. Po zadání hesla a potvrzení tlačítkem „Přihlásit“ vyhodnotí program heslo. Vyhodnocení je provedeno porovnáním „správného“ hesla uloženého ve zdrojovém kódu stránky s heslem zadaným uživatelem, viz. níže.

Obr. 8. Přihlášení do webové aplikace vytvořené v rámci diplomové práce.

Uvedme si na ukázkou část přeloženého zdrojového kódu této přihlašovací části webové aplikace:

```
<table align="center">
  <tr><font color="white"><h2>P&#345;hlášení </h2></font></tr>
  <tr><td><font color="white">Jméno:</td></font>
    <td><input type="text" name="jmeno" value="student"></td>
  <br><tr><td></td></tr></td>
  <tr><td><font color="white">Heslo:</td><td></font>
    <td><input type="password" name="heslo" value=""><br></td><br>
  <center>
    <input type="submit" name="prihlas" value="P&#345;hlásit"></center>
```

Z kódu je patrné, že tato část jen načítá hodnoty zadané uživatelem. K porovnání hesla zadaného a referenčního (uloženého v proměnné) dochází až v dalším kroku po stisknutí tlačítka „Přihlásit“. Příkaz vykonávající autentizaci uživatele porovnáním hesel je umístěn mezi tagem *BODY* a tagem *DIV*. V ukázce kódu na další stránce je pro lepší orientaci zvýrazněn žlutou barvou:

```

<BODY bgcolor="#f4a460">
<CSP:IF CONDITION='$_G(%request.Data("heslo",1))="xxx"'>
<div id="content">
<div id="logo">
  <table align="center" width="100%" high="100%" cellpadding="6">
    <tr><td align="left" width="10%"></td>
    <td align="center" ></tr></td>
  </table>
</div>

```

Nutno dodat že v programu je ve skutečnosti místo hodnoty „xxx“ uloženo jiné, skutečně použité heslo. Uvedme si nyní opět ukázkou části přeloženého zdrojového kódu provádějícího již autentizaci uživatele:

```

<BODY bgcolor="#f4a460">
<div id="content">
<div id="logo">
  <table align="center" width="100%" high="100%" cellpadding="6">
    <tr><td align="left" width="10%"></td>
    <td align="center" >
      </tr></td>
  </table>
</div>

```

Porovnáním obou kódů zjistíme, že funkce porovnávající hesla se v přeloženém kódu neobjeví. To je z hlediska zabezpečení podstatné, neboť to ztěžuje případnému útočníkovi možnost heslo odposlechnout.

Webová aplikace vytvořená v rámci této diplomové práce je koncipována tak, že do hlavního okna se načítají funkce napsané v jiných souborech CSP, viz. kapitola 7. Z tohoto důvodu je nutné zabezpečit i tyto ostatní soubory. Pokud by totiž uživatel zadal přímo cestu k souboru např. <http://147.229.77.5:19727/csp/trn/velicka/Osetreni.csp>, tak by se mu zobrazila samostatná stránka s údaji o hospitalizovaných pacientech i bez zadání hesla.

V této webové aplikaci je zabezpečení souborů kromě kontroly hesla na vstupní stránce řešeno ještě přenášením určité informace přes tzv. *SESSION*, viz. kapitola 6. Pro další výklad je nutno zobrazit ukázkou zdrojového kódu z naší webové aplikace, konkrétně z úvodní stránky, jejíž zdrojový kód je uložen v souboru *DPVelicka.csp*:

```

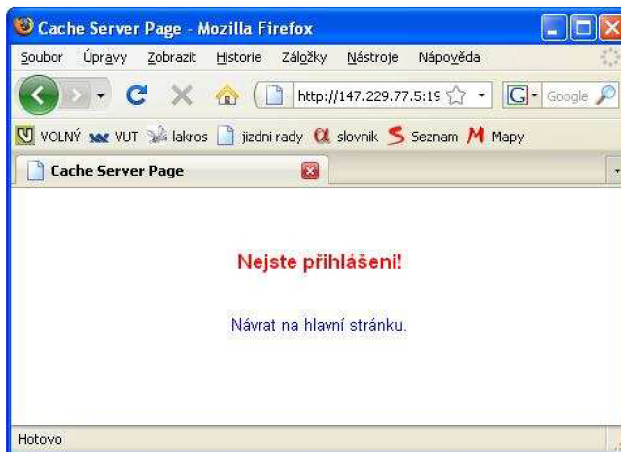
<SCRIPT LANGUAGE="CACHE" RUNAT="SERVER">
  Set %session.Data("pristup") = ""
</SCRIPT>
<CSP:IF CONDITION='$_G(%request.Data("heslo",1))="xxx"'>
  <SCRIPT LANGUAGE="CACHE" RUNAT="SERVER">
    set %session.Data("pristup") = "ok"
  </SCRIPT>

```

Nejprve je iniciována relace s názvem *pristup* a její obsah je nastaven na prázdný. Následně je porovnáno heslo zadané uživatelem s heslem referenčním, jak jsme si popsali výše. Pokud je heslo zadané uživatelem správné, je do relace uloženo slovo „ok“. Na všech dalších stránkách, jejichž zdrojové kódy jsou zapsány v dalších souborech (viz. příloha č.1), je porovnávána hodnota relace *pristup*. Pokud její hodnota vyhovuje podmínce:

```
<CSP:IF CONDITION='%session.Data("pristup") = "ok"'>
```

je umožněn přístup k funkci této stránky. V případě nesplnění podmínky je uživatel informován o nepovolení přístupu a odkázán na hlavní stránku, jak je doloženo na Obr.9.



Obr. 9. Nepovolení přístupu ke stránce.

Jen v krátkosti si vysvětlíme princip přenosu informace o relaci *pristup* mezi jednotlivými stránkami. Hodnota relace je totiž přenášena pomocí tzv. *COOKIE*. Jedná se o malé množství dat, která server pošle prohlížeči, který je uloží na počítači uživatele. Při každé další návštěvě téhož serveru (např. při otevření další stránky) pak prohlížeč tato data posílá zpět serveru, který vykoná porovnání tak, jak jsme si popsali výše. Takto provedená autentizace je tedy poměrně jednoduchá a bezpečná. Dá se však také zneužít, viz. [23], a proto je v prohlížečích možnost *COOKIES* vypnout. V případě, že má tedy uživatel *COOKIES* vypnuté, bude ho prohlížeč i po správném zadání hesla informovat o chybě. Tomuto se můžeme vyhnout pouze informováním uživatele, že pro správnou činnost stránek je třeba mít zapnuté *COOKIES*.

Nicméně je třeba dodat, že takovéto zabezpečení není pro ochranu citlivých dat dostačující a u webových aplikací vyžadujících vyšší úroveň zabezpečení se používá jiných metod autentizace uživatele pomocí hesla. Takto uložené heslo je totiž jednoduše dostupné správci systému, což je například u internetového bankovníctví nepřijatelné.

5. Databázové systémy v NIS Clinicom

5.1. Caché

Jedná se o vyspělý nástroj pro tvorbu komplexních aplikací vyvinutý firmou InterSystems sídlící v USA. Databázová platforma Caché byla na trh uvedena v roce 1997. Jádrem celého systému je server, který plní dvojí roli. Na nejnižší úrovni je to role databázového stroje. Tento stroj má na starost manipulaci s daty, jejich ukládání, vybírání a aktualizaci v databázi.

Podstatným rysem Caché je, že neobsahuje pouze uživatelská data, ale jsou v ní uloženy i prvky návrhu, které z pasivního datového úložiště vytvářejí aplikaci. Nad databázovým strojem „sídlí“ aplikační server, který vytváří běhové prostředí pro aplikace Caché. Datový a aplikační server tvoří navzájem neoddelitelné části, které se navzájem doplňují.

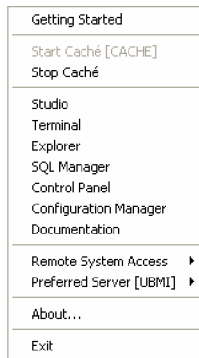
Caché je databázová technologie nové generace označovaná jako „postrelační“. Jako postrelační databáze kombinuje objektovou databázi, jazyk SQL a rychlý přístup k vícerozměrným datům. Všechny těmito způsoby lze přistupovat k týmž datům současně. Data jsou popsána pouze jednou v jediném integrovaném slovníku dat (fyzicky uloženo v souborovém systému jako samostatný soubor s názvem CACHE.DAT) a jsou okamžitě dostupná prostřednictvím všech přístupových metod. Systém Caché poskytuje takovou úroveň výkonu, rozšiřitelnosti, rychlého programování a snadného použití, která je relačními technologiemi nedosažitelná.

Kromě toho, že je Caché databázová technologie, zahrnuje i aplikační server s progresivními možnostmi objektového programování, schopnost snadné integrace se širokou paletou technologií a mimořádně výkonný databázový stroj s jedinečnou technologií vyrovnávacích pamětí.

5.1.1. Caché Cube

Jedná se o vývojové prostředí, které obsahuje administrativní nástroje potřebné pro správu celé databáze. Pomocí tohoto prostředí jsme schopni nastavit a ovládat databáze Caché i na vzdálených serverech.

Uveďme si nyní, z jakých částí se „kostka“ Caché skládá. Podrobněji si popíšeme jen aplikace, které byly v této práci přímo využívány. Na Obr.10 je ukázka menu, které nám nabízí právě nástroje „kostky“.



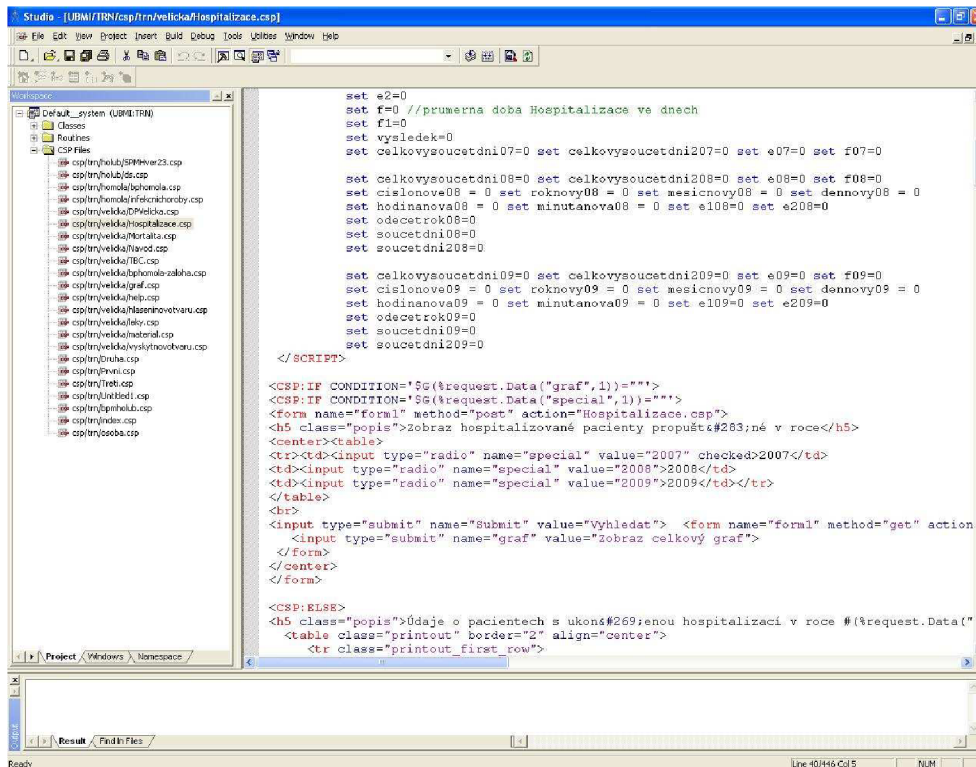
Obr. 10. Menu Caché Cube.

- Getting Started* – slouží k získání základních informací převážně pro nové uživatele
- Start Caché [CACHE]/Stop Caché* – tyto položky slouží pro spuštění/zastavení činnosti databáze Caché
- Studio* – viz. kapitola 5.1.2. Caché Studio
- Terminal* – umožňuje v prostředí terminálové emulace přihlášení v příkazovém řádku k lokálnímu, či vzdálenému systému Caché
- Explorer* – je určen pro správu databáze a jmenných prostor spolu s jimi asociovanými rutinami, třídami a globály
- SQL Manager* - viz. kapitola 5.1.3. SQL Manager
- Control Panel* – umožňuje administrovat spuštěné systémy Caché a jejich procesy
- Configuration Manager* – slouží k nastavení systémové a síťové volby Caché
- Documentation* – zprostředkovává přístup k on-line dokumentaci a výukovým lekcím Caché
- Remote System Access* – vyvoláním utilit z této položky můžeme přistupovat k jinému než přednastavenému serveru.
- Preferred Server* – pomocí utilit v této záložce nastavíme preferovaný server Caché, z kterého chceme získávat data
- About...* - zobrazení copyrightu a informací o verzi Caché
- Exit* – zavře „kostku“ Caché

5.1.2. Caché studio

Tento nástroj je jedním z nejdůležitějších nástrojů pro správu databáze a tvorbu webových aplikací. Je to editor s grafickým rozhraním pro vytváření tříd, stránek CSP a rutin pomocí jazyků Caché ObjectScript a Caché Basic. Ukázka okna tohoto prostředí je na Obr.11.

Při tvorbě této diplomové práce bylo Caché Studio využito jako editor CSP stránek, určených pro tvorbu internetových stránek.



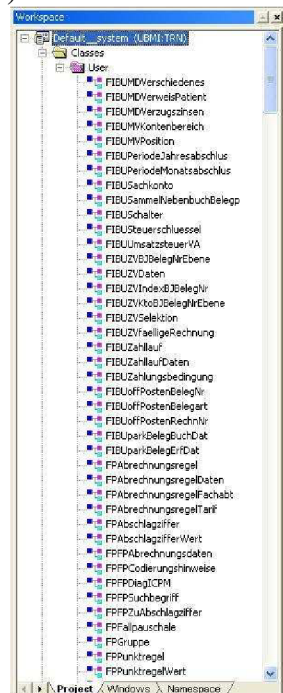
Obr. 11. Ukázka prostředí Caché Studio

V záložce *Classes* jsou obsaženy všechny třídy v příslušné databázi. Jedná se vlastně o data rozdělená do jednotlivých tabulek a podrobněji do jednotlivých sloupců. Nicméně pro práci a orientování se v jednotlivých tabulkách byl v této diplomové práci využit SQL Manager a program MS Excel, viz. dále. Ukázka přehledu jednotlivých tříd v záložce *Classes* je na Obr.12 a).

a)



b)



Obr. 12. a) Záložka Classes. b) Záložka CSP Files.

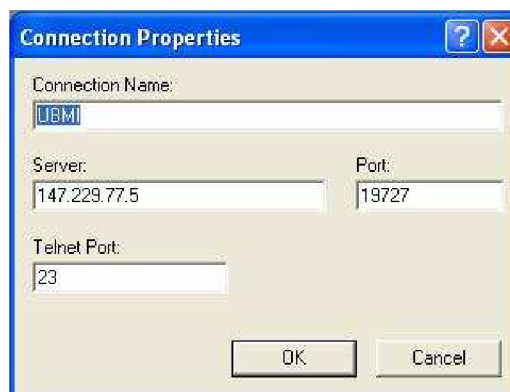
Jelikož CSP (Caché Server Pages) technologií je vytvořena praktická část této diplomové práce, budeme se popisu CSP věnovat v samostatné kapitole. Nyní si jen uvedme, že v Caché Studiu je třeba vytvořit CSP soubory. Toho dosáhneme příkazem z nabídky *File*→*New*→*CSP File*→*Caché server Page*. Do těchto souborů s příponou *csp* budeme zapisovat zdrojový kód pro naši webovou aplikaci. Na Obr.12 b) je ukázka přehledu souborů CSP uložených na příslušném serveru využívaném při tvorbě praktické části této diplomové práce.

V Caché studiu také nastavujeme přístupová data pro spojení se serverem, se kterým chceme komunikovat. Nastavení se provádí v záložce *File*→*Change Namespace*. Zobrazí se nám rámeček *Connection Manager* (Obr.13 a)). Pokud chceme vytvořit nové připojení zmáčkneme tlačítko „New...“. Tím se vyvolá rámeček *Connection Properties*, v němž bude předvolen pouze obecný port pro připojení k serveru Caché: 1972, a Telnet Port: 23. Ostatní položky budou prázdné. Server využívaný při tvorbě této diplomové práce je server Caché spravovaný VUT v Brně, konkrétně ústavem Biomedicínského inženýrství. Údaje potřebné pro připojení k tomuto serveru (přímo vepsané do položek *Connection Properties*) jsou uvedeny na Obr.13 b). V *Connection Manageru* je třeba zvolit v části „Namespace“ položku TRN. TRN je adresář uložený na školním databázovém serveru Caché, ve kterém jsou uloženy CSP soubory. Rovněž je nutno vyplnit uživatelské jméno a heslo, viz.Obr.13 a).

a)



b)

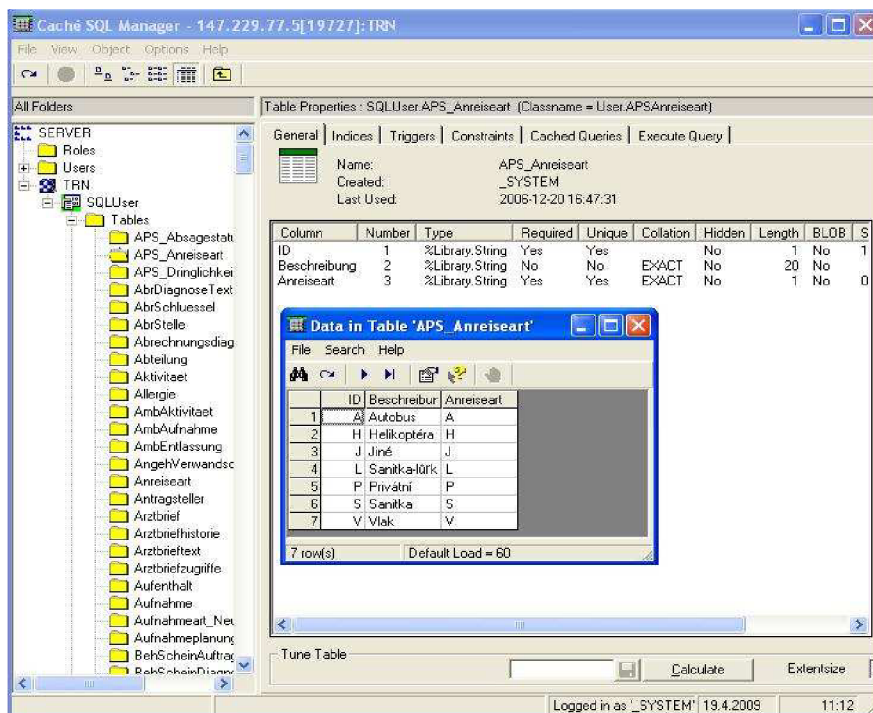


Obr. 13. a) Connection Manager b) Connection Properties.

5.1.3. SQL Manager

Tento nástroj slouží ke kompletní správě přístupu k datům obsaženým v databázi. Pomocí SQL Manageru můžeme prohlížet a editovat jednotlivé tabulky. Je vhodný pro orientaci v datech a tabulkách obsažených v databázi. Umožňuje získat přehled o tom, která data jsou obsažena v kterých tabulkách, jak jsou tabulky vzájemně provázány apod. Kromě tabulek zobrazuje rovněž přehled tak zvaných *Views*, což jsou tabulky sestavené z dat z různých tabulek. Pomocí nástroje *Execute Query* můžeme testovat naše SQL dotazy a přímo vidíme výsledek. Tímto nástrojem tedy můžeme „odladovat“ naše SQL dotazy, dokud

nebudeme s výsledkem spokojeni a poté je vložíme do stránky CSP. Ukázka prostředí SQL Manageru je na Obr.14.



Obr. 14. SQL Manager.

5.2. SQL

Jazyk SQL (Structured Query Language) byl vytvořen pro práci s daty a jejich správu v relačních databázových systémech. Zjednodušeně řečeno zprostředkovává komunikaci mezi počítačem uživatele (klientem) a databázovým serverem. Jedná se o rozšířený jazyk využívaný pro tvorbu a údržbu celé databáze.

První verzi SQL vyvinula firma IBM na začátku 70.let minulého století pod názvem SEQUEL. Tento jazyk byl standardizován v roce 1986 normami ANSI (American National Standards Institute) jako SQL-86, později také normami ISO. Poslední standard je z roku 2008 a oproti předchozímu byl doplněn o některé příkazy, např. ORDER BY, atd.

I když by se zdálo, že databáze využívající jazyka SQL by vzhledem k vydaným standardům měly být vzájemně kompatibilní, není to tak úplně pravda. Většina databází od různých výrobců totiž vyhovuje normám s určitými výjimkami, takže některé části norem nejsou implementovány. Navíc jsou v jednotlivých databázových systémech přidána vlastní rozšíření ke standardním prvkům. Tak dochází k tomu, že jednotlivé databáze (od různých výrobců) mají svá specifika. Vydaná rozšíření pro určitý databázový systém tedy nejsou kompatibilní s jinými databázovými systémy.

Původně byl jazyk SQL koncipován čistě jako dotazovací jazyk, ale postupem času byl rozšířen o další funkce umožňující např. vytváření tabulek, vkládání a změnu dat, atd.

Dle [11] je základ SQL tvořen jazyky DDL (Data Definition Language) a DML (Data Manipulation Language). Jelikož v praktické části této diplomové práce jsou použity příkazy z těchto jazyků, rozebereme si je podrobněji.

DDL

Instrukce jazyka DDL definují strukturu databází, jako řádky, sloupce, tabulky, indexy, apod. Na ukázkou si uvedeme některé typické příkazy:

CREATE – vytváří nový objekt. Nejznámější použitím tohoto příkazu je *CREATE TABLE*.

DROP - maže existující objekt. Nejrozšířenější použitím tohoto příkazu je *DROP TABLE*.

RENAME TABLE – přejmenuje tabulku.

SHOW TABLES – vypíše názvy všech tabulek v konkrétní databázi.

SHOW COLUMNS – vypíše názvy všech sloupců v konkrétní tabulce.

CREATE TEMPORARY TABLE – vytvoří dočasnou tabulku, která existuje jen během připojení. Po ukončení připojení tabulka přestává existovat.

DML

Tento jazyk je využíván pro získání, vkládání, mazání a obnovení dat ve vytvořené databázi. Opět si uvedeme několik příkladů instrukcí:

SELECT – slouží pro získání dat z databáze/tabulky.

INSERT – slouží pro vkládání dat do databáze/tabulky.

UPDATE – slouží pro aktualizaci dat v databázi/tabulce.

DELETE - odstraní data z tabulky po řádcích.

TRUNCATE TABLE - odstraní všechny záznamy z tabulky. V případě, že chceme odstranit všechna data z tabulky, tak je použití *TRUNCATE TABLE* rychlejší než použití *DELETE*.

INDEX – jedná se o pomocnou datovou strukturu, která nám udává umístění dat v tabulce podle jejich hodnoty. Index můžeme vytvořit zároveň s vytvořením tabulky, např. *CREATE TABLE tabulka (index(sloupec1))*. O vytvořený index se „stará“ databáze.

5.2.1. Využití SQL v diplomové práci

Nejprve si podrobně popíšeme příkaz *SELECT* využitý v praktické části této práce. Příkaz *SELECT* slouží především k výběru všech nebo některých sloupců z tabulky, omezení vybraných dat podle požadavků (filtrování dat), seřazení dat, seskupení dat, spojování tabulek a omezení počtu vrácených dat na předem daný počet. Toho dosahuje pomocí klauzulí, které si popíšeme vzápětí. Příkaz *SELECT* vybírá data z tabulky po sloupcích. Pokud chceme vybrat všechny sloupce, vepíšeme za příkaz *SELECT* zástupný znak „*“. Pokud chceme vybrat jen některý sloupec, napíšeme místo „*“ název tohoto sloupce. Název zdroje, ze kterého se data získávají, se uvádí za klauzuli *FROM*. Nejčastějším zdrojem dat jsou tabulky. Uvedme si příklad:

```
SELECT DGICD10 FROM SQLUser.VDiagnoza
```

Slovně by se výše uvedený příkaz dal popsat jako: Vyber data ze sloupce DGICD10 z tabulky VDiagnoza. Zápis SQLUser udává, kde se tabulka nachází na serveru.

Nyní si popíšeme další klauzule:

WHERE

Klauzule *WHERE* slouží k filtrování dat. Pokud chceme například vybrat ze sloupce jen data splňující nějakou podmínku může zápis vypadat následovně:

```
SELECT *  
From SQLUser.VPacient where PAPACIENT=2
```

Tento zápis říká: Pokud je ve sloupci PAPACIENT hodnota 2, vyber všechna data na odpovídajícím řádku tabulky VPacient.

ORDER BY

ORDER BY zajistí seřazení dat podle nějakého výrazu. Nejčastěji podle hodnoty určitých polí ve sloupci. Řadit můžeme podle čísel i podle abecedy.

```
SELECT *  
From SQLUser.VPacient Order by PAPACIENT
```

Dle výše uvedeného zápisu budou data z tabulky VPacient při výpisu srovnána dle hodnot ve sloupci PAPACIENT (ať už se jedná o hodnoty číselné nebo písmena).

JOIN

Spojení tabulek se nejčastěji provádí pomocí klauzule *JOIN*. Spojení se obecně dělí na vnitřní a vnější. O vnitřním nám stačí vědět, že jsou zobrazena pouze data, která si v obou tabulkách přesně odpovídají, a že se provádí klausulí *INNER JOIN*.

Vnější spojení je využito přímo v diplomové práci. Provádí se pomocí syntaxe *LEFT JOIN ... ON*. Atribut *LEFT* má za následek, že se načtou všechna data z první tabulky a z druhé tabulky se načtou pouze data související. Pomocí klíčového slova *ON* stanovíme podmínku, podle níž se mají data z tabulek sloučit. Uveďme si opět příklad:

```
SELECT *  
FROM SQLUser.tabulka1  
LEFT JOIN SQLUser.tabulka2 ON dgicd10=ICICD10
```

Obě tabulky jsou propojeny tak, že ve sloupcích dgicd10 a ICICD10 jsou shodné hodnoty, konkrétně označení diagnóz.

Vnější spojení mohou být „levá“ (*LEFT JOIN*) nebo „pravá“ (*RIGHT JOIN*). Funkčně se ničím neliší. Možnosti „pravého“ nebo „levého“ spojení se využívají jen v některých případech, hlavně při spojení více tabulek. Výše uvedený příklad by tedy mohl vypadat takto:

```

SELECT *
FROM SQLUser.tabulka1
RIGHT JOIN SQLUser.tabulka2 ON dgicd10=ICICD10

```

a významově jsou v tomto případě oba příklady shodné.

Na závěr kapitoly si ukážeme zápis jednoho SQL dotazu použitého přímo v praktické části diplomové práce:

```

SELECT SUBSTRING(DGDIAGNOZA,1,3) AS DGZ, DGSkupinaDiagnoz, DGPoprve,
      DGICD10, ICICD10, ICPopis, Aenderungsdatum, geaendert_durch,
      SUBSTRING(Aenderungsdatum,1,4) AS rok,
      SUBSTRING(Aenderungsdatum,5,2) AS mesic,
      SUBSTRING(Aenderungsdatum,7,2) AS dden,
      SUBSTRING(Aenderungsdatum,1,8) AS datum
FROM SQLUser.VDiagnoza
LEFT JOIN SQLUser.VICD10KS ON dgicd10=ICICD10
LEFT JOIN SQLUser.Diagnosegruppe ON DGSkupinaDiagnoz=Diagnosegruppe

```

5.3. Objektové databáze

Objektová technologie se pokouší napodobovat způsob, jakým lidé opravdu přemýšlejí o informacích a jejich použití. Na rozdíl od relačních tabulek spojují objekty dohromady data i kód.

V objektové technologii je složitost dat obsažena v objektu a přístup k datům je realizován prostřednictvím jednoduchého rozhraní. Relační technologie sice rovněž používá jednoduché konzistentní rozhraní, ale protože nijak neřeší složitost skutečných dat, je za řešení problémů se složitostí dat vždy odpovědný uživatel nebo programátor.

Na objekt můžeme nahlížet jako na balíček, který obsahuje zároveň hodnoty dat tohoto objektu (vlastnosti) a kopii veškerého příslušného kódu (metody). Komunikace mezi metodami objektu je zajišťována předáváním zpráv. Objekty téže třídy mohou sdílet společné kopie kódu a tím snižovat nároky na paměť. V systému Caché také volání metod obvykle vede k volání vhodné funkce, ne k neustálému předávání zpráv.

5.4. ODBC

ODBC (*Open Database Connectivity*) je strategické rozhraní firmy Microsoft pro datový přístup v heterogenním prostředí pro relační a nerelační databázové správní systémy. Pro přístup využívají ODBC drivery jazyku SQL.

ODBC ovladače jsou spjaty s členěním aplikace do dvou částí, oddělených ODBC rozhraním. Vlastní ODBC ovladač přejímá od aplikace příkazy pro zpracování dat a vykonává tyto příkazy v datové základně. Výhodou tohoto řešení je možnost jednoduché záměny ODBC ovladače za jiný. ODBC otevírá soubory vždy ve sdíleném režimu.

Kromě ODBC ovladačů existují tzv. nativní rozhraní. Jedná se o ovladač síťového rozhraní specifický pro daný databázový server. ODBC rozhraní je univerzální, se kterým by se dle norem mělo pracovat stejně pro libovolný SQL server i pro libovolnou ODBC-schopnou aplikaci.

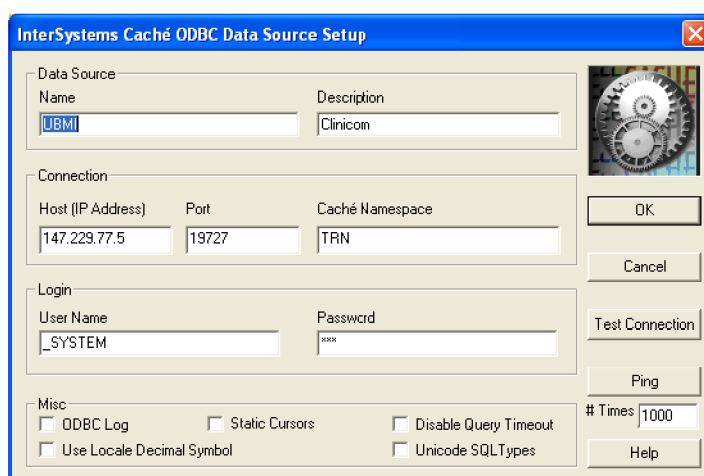
Jednou z možností přístupu k datům v databázi je přímý přístup pomocí SQL dotazů, viz. kapitola 5.2. SQL. Další možností je právě přístup přes rozhraní ODBC. Obě možnosti byly v práci testovány. Přístup k datům přes ODBC a jejich následné zpracování v MS Excel bylo v práci využito jen pro kontrolu výsledků vypočítaných v souborech CSP. Tato kontrola byla zvolena na základě přehlednosti a snadné ovladatelnosti programu MS Excel. Popíšme si tedy v krátkosti realizaci přístupu k databázovým datům pomocí ODBC a MS Excel.

5.4.1. Instalace softwaru

Pro komunikaci byl vybrán softwarový balík: cache-5.0.20.6305-win. Při instalaci si můžeme vybrat, jaké části nainstalujeme, ale v každém případě musíme nainstalovat ODBC driver. Po instalaci tento driver přidáme do systémových zdrojů dat následujícím postupem:

Tento počítač→*Ovládací panely*→*Nástroje pro správu*→*Datové zdroje ODBC*
→záložka *Systémové DSN*→*Přidat*

Z nabízených možností vybereme ovladač InterSystems ODBC a nastavíme podle Obr.15. Úspěšnost nastavení ověříme stiskem tlačítka *Test Connection*. Data můžeme nejlépe načíst pomocí programu MS Excel.



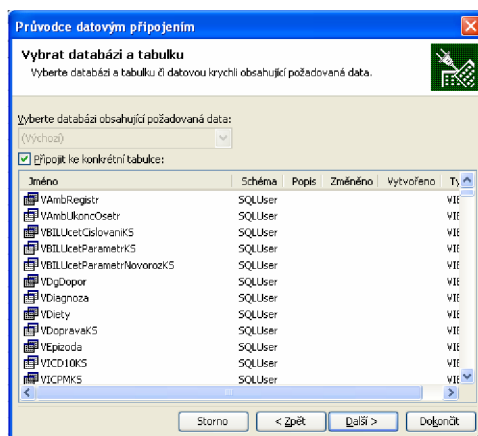
Obr. 15. Nastavení ODBC driveru

5.4.2. Načtení dat pomocí MS Excel

V programu MS Excel data načítáme pomocí následujícího návodu:

Data→*Importovat externí data*→*Importovat data*→ *Připojit k novému zdroji dat*
→*Název zdroje dat ODBC*→*UBMI*

Otevře se nám přehled tabulek, které můžeme importovat do programu MS Excel, viz. Obr.16. Každá tabulka obsahuje různá data seřazena do sloupců podle charakteru.



Obr. 16. Přehled tabulek importovatelných z NIS Clinicom

6. CSP

CSP (Caché Server Pages) je základní technologií pro dynamické vytváření obsahu na základě získávání informací z databáze a umožňuje tyto získané informace prezentovat ve webovém prohlížeči. Prostředky technologie CSP (Caché Server Pages) jsou umístěny na datovém serveru Caché, kde také pracují a využívají výhody rychlé vnitřní komunikace procesu. Připojení k webovému serveru se uskutečňuje pomocí rychlých standardních rozhraní API. Přístup na web je kompatibilní se všemi hlavními webovými servery. Centralizovaná logika aplikace umožňuje rychlé a snadné změny aplikace.

Obecně lze přistupovat k vývoji stránek dvěma způsoby:

1. Tvorba stránky jako souboru HTML nebo XML s vnořenými speciálními značkami CSP a dalšími prvky jazyka CSP. Tato metoda přístupu se nazývá *Značkový vývoj*.
2. Soubor je tvořen jako třída CSP, která je podtřídou `%CSP.page`. Výkonná logika a požadované značky HTML a XML jsou součástí metody `OnPage()`. Tento přístup je označován jako *Kódový vývoj*.

Jazyk Caché je vystavěn na starším programovacím jazyku MUMPS (Massachusetts General Hospital Utility Multi-Programming System). Tento databázový programovací jazyk byl vytvořen v 60. letech minulého století přímo pro využití v nemocnicích. Některé funkce z tohoto jazyka jsou přímo převedeny do novějšího jazyka Caché beze změny zápisu a významu. Obecně jsou tyto předdefinované funkce funkcemi Caché ObjectScriptu a začínají vždy znakem „\$“. Patří mezi ně např.:

\$Extract – rozdělí proměnnou na předem nadefinované části

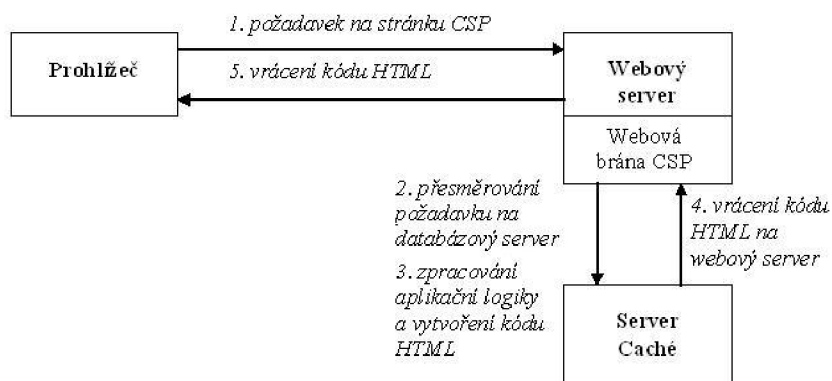
\$Horolog – obsahuje datum a čas jako dvě počítadla. První počítadlo určuje pořadí dne od 31. prosince 1840. Druhé počítadlo určuje počet sekund od poslední půlnoci.

\$ZNSpace – obsahuje název aktuálního jmenného prostoru.

\$ZHorolog – obsahuje čas v sekundách a milisekundách od posledního spuštění Caché.

a mnoho dalších. Jednotlivé argumenty funkce se uzavírají do kulatých závorek a oddělují čárkami. Např.: *\$Extract(proměnná,7,8)*.

Aby mohl webový server a server Caché spolupracovat, musí být na webovém serveru nainstalováno rozhraní Caché. Celá komunikace je nastíněna na Obr.17.



Obr. 17. Způsob komunikace webového a databázového serveru pro zpracování stránek CSP.

CSP server nejprve zpracuje skripty jemu určené a uživateli je poté přenesen jen „přeložený“ kód s již zpracovanými funkcemi. Výsledek je zobrazen ve struktuře jazyka HTML.

CSP kromě výše uvedených předdefinovaných funkcí nabízí další objekty z Caché ObjectScriptu. Jsou jimi například balíčky obsahující systémové třídy Caché. Vlastnosti těchto balíčků začínají vždy znakem „%“. Uveďme si objekty využívané při tvorbě webové aplikace této diplomové práce:

%REQUEST - tento objekt zpřístupňuje určitá data serverovým aplikacím

%SESSION – pomocí tohoto objektu můžeme sledovat stav spojení včetně uživatele, prohlížeče, webového serveru a databáze. Umožňuje, aby jedna množina dat byla přístupná po celou dobu spojení serveru s klientem. V naší webové aplikaci je využita pro přenos informace o autentizaci uživatele mezi jednotlivými stránkami, viz. kapitola 4.5.

Popišme si nyní jen přehledově, jak vlastně webová aplikace vytvořená v rámci této diplomové práce v CSP funguje. Vlastní zdrojový kód je tvořen z několika částí:

6.1. HTML

Jak bylo uvedeno dříve CSP umožňuje využití známého jazyka HTML. Podrobněji tento jazyk popisovat nebudeme, uveďme si jen na ukázkou některé používané tagy: `<table>`, `<select>`, `<center>`, `<h3>`, a mnoho dalších.

6.2. Větvení kódu (podmínky)

Základem všech výkonnějších webových aplikací, ale obecně všech „inteligentních“ programů, jsou podmínky. Jedná se zpravidla o příkaz *IF*. V CSP konkrétně o sérii příkazů `<CSP:IF>`, `<CSP:ELSEIF>`, `<CSP:ELSE>` a `</CSP:IF>`. V „těle“ funkce `<CSP:IF>` a `<CSP:ELSEIF>` nastavíme určitou podmínku. Příkaz *IF* a *ELSEIF* testuje, zda je tato podmínka splněna. Tímto způsobem se zdrojový kód větví vždy podle toho, zda je podmínka splněna, či ne. Pokud je podmínka splněna vykoná se zdrojový kód vepsaný bezprostředně za touto podmínkou `<CSP:IF>`. Pokud podmínka splněna není, pokračuje se na další podmínku `<CSP:ELSEIF>`, případně se vykoná kód zapsaný za funkcí `<CSP:ELSE>`. Blok obsahu jedné podmínky `<CSP:IF>` ukončíme značkou `</CSP:IF>`.

Jednotlivé bloky podmínky `<CSP:IF>` můžeme vkládat do sebe. Nesmíme však ztratit přehled o jednotlivých blocích, jejich pořadí (tzn. v jakém pořadí se budou podmínky testovat) a na závěr bloku musíme vždy umístit značku `</CSP:IF>`. Využití bloků podmínek je ukázáno ve zdrojovém kódu webové aplikace v příloze č.1.

6.3. Cykly

Dalším příkazem, který neodmyslitelně patří k programování, je příkaz pro provádění cyklů. Velmi často, a v této práci výhradně, využívaný je příkaz WHILE. V CSP má konkrétní podobu `<CSP:WHILE>`. Obsah cyklu je vykonáván, dokud neskončí podmínka. Pokud např. procházíme data v databázi, je logická podmínka cyklu: „procházej, pokud jsou na dalším řádku tabulky data.“ V CSP by tento zápis mohl vypadat např. takto:

```
<CSP:WHILE CONDITION=tabulka.Next ()>
```

Cyklus while je ukončen značkou `</CSP:WHILE>`. Jednotlivé cykly můžeme rovněž vkládat „do sebe“.

6.4. Skripty

Samotný skript je uvozen značkami `<SCRIPT>` a `</SCRIPT>`. Pomocí tohoto příkazu můžeme v CSP využívat tyto skriptovací jazyky: Caché ObjectScript, Caché SQL, Caché Basic, VBScript a JavaScript. Jazyk zvolíme tak, že do příkazu `<SCRIPT>` vložíme jeho název, např.:

```
<SCRIPT LANGUAGE="SQL" NAME="pacient">
```

Skripty můžeme pojmenovávat. Můžeme rovněž zvolit, zda je skript určen pro server nebo pro compiler (překladač). Toto provedeme pomocí klíčového slova `RUNAT="SERVER"` nebo `RUNAT="COMPILER"`.

6.5. Databázové dotazy

Pomocí databázových dotazů můžeme pracovat s daty uloženými v databázi. CSP umožňuje dvojí přístup k tvorbě databázových dotazů. Jednak pomocí předdefinovaných databázových dotazů vložených do značek „SQL:QUERY“, nebo pomocí výše popsaných skriptů. Varianta se skripty je využita při tvorbě této diplomové práce. Oba přístupy však využívají jazyka SQL, viz. kapitola 5.2.

6.6. Práce s proměnnými a daty

Ve všech programech potřebujeme naše výpočty ukládat pro další použití do proměnných. V CSP toto můžeme provádět pomocí zápisu:

```
<SCRIPT LANGUAGE="CACHE" RUNAT="SERVER">  
  set a = 6  
</ SCRIPT >
```

Blok `SCRIPT` jsme si popsali dříve. Hodnotu vložíme do proměnné pomocí příkazu `set`. Do proměnné můžeme ukládat libovolná data.

Pokud chceme proměnnou vypsat na straně klienta, použijeme uvozovacího znaku „#“, např.: #(a)#. Hodnota proměnné bude při generování nahrazena jejím obsahem.

Stejným způsobem vypisujeme i data získaná přímo z databáze. Musíme je ale v závorce doplnit o data, která specifikují konkrétní tabulku a sloupec např.:

```
#(pacient.Data("PAPrijmeni"))#
```

V tomto případě se jedná o výpis dat ze sloupce PAPrijmeni tabulky specifikované ve skriptu s názvem „pacient“.

Pomocí příkazu *set* můžeme provádět některé matematické operace. Této vlastnosti příkazu *set* bylo využito v této diplomové práci pro provádění výpočtů. Uveďme si pro názornost konkrétní příklad ze zdrojového kódu vytvořeného v rámci diplomové práce:

```
set soucetdni= (((hodnoty.Data("mesic"))*30) +  
(hodnoty.Data("den"))) - (mesicnovy*30 + dennovy)
```

Z příkladu je patrné, že výpočet ukládáme do proměnné. Hodnoty potřebné pro výpočet mohou být uloženy v nějaké proměnné, nebo mohou být brány přímo z databázové tabulky.

7. Popis vytvořené webové aplikace

V předchozích kapitolách jsme si popsali teoreticky a částečně i prakticky struktury použité pro tvorbu vlastní webové aplikace. V této kapitole se zaměříme na popis aplikace z hlediska její funkčnosti.

Celá aplikace je realizována na školním serveru, na němž je nainstalována plná verze Intersystems Caché. Pro tvorbu aplikace by bylo možno využít i jiný server a na databázový školní server se pouze odkazovat. Tím by ale zdrojový kód nebyl psán v CSP, a proto by tato varianta nesplňovala zadání této diplomové práce. Data byla brána z databáze nemocničního informačního systému Clinicom, který je rovněž uložen, včetně všech patientských dat, na školním serveru. Pro výpočet statistických údajů bylo třeba vyhledat „správné“ tabulky, v nichž se nacházejí potřebná data. Vyhledávání těchto tabulek bylo poměrně obtížné, poněvadž ne vždy se podle názvu dalo zjistit, co je obsahem tabulky. Tabulky byly prohledávány a otevírány pomocí SQL Manageru a programu MS Excel. Také bylo třeba zjistit, jak jsou některé tabulky vzájemně svázány. Konkrétní využití tabulky si popíšeme v následujících kapitolách.

Webovou aplikaci spustíme na adrese: <http://147.229.77.5:19727/csp/trn/velicka/DPVelicka.csp>. Po přihlášení, popsaném v kapitole 4.5. a zobrazeném na Obr.8, se nám zobrazí úvodní stránka, viz. Obr.18.

Statistická data v nemocničním informačním systému Clinicom

Slovo autora
Tyto stránky vznikly v rámci diplomové práce na VUT v Brně, FEKT, 2009.
V dalších odstavcích jsou popsány jednotlivé funkce těchto stránek, určené pro výpočet některých statistických údajů z dat získaných ze školní patientské databáze. Funkce jsou děleny do tří hlavních oblastí: Hospitalizace, Povinná hlášení a Spotřeba Materiálu. Jednotlivé funkce se spouštějí stiskem příslušného tlačítka v menu stránky.

Hospitalizace
Průměrná doba
Tato funkce vypíše seznam všech pacientů propuštěných z nemocničního ošetření ve zvoleném roce, doplněný o některé údaje. Na konci seznamu (ve spodní části stránky) je uvedena průměrná doba ošetření.

Výpočet mortality
Po zvolení roku se zobrazí seznam zesnulých pacientů s některými základními údaji. Pod tabulkou pacientů je vypsán jejich počet pro daný rok. Funkce je doplněna o možnost zobrazení grafu.

Povinná hlášení
Výskyt novotvarů
V rolovací nabídce vybereme požadovanou diagnózu. Poté můžeme vybrat konkrétní rok a měsíc (musí být vybráno současně), nebo zaškrtnutím políčka "všechny" zvolíme výpis diagnózy pro všechna období. Po stisku tlačítka "Vyhledat" se zobrazí tabulka s údaji o pacientech, u kterých byla nalezena zvolená diagnóza pro zvolené období.

Hlášení novotvarů
Označením některého z tlačítek "poslední rok", "poslední měsíc" nebo "poslední den", zvolíme období, pro které chceme vypsat všechny diagnózy popisující výskyt novotvaru. Po stisku tlačítka "Vyhledat" se zobrazí tabulka s údaji o pacientech, u kterých byla nalezena zvolená diagnóza pro zvolené období. V tabulce je na každém řádku vpravo na konci tlačítko "Formulář". Stiskem tohoto tlačítka zobrazíme ukázkou vyplněného hlášení o nálezů nového novotvaru. Toto hlášení je doplněno samostatnou nápovědou.

Vyskyt TBC
Funkce pracuje shodně s funkcí "Výskyt novotvarů". Vyhledává však pacienty u nichž byl prokázán výskyt TBC.

Spotřeba materiálu
Spotřeba léků
Z nabídky vybereme požadovaný lék a potvrdíme tlačítkem "Vyhledat". Zobrazí se počet spotřebovaných kusů zvoleného léku.

Ostatní materiál
Funkce pracuje shodně s funkcí "Spotřeba léků". Vyhledává ale zdravotnický materiál vyjma léků.

Tyto stránky vytvořil Tomáš Velická v rámci Diplomové práce, VUT v Brně, 2009

Obr. 18. Úvodní stránka webové aplikace

V „hlavním“ rámečku úvodní stránky je zobrazen text, který ozřejmuje, k čemu webová aplikace slouží. Jedná se v podstatě o návod, který vysvětluje a ulehčuje uživateli ovládání jednotlivých funkcí webové aplikace. Funkcí stránek vytvořených v rámci této diplomové práce je výpočet některých statistických údajů z databáze patientských dat NIS Clinicom. Jednotlivé funkce, sloužící pro výpočet těchto statistických údajů, se spouštějí stiskem příslušného tlačítka v menu stránky, které je ukázáno v levé části Obr.18. Funkce jsou rozděleny do 3 základních skupin: Hospitalizace, Povinná hlášení a Spotřeba materiálu. Rovněž tlačítka na úvodní stránce, Obr.18, jsou nazvána stejně. Po stisknutí příslušného tlačítka, se rozbalí menu s jednotlivými funkcemi, patřícími do této skupiny, viz. Obr.19.



Obr. 19. Menu webové aplikace.

Výběr statistických údajů (např. délka hospitalizace) a jiných funkcí (např. povinná hlášení) nacházejících se v naší webové aplikaci byl inspirován [6] a [24]. Celá stránka je koncipována tak, že do hlavní stránky, jejíž zdrojový kód je uložen v souboru DPVelicika.csp, se načítají další stránky. Tyto stránky provádějí výpočty a vyhledávání vyžádaných informací. Zdrojové kódy těchto stránek jsou ukázány v příloze na konci textu diplomové práce.

Popišme si nyní jednotlivé dílčí funkce webové aplikace, které se spouštějí příslušnými tlačítky ukázanými na Obr.19.

7.1. Hospitalizace - Průměrná doba

V této části využíváme data z tabulek VPacient a VNZIS. Ukázkou těchto tabulek dokládají tabulky 2 a 3. Tabulka VPacient není zobrazena celá, jsou vybrány jen sloupce, které byly pro tvorbu webové aplikace podstatné. V tabulkách 2 a 3 jsou pro názornost ukázána data jednoho pacienta.

Tabulka 2.: Ukázka tabulky VPacient

PA Mesto	PA Titul Akad	PA Umrți Datum	PA Epizoda AktCislo	PA Jmeno	PA Jmeno Rodne	PA Narodeni Misto	PA Narodeni Datum
		20080311	1	Jozef	Zisko	Handlova	19331025

PA Narodeni Den	PA Narodeni Mesic	PA Narodeni Rok	PA PACIENT	PA Acient Cislo	PA Pohlavi	PA Psc	PA Prijmeni	PANarodnost
25	10	1933	153	627	2		Zisko	CZ

PA StavRodinny	PA Telefon	PAUlice	PAZamestnani	PAHospitalizovan	PACisloRodne
S			Ucitel	1	331025113

Tabulka 3.: Ukázka tabulky VNZIS

NZ Epizoda	NZaktivita Datum	NZPrijeti Specializace	NZPrijeti Doporucil	NZ Prijeti Duvod	NZ Prijeti Druh	NZ Zač Prizn DatumCas	NZ Pooperacni Komplikace
153	1	200710030001	INTS	0	2	1	197010070001
153	1	200710031811	GYPS	6	2	1	200710030001

NZ Nahrada	NZ Potreba DalsiPece	NZ Reoperace	NZ Druh Operace	NZNemocnicni Nakaza	NZ Propuštění Druh	NZCZNZIS
						153 1 200710030001
	3			5	3	153 1 200710031811

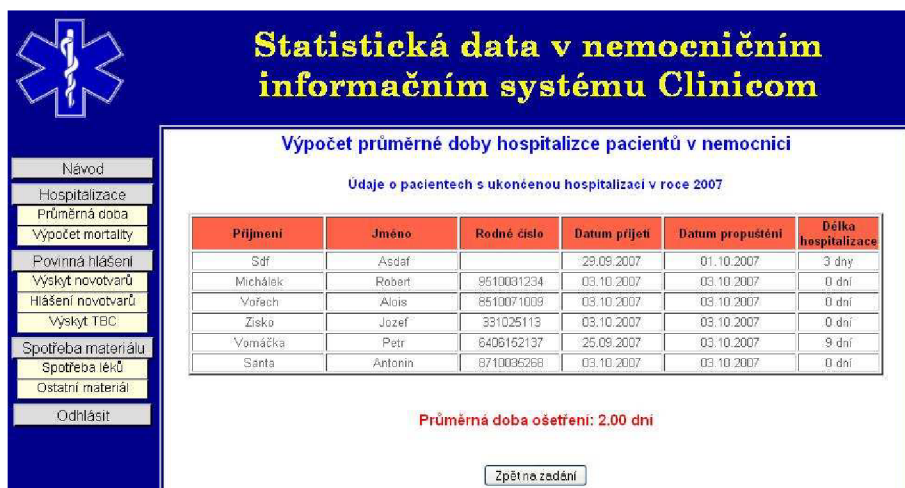
Tabulky jsou propojeny číslem uvedeným v tabulce VPacient ve sloupci „PAPACIENT“ a v tabulce VNZIS ve sloupci „NZEpozoda“. V tabulkách 2 a 3 jsou tato čísla pro názornost zvýrazněna červeně.

Po spuštění funkce „Průměrná doba“ se nám načte do „hlavního“ okna stránky stránka *Hospitalizace.csp*, viz. Obr.20.

Obr. 20. Funkce „Průměrná doba“.

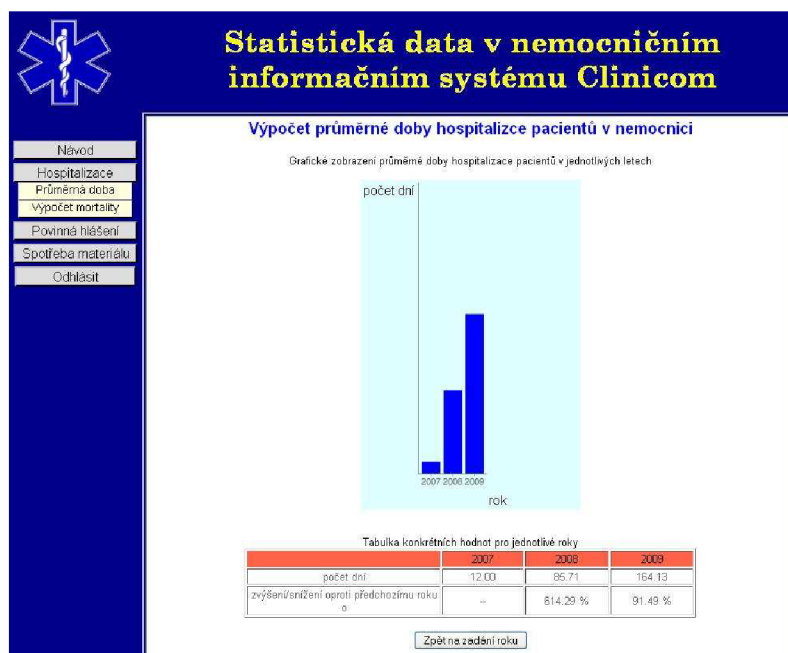
Po výběru příslušného roku a potvrzení tlačítkem „Vyhledat“, se nám zobrazí počet pacientů propuštěných z hospitalizace v nemocnici ve zvoleném roce a tabulka s některými patientskými údaji, jak je doloženo na Obr.21 na následující straně.

Výpočet je založen na porovnávání sousedních řádků v tabulce VNZIS a hledání stejného identifikačního čísla pacienta na dvou, či více po sobě následujících řádcích. To totiž znamená, že se u jednoho pacienta vyskytlo více událostí. V dalším kroku následuje testování, zda je pole ve sloupci „NZPropuštěníDruh“ tabulky VNZIS prázdné, či nikoli. V případě, že není, pacient byl propuštěn. V tomto případě na druhu propuštění nezáleží. Nalezneme pouze čas odpovídající této události. Následuje vyhledání času přijetí daného pacienta. Poté, protože už máme čas přijetí i propuštění, oba časy prostě odečteme. Celou operaci zprůměrnujeme počtem propuštěných pacientů a máme průměrnou dobu ošetření. Zdrojový kód stránky provádějící tento výpočet je zobrazen v příloze č.1. v souboru *Hospitalizace.csp*.



Obr. 21. Výpočet průměrné doby hospitalizace pacientů v nemocnici ve zvoleném roce.

Kromě výpočtu průměrné doby hospitalizace pro vybraný rok, můžeme zvolit zobrazení grafu vypočtených hodnot pro všechny roky obsažené v databázi. Toto provedeme stiskem tlačítka „Zobrazit celkový graf“, viz. Obr.20. Zobrazí se nám příslušný graf doplněný tabulkou s konkrétními údaji průměrné délky hospitalizace v jednotlivých letech a procentuálním vyjádřením změny této hodnoty oproti předchozímu roku (Obr.22).

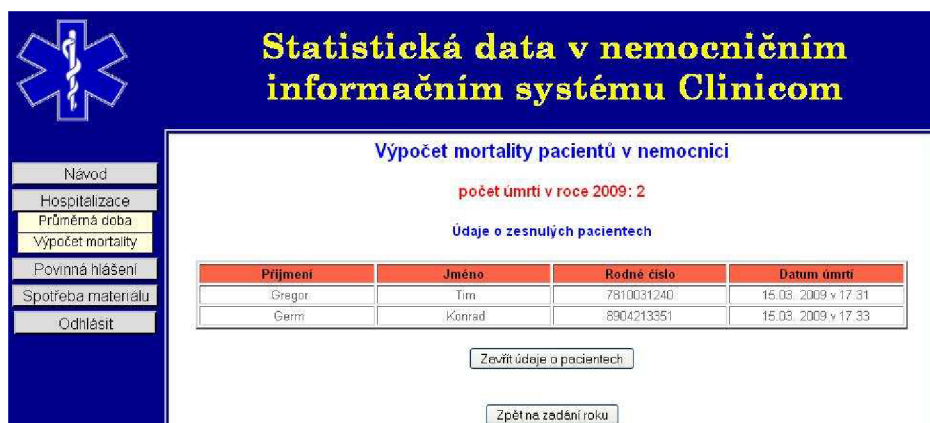


Obr. 22. Graf průměrné doby hospitalizace pacientů v nemocnici.

7.2. Hospitalizace – Výpočet mortality

Tato funkce využívá stejných tabulek jako předchozí funkce, tedy tabulek VPacient a VNZIS, viz. tabulky 2 a 3. Ve sloupci „NZPropuštěníDruh“ tabulky VNZIS testujeme, zda je hodnota v daném poli shodná s hodnotou 7 nebo 8 (což je propuštění z důvodů úmrtí). Podoba výstupních dat je podobná jako u předchozí funkce popsané v kapitole 7.1. na Obr.21 a 22. Funkce se spouští stiskem tlačítka „Výpočet mortality“ a její zdrojový kód je uložen

v souboru *Mortalita.csp*. Jen pro srovnání je na Obr.23 ukázán výsledek výpočtu mortality pacientů pro rok 2009.



Obr. 23. Výpočet mortality pacientů v nemocnici ve zvoleném roce.

7.3. Povinná hlášení – Výskyt novotvarů

Pro výpočet výskytu novotvarů mezi ošetřenými pacienty, byly využity tabulky: VPatient (viz. tabulka 2 v kapitole 7.1.), VDiagnoza (tabulka 4), Diagnosegruppe (tabulka 5) a VICD10KS (tabulka 6). Ukázky těchto tabulek, s údaji o jednom pacientovi, jsou uvedeny v následujících tabulkách:

Tabulka 4.: Ukázka tabulky VDiagnoza

DG Poradi	DG DIAGNOZA	DGSkupina Diagnoz	DG Poprve	DG ICD10	DG Volny Text	DGDg Group	DG Dispenzarizace	DG DispZde
10	153 1 1 10	153 1 1	1	I12.0		I12		
11	153 1 1 11	153 1 1		C71.5		C71		

Tabulka 5.: Ukázka tabulky Diagnosegruppe

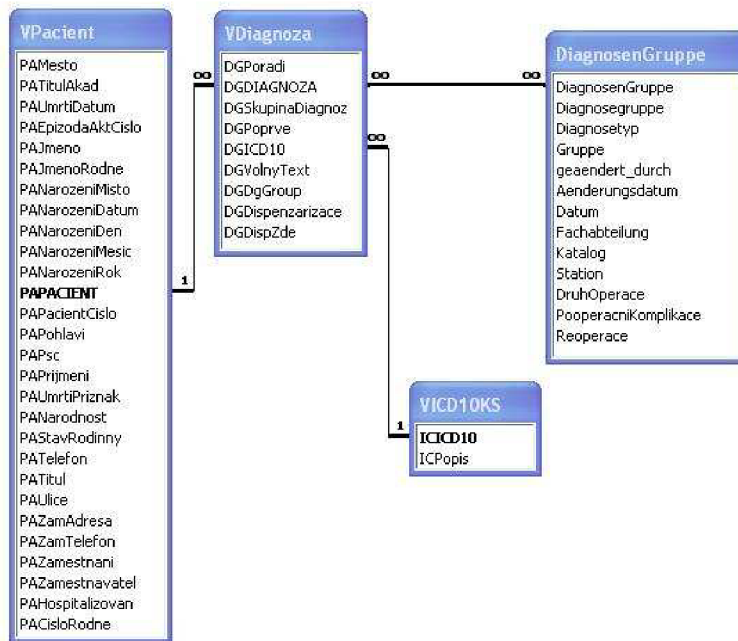
DiagnosenGruppe	Diagnosegruppe	Diagnosetyp	Gruppe	geändert_durch	Aenderungsdatum
153 1	153 1 1	2	1	ST1	200803112321

Datum	Fachabteilung	Katalog	Station	Druh Operace	Pooperační Komplikace	Reoperace
200803112322	GYPS	1	GYNL			

Tabulka 6.: Ukázka tabulky VICD10KS

ICICD10	ICPopis
C71.5	ZN- mozková komora mimo čtvrtou komoru (C71.7)

Barevně jsou v jednotlivých tabulkách zvýrazněny sloupce, kterými jsou tabulky vzájemně propojeny. Toto spojení je více patrné na E-R diagramu na Obr. 24.



Obr. 24. E-R diagram vybraných tabulek z NIS Clinicom.

Samotný výpočet je založen na porovnávání předem zadaného znaku se znaky uvedenými v tabulce VDiagnoza ve sloupci „DGICD10“. Princip celého výpočtu je uveden v příloze č.1 v souboru *vyskytnovotvaru.csp*.

Jak bylo uvedeno v kapitole 3.1., do Národních onkologických registrů se uvádí diagnózy C 00 - C97 a D00 - D76.0. Jelikož nám nešlo o nalezení přesné diagnózy, ale pouze o novotvary jakéhokoli výskytu i povahy, jsou předem zadané znaky pro porovnávání „C“ a „D“.

Spuštěním funkce „Výskyt novotvarů“ se zobrazí výběr hledaných diagnóz a nabídky pro zvolení měsíce a roku, pro který chceme hledat počet pacientů s hledanou diagnózou, viz. Obr.25.



Obr. 25. Funkce „Výskyt novotvarů“.

Po zvolení konkrétní diagnózy určíme časový úsek, pro který chceme výpočet provést. To uděláme buď zvolením konkrétního roku a měsíce, nebo zaškrtnutím políčka „všechny“. Pokud zaškrtneme políčko „všechny“, určíme tím, že výpočet bude prováděn bez ohledu na čas výskytu diagnózy. Dojde tedy k nálezům všech pacientů, u kterých byla stanovena zvolená diagnóza kdykoli v minulosti.

Po potvrzení volby se zobrazí seznam a počet pacientů s hledanou diagnózou v hledaném období, jak je ukázáno na Obr.26.

Příjmení	Jméno	Rodné číslo	Místo narození	Národnost	číslo diagnózy	Popis diagnózy	Datum
Uterý	Petr	8111043215		CZ	C44.7	Jiný ZN- kůže dolní končetiny, včetně kyčle	18.11.2008

Počet pacientů s danou diagnózou: 1

Zpět na zadání

Obr. 26. Výpočet výskytu zadaného typu novotvaru u pacientů ve zvoleném období.

7.4. Povinná hlášení – Hlášení novotvarů

Jak bylo uvedeno v kapitole 3.1., mají pracoviště, která uskuteční nález novotvaru, ze zákona povinnost tento nález hlásit do Národního onkologického registru (NOR). Vyhledávání pacientů s onkologickým nálezem provádí i funkce „Hlášení novotvarů“. Spustíme ji stiskem stejnojmenného tlačítka v menu. Vybereme období, pro které chceme počet pacientů s onkologickým nálezem vypočítat a potvrdíme, Obr.27.

Výpočet výskytu novotvarů

Zobraz všechny diagnózy ve zvoleném období:

poslední rok poslední měsíc poslední den

Vyhledat

Obr. 27. Funkce „Hlášení novotvarů“.

Zobrazí se tabulka s jmény a jinými údaji o pacientech s potvrzeným onkologickým nálezem v zadaném období, jak dokládá Obr.28. na následující straně.

Funkce využívá stejných výpočtů a tabulek jako funkce „Výskyt novotvarů“ popsaná v předchozí kapitole. Rozdíl je ale v tom, že nevyhledáváme konkrétní diagnózu. Zajímají nás všechny onkologické nálezy v daném období. Vypočítáme tedy, kolik nových onkologických nálezů jakéhokoli typu přibýlo v patientské databázi v určitém období.

Obr. 28. Výpočet výskytu všech novotvarů u pacientů ve zvoleném období.

V kapitole 3.2. jsme rozebírali zvláštní pravidla pro hlášení zhoubných novotvarů. Tyto novotvary nestačí nahlásit do NOR, ale příslušná zdravotnická pracoviště musí ještě vyplnit formulář s názvem „Incidence a léčba zhoubného novotvaru“. Tento formulář se skládá ze dvou částí. Názvy těchto částí a pravidla pro vyplňování jsou uvedena v kapitole 3.2. Ve funkci „Hlášení novotvarů“ je možno vyplnit první část hlášení, která je určena pracovišti, které jako první nález zhoubného novotvaru diagnostikovalo. U pacientů s diagnostikovaným zhoubným novotvarem je, ve výsledné tabulce s některými údaji, přidáno tlačítko „Formulář“. Tímto tlačítkem vygenerujeme návrh na formulář, inspirovaný první částí formuláře „Incidence a léčba zhoubného novotvaru“, který je doložen v příloze č.2. Tento formulář je částečně vyplněn údaji získanými ze školní patientské databáze NIS Clinicom, viz. Obr 29.

Obr. 29. Formulář „Incidence zhoubného novotvaru“.

Nevyplněné údaje, které nebyly ve školní databázi obsaženy, musí doplnit uživatel. Po vyplnění formuláře je možno generovat upravenou verzi pro tisk. Ukázka vyplněného a vtištěného formuláře je uvedena v příloze č.3.

Zdrojový kód funkce „Hlášení novotvarů“ je ukázán v příloze č.1 v souboru *hlaseninovotvaru.csp*.

7.5. Povinná hlášení – Výskyt TBC

Tato funkce je v podstatě shodná s funkcí „Výskyt novotvarů“ popsanou v kapitole 7.3. Jediným rozdílem je, že zadané znaky pro porovnávání jsou A15 – A19. Tyto znaky odpovídají infekčním chorobám typu TBC. Podoba výstupních dat odpovídá podobě výstupních dat u funkce „Výskyt novotvarů“, viz. Obr.26. Název souboru obsahujícího zdrojový kód stránky vykonávající funkci „Výskyt TBC“ je uložen v souboru *TBC.csp*.

7.6. Spotřeba materiálu – Spotřeba léků

Tato funkce pracuje s tabulkami NLeistung a ErbrachteNL. Ukázka uspořádání a obsahu těchto tabulek je v tabulkách 7 a 8. V tabulkách 7 a 8 nejsou zobrazeny všechny sloupce tabulek NLeistung a ErbrachteNL, ale jen sloupce vybrané pro ukázkou a potřebné pro výpočet množství spotřebovaného materiálu a léků.

Tabulka 7.: Ukázka tabulky NLeistung

NLeistung	Khs	Leistung	Zusatz	Beschreibung	Beschreibung_Englisch	Hausleistung	Ende datum
SVI MH03837	SVI	MH03837	5	PARALEN	tbl 10x500mg	0	

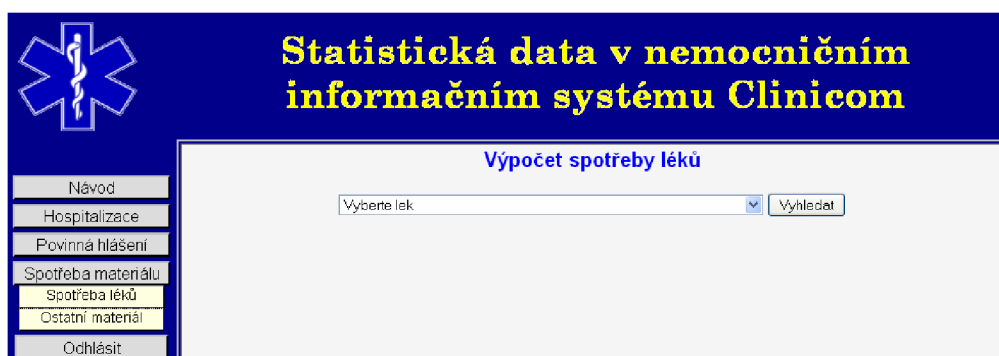
PocetFormat	ZadavatCastTela	Informace	CenaPovinna	Podskupina
5,3	0		0	

Tabulka 8.: Ukázka tabulky ErbrachteNL

ID	L_Datum	Uhrzeit	Leistung	Zusatzwert	Menge	Anf_Kst	Ausf_Stelle
36	20071003	1807	MH03837	1	1	2131	CHIL

Anf_Arzt	Erbr_Arzt	Dienststart	PATient	Episode_Nr
	VS3	1	149	1

Tabulky by měly být propojeny sloupci Leistung, zvýrazněno šedým tučným písmem. Jelikož se sloupce v obou tabulkách jmenují stejně, vzniká problém s autentizací sloupců při SQL dotazu. Proto se sloupce v jedné databázi zásadně nepojmenovávají stejně. Z důvodu stejného pojmenování sloupců nebylo možno propojit tabulky přímo. V rozbalovacím menu pro výběr hledaného léku jsou tedy všechny názvy léků uvedené v tabulce NLeistung a ne jen léky spotřebované. To prodlužuje dobu načítání rozbalovacího menu a zobrazení výpočtu spotřeby vlastního léku. Úvodní stránku funkce „Spotřeba léků“ vidíme na Obr.30.



Obr. 30. Funkce „Spotřeba léků“.

Po zvolení léku a potvrzení se nám zobrazí název léku, jeho kód a celkové množství spotřebovaného léku. Ukázka pro výpočet spotřeby léku Paralen je na Obr.31.

The screenshot displays the 'Statistická data v nemocničním informačním systému Clinicom' interface. On the left is a navigation menu with options: 'Návod', 'Hospitalizace', 'Povinná hlášení', 'Spotřeba materiálu' (highlighted), 'Spotřeba léků', 'Ostatní materiál', and 'Odhlásit'. The main content area is titled 'Výpočet spotřeby léků' and shows the following information: 'Název léku: PARALEN 500 , tbi (10x10)x500mg', 'Kód léku: MH99381', and 'Spotřebované množství léku: 114 ks'. A 'Zpět na zadání' button is located at the bottom of the main area.

Obr. 31. Výpočet spotřeby léku Paralen.

7.7. Spotřeba materiálu – Ostatní materiál

Tato funkce pracuje zcela shodně s funkcí popsanou v předchozí kapitole. Nevyhledává však léky, ale všechny ostatní zdravotnický materiál. Z důvodu nemožnosti přímého propojení tabulek NLeistung a ErbrachteNL, viz kapitola 7.6., a velkého množství typů materiálu v tabulce NLeistung, je načítání rozbalovacího menu a výpočet spotřeby materiálu velmi pomalý.

Závěr

Práce je rozdělena do několika samostatných částí, z nichž každá se věnuje určitému logickému celku. Jednotlivé části se však prolínají a dohromady tvoří souhrnný popis zadaného úkolu. Informace zde obsažené jsou teoretické i praktické.

První část práce popisuje princip, základní funkce a využití nemocničních informačních systémů. Je zde nastíněno, na jakých základech tyto systémy pracují a v čem jsou jejich výhody oproti klasickým papírovým záznamům, které se ve zdravotnictví stále ještě využívají. Zvláštní pozornost je věnována systému NIS Clinicom, který měl být při řešení této práce využit.

Jelikož jednou ze základních úloh každého NIS je správa diagnóz u pacientů, věnuje se druhá část mezinárodní klasifikaci nemocí MKN-10, která v dnešní době platí za celosvětový standard. Je zde popsán vývoj, smysl a princip této klasifikace.

Tato práce se mimo jiné zabývá i vyhledáváním a způsobem hlášení onkologických nálezů, tzv. novotvarů. Proto je třetí kapitola věnována Národním zdravotním registrům, jakožto hlavním činitelům monitorování společensky závažných onemocnění a stavů. Zvláštní pozornost je věnována onkologickým registrům. Součástí práce je i návrh na on-line formulář, který vychází z konkrétních požadavků Národního onkologického registru.

Výstupem diplomové práce je webová aplikace přístupná veřejně na internetu. Tato aplikace obsahuje množství patientských dat, která je ze zákona nutno chránit proti nepovolaným uživatelům. Různými způsoby ochrany dat se zabývá čtvrtá kapitola. Je rozdělena na část teoretickou a na část věnující se konkrétnímu návrhu a realizaci ochrany dat webové aplikace vytvořené v rámci této diplomové práce.

Vzhledem k tomu, že práce je z velké části založena na využití databází, převážně jde o vyhledávání konkrétních dat v nemocničních databázích, je pátá část zaměřena na popis databází využitých konkrétně v NIS Clinicom.

Jednou z podmínek při tvorbě webové aplikace bylo využití jazyka CSP. Popis tohoto jazyka a konkrétní ukázky jeho využití v praxi obsahuje šestá kapitola.

Závěrečná část celé diplomové práce obsahuje konkrétní návrh algoritmu pro získání některých důležitých klinických a statistických dat a prezentace těchto dat na internetu. Konkrétně jde o výskyt novotvarů, výskyt TBC, mortalitu, průměrnou délku hospitalizace pacientů a spotřebu různého lékařského materiálu. Vlastní výpočet i prezentace výsledků byly realizovány pomocí programovacího jazyka CSP. Všechna patientská data využívaná při řešení této práce byla získána ze školní databáze NIS Clinicom VUT.

Tato práce je koncipována jako rozvinutí předchozího semestrálního projektu o využití databázových systémů a o prezentaci výsledků na internetu formou webové aplikace. Je nutno říci, že využití databázových systémů zjednodušuje a zefektivňuje práci s daty. Rovněž umístění výsledků statistických výpočtů a patientských dat na internet zjednodušuje práci zdravotnickému personálu. Data umístěná na internetu jsou přístupnější než data v papírové podobě. Proto snaha o větší zapojení informačních technologií ve zdravotnictví neustále narůstá.

Seznam použitých zdrojů

- [1] Škutková, Helena. *Nemocniční informační systém Clinicom*. Brno, 2007. Bakalářská práce. Vysoké učení technické v Brně. Fakulta elektrotechniky a komunikačních technologií. Vedoucí bakalářské práce Ing. Petr Fedra.
- [2] Zvárová, Jana a kol. *Základy informatiky pro biomedicínu a zdravotnictví*. Karolinum. Praha, 2002.
- [3] World Health organization. *The WHO Family of International Classifications* [online]. 2008 [cit. 2008-03-19]. Dostupné na WWW: <<http://www.who.int>>.
- [4] MUDr. Ondřej Franěk. *Přehled diagnóz MKN-10* [online]. 2002 – 2008, [cit. 2008-04-22]. Dostupné na WWW: <<http://www.zachrannasluzba.cz/propacienty/dg.htm>>.
- [5] Wikipedia The Free Encyclopedia. *Health informatics* [online]. [cit. 2008-04-22]. Dostupné na WWW: <http://en.wikipedia.org/wiki/Health_informatics>.
- [6] Ústav zdravotnických informací a statistiky ČR. *Legislativa NZIS* [online]. Mediasystem. 2007, [cit. 2008-04-23]. Dostupné na WWW: <<http://www.uzis.cz>>.
- [7] Koordinační středisko pro resortní zdravotnické informační systémy. *NOR - Národní onkologický registr* [online]. Praha 4. [cit. 2008-12-04]. Dostupné na WWW: <<http://www.ksrzs.cz/Pages/139-NOR-Narodni-onkologicky-registr.html>>.
- [8] SMS Informační technologie pro české a slovenské zdravotnictví. *Nemocniční informační systém CLINICOM* [online]. 2005, [cit. 2008-12-04]. Dostupné na WWW: <<http://test.nlogy.cz/smsbrno/public/produktysluzby/nis.html>>.
- [9] *Národní onkologický registr*. Ústav zdravotnických informací a statistiky ČR. Praha, 2006.
- [10] Prokopiu, Anna. *Klinický informační systém CLINICOM v Městské nemocnici Ostrava* [online]. 2006, [cit. 2009-01-21]. Dostupné na WWW: <<http://www.cvis.cz/hlavni.php?stranka=novinky/clanek.php&id=487>>.
- [11] František, Skřivánek. *Vítejte ve světě SQL* [online]. 2008, [cit. 2009-01-21]. Dostupné na WWW: <<http://www.dbsvet.cz/view.php?cislocclanku=2008121601>>.
- [12] Zajíc, Petr. *MySQL (16) – (20)* [online]. 2005, [cit. 2009-03-12]. Dostupné na WWW: <http://www.linuxsoft.cz/article_list.php?offset=48&search_article=mysql>.
- [13] Zajíc, Petr. *MySQL (21) – (24)* [online]. 2005, [cit. 2009-03-12]. Dostupné na WWW: <http://www.linuxsoft.cz/article_list.php?offset=32&search_article=mysql>.
- [14] Holub, Martin. *Datový standard ministerstva zdravotnictví České republiky*. Brno, 2008. Bakalářská práce. Vysoké učení technické v Brně. Fakulta elektrotechniky a komunikačních technologií. Vedoucí bakalářské práce Ing. Petr Fedra.

- [15] Dostálek, Libor a kolektiv. *Velký průvodce protokoly TCP/IP Bezpečnost*. Computer Press. Praha, 2003.
- [16] Kirsten, Wolfgang. Ihringer, Michael. Kühn, Mathias. Röhrig, Bernhard. *Caché. Databáze postrelačního typu a tvorba aplikací*. CP Books, a.s. Brno, 2005.
- [17] Janovský, Dušan. *HTTP protokol* [online]. [cit. 2009-04-06]. Dostupné na WWW: <<http://www.jakpsatweb.cz/server/http-protokol.html>>.
- [18] WIKIPEDIE otevřená encyklopedie. *Klient – server* [online]. [cit. 2009-04-09]. Dostupné na WWW: <http://cs.wikipedia.org/wiki/Klient-server#Charakteristika_klienta>.
- [19] WIKIPEDIE otevřená encyklopedie. *Firewall* [online]. [cit. 2009-04-12]. Dostupné na WWW: <<http://cs.wikipedia.org/wiki/Firewall>>.
- [20] IZIP. *Internetový přístup ke zdravotním informacím pacienta* [online]. [cit. 2009-04-19]. Dostupné na WWW: <<https://as.izip.cz/auth/clientlogin/>>.
- [21] Intersystems. *Using Caché Server Pages (CSP)* [online]. 2005, [cit. 2009-04-19]. Dostupné na WWW: <<http://www.intersystems.com/cache/documentation/cache5docs/PDFS/GCSP.pdf>>.
- [22] WIKIPEDIE otevřená encyklopedie. *HTTP cokie* [online]. [cit. 2009-05-01]. Dostupné na WWW: <http://cs.wikipedia.org/wiki/HTTP_cookie>.
- [23] Ústav zdravotnických informací a statistiky ČR. *Údaje a informace dostupné z Národního zdravotnického informačního systému a navazujících zdrojů*. ÚZIS ČR, Praha, 1998.
- [24] *Secure Sockets Layer* [online]. 6.3. 2009 [cit. 2009-04-01]. Dostupné na WWW: <http://cs.wikipedia.org/wiki/Secure_Sockets_Layer>.
- [25] KASAL, Pavel. *Internet a medicína*. Praha: Grada, 2001. ISBN 80-247-0119-7.

Abecední seznam použitých zkratk

ANSI	-	American National Standards Institute
CSP	-	Caché Server Pages
DDL	-	Data Definition Language
DML	-	Data Manipulation Language
DSS	-	Decision Support System – Manažerský informační systém
EKG	-	Elektrokardiogram
E-R	-	Entity-Relationship
GSM	-	Groupe Spécial Mobile - Globální systém pro mobilní komunikaci
HHO	-	World Health Organization
HTTP	-	Hypertext Transfer Protocol
IACR	-	International Association for Cryptologic Research
ICD	-	International Classification of Diseases
IDS	-	Intrusion Detection Systems – systémy pro detekci útoků
IMAP	-	Internet Message Access Protocol
IMIA	-	International Medical Informatics Association
IP	-	Internet Protocol
ISO	-	International Organization for Standardization
MKN	-	Mezinárodní Klasifikace Nemocí
MS	-	MicroSoft
MUMPS	-	Massachusetts general hospital Utility Multi-Programming System
NIS	-	Nemocniční Informační System
NKCHR	-	Národní kardiochirurgický registr
NOR	-	Národní onkologický registr
NRCCH	-	Národní registr cévní chirurgie
NRHOSP	-	Národní registr hospitalizovaných
NRKI	-	Národní registr kardiovaskulárních intervencí
NRKN	-	Národní registr kloubních náhrad
NRNAR	-	Národní registr novorozenců
NRNP	-	Národní registr nemocí z povolání
NRPOT	-	Národní registr potratů
NRROD	-	Národní registr rodiček
NRUSL	-	Národní registr uživatelů lékařsky indikovaných substitučních látek
NRVV	-	Národní registr vrozených vad
NZIS	-	Národní Zdravotní Informační System
ODBC	-	Open Database Connectivity
PACS	-	Picture Archiving and Communication System
RLZF	-	Registr lékařů, zubních lékařů a farmaceutů
SMS	-	Short Message Systems
SMTP	-	Simple Mail Transfer Protocol
SQL	-	Structured Query Language
SSL	-	Secure Socket Layer
TBC	-	Tuberkulóza
TCP	-	Transmission Control Protocol
UBMI	-	Ústav biomedicínkého inženýrství
ÚZIS	-	Ústav zdravotnických informací a statistiky
XML	-	Extensible Markup Language
ZN	-	Zhoubný novotvar

Příloha č.1 – Kompletní zdrojový kód webové aplikace

Soubor DPVelicka.csp

```
<HTML>
<HEAD>
<TITLE>Diplomova práce, Tomas Velicka</TITLE>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1250" />
<script NOCHARSETCONVERT="1">
<link rel="stylesheet" type="text/css" href="styly.css">
<script LANGUAGE="Javascript" NAME="skryte">
function Ukaz(specific){
el=document.getElementById(specific).style;
el.display=(el.display == 'block')?'none':'block';
}
</script>

</HEAD>
<BODY bgcolor="white">
<script LANGUAGE="CACHE" RUNAT="SERVER">
Set %session.Data("pristup") = ""
Set %session.AppTimeout = 10800
</SCRIPT>
<CSP:IF CONDITION='$G(%request.Data("heslo",1))="doktor"'>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set %session.Data("pristup") = "ok"
</SCRIPT>

<div id="content">
<div id="logo">
<table align="center" width="100%" high="100%" cellpadding="6">
<tr><td align="left" width="10%"></td>
<td align="center" ></td></tr></table>
</div>

<div id="menu2">
<br><br>
<table class="menu">
<tbody>
<tr><td><a href="Navod.csp" target="prace" class="tlacitko2">Návod</a></td></tr>

<tr><td><div class="tlacitko2" onclick="Ukaz('hospitalizace')">Hospitalizace</div>
<div id="hospitalizace" class="skryvany"><center>
<a href="Hospitalizace.csp" target="prace" class="tlacitko3" align="center">Pr&#367;m&#283;rná doba</a>
<a href="Mortalita.csp" target="prace" class="tlacitko3">Výpo&#269;et
mortality</a></center></td></tr></div>

<tr><td><div class="tlacitko2" onclick="Ukaz('hlaseni')">Povinná hlášení</div>
<div id="hlaseni" class="skryvany"><center>
<a href="vyskytnovotvaru.csp" target="prace" class="tlacitko3">Výskyt novotvar&#367;</a>
<a href="hlaseninovotvaru.csp" target="prace" class="tlacitko3">Hlášení novotvar&#367;</a>
<a href="TBC.csp" target="prace" class="tlacitko3">Výskyt TBC</a></center></td></tr></div>

<tr><td><div class="tlacitko2" onclick="Ukaz('material')">Spot&#345;eba materiálu</div>
<div id="material" class="skryvany"><center>
<a href="leky.csp" target="prace" class="tlacitko3">Spot&#345;eba lék&#367;</a>
<a href="material.csp" target="prace" class="tlacitko3">Ostatní materiál</a></center></td></tr></div>
```

```

<tr><td align="center"><form name="form" method="post" action="DPVelicka.csp">
<input type="submit" name="odhlasit" class="menu" value="Odhlásit"></form>
</td></tr>
</tbody></table>
</div>

<div id="text">
<table border="2" cellpadding="2" cellspacing="2" height="100%" width="100%">
<tbody>
<tr>
<td align="center" bgcolor="#f5f5f5" height="950" valign="top" width="100%">
<iframe src="Navod.csp" width="100%" height="100%" frameborder="0" align="baseline"
name="prace"></iframe>
</td>
</tr>
</tbody>
</table>
</div>

<div id="spodek">
<table align="center">
<tr><td align="center">
<font color="#d2691e" size="2">Tyto stránky vytvořil Tomáš Velička v rámci Diplomové práce,
VUT v Brně, 2009</font>
</td></tr></table>
</div>
</div>

<!-- pokud není přihlášen uživatel -->
<CSP:ELSE>
<div id="content">

<div id="logo">
<table align="center" width="100%" height="100%" cellpadding="6">
<tr><td align="left" width="10%"></td>
<td align="center" ></td></tr>
</table>
</div>

<table border="0" height="100%" width="100%">
<tbody>
<tr>
<td align="center" height="1200" valign="top" width="100%">
<center><form name="form" method="post" action="DPVelicka.csp">
<br><br>
<table align="center">
<tr><font color="white"><h2>Přihlášení </h2></font></tr>
<tr><td><font color="white">Jméno:</td><td> </font><input type="text" name="jmeno"
value="doktor"><br></td>
<tr><td></tr></td>
<tr><td><font color="white">Heslo:</td><td> </font><input type="password" name="heslo"
value=""><br></td>
</table>
<br><input type="submit" name="prihlas" value="Přihlásit">
</form>

<font color="white">*) pro správnou funkci stránek je nutné povolit cookies!</center>
<CSP:IF CONDITION='$G(%request.Data("prihlas",1))="">
<CSP:ELSE>

```

```

<CSP:IF CONDITION='$G(%request.Data("odhlasit",1))="">
<CSP:IF CONDITION='$G(%request.Data("heslo",1))="doktor">
<CSP:ELSE>
<script LANGUAGE="Javascript" NAME="alert">
alert("Špatné heslo! Zadejte heslo znovu");
</script>
</CSP:IF>
</CSP:IF>
</CSP:IF>
<CSP:IF CONDITION='$G(%request.Data("odhlasit",1))="Odhlásit">
<script LANGUAGE="Javascript" NAME="alert">
alert("Úspěšné odhlášení uživatele");
</script>
</CSP:IF>
</td>
</tr>
</tbody>
</table>

<div id="spodek">
<table align="center">
<tr><td align="center">
<font color="#d2691e" size="2">Tyto stránky vytvořil Tomáš Velička v rámci Diplomové práce,
VUT v Brně, 2009</font>
</td></tr></table>
</div>
</div>
</CSP:IF>
</BODY>
</HTML>

```

Soubor Navod.csp

```

<HTML>
<HEAD>
<TITLE>Cache Server Page </TITLE>
<link rel="stylesheet" type="text/css" href="styly.css">
</HEAD>

<BODY bgcolor="white">
<CSP:IF CONDITION=%session.Data("pristup") = "ok">
<table>
<tr><td>
<h3 class="nadpis_navod">Slovo autora</h3>
<p>Tyto stránky vznikly v rámci diplomové práce na VUT v Brně, FEKT, 2009. <br> V dalších odstavcích jsou popsány jednotlivé funkce těchto stránek, určené pro výpočet některých statistických údajů; z dat získaných ze školní patientské databáze. Funkce jsou dle hlavního obsahu rozděleny do tří hlavních oblastí: Hospitalizace, Povinná hlášení a Spotřeba Materiálu. Jednotlivé funkce se spouští stiskem příslušného tlačítka v menu stránky.</p>

<h3 class="nadpis_navod_red">Hospitalizace</h3>
<h3 class="nadpis_navod">Průměrná doba</h3>
<p>Tato funkce vypíše seznam všech pacientů propuštěných z nemocničního ošetření ve zvoleném roce, doplněný o některé údaje. Na konci seznamu (ve spodní části stránky) je uvedena průměrná doba ošetření. Funkce je doplněna o možnost zobrazení grafu.</p>

<h3 class="nadpis_navod">Výpočet mortality</h3>

```

<p>Po zvolení roku se zobrazí seznam zesnulých pacientů; s nimiž; kterými základními údaji. Pod tabulkou pacientů; je vypsan jejich počet pro daný rok. Funkce je doplněna o možnost zobrazení grafu.

<h3 class="nadpis_navod_red">Povinná hlášení</h3>
<h3 class="nadpis_navod">Výskyt novotvarů;</h3>
<p>V rolovací nabídce vybereme požadovanou diagnózu. Poté můžeme vybrat konkrétní rok a síc (musí být vybráno souasně;), nebo zaškrtnutím polí;ka "všechny" zvolíme výpis diagnózy pro všechna období. Po stisku tlačítka "Vyhledat" se zobrazí tabulka s údaji o pacientech, u kterých byla nalezena zvolená diagnóza pro zvolené období.

<h3 class="nadpis_navod">Hlášení novotvarů;</h3>
<p>Označením kterého z tlačítek "poslední rok", "poslední síc" nebo "poslední den", zvolíme období, pro které chceme vypsat všechny diagnózy popisující výskyt novotvaru. Po stisku tlačítka "Vyhledat" se zobrazí tabulka s údaji o pacientech, u kterých byla nalezena zvolená diagnóza pro zvolené období. V tabulce je na každém řádku vpravo na konci tlačítko "Formulář;". Stiskem tohoto tlačítka zobrazíme ukázkou vyplněného hlášení o nález nového novotvaru. Toto hlášení je doplněno samostatnou návodou.

<h3 class="nadpis_navod">Výskyt TBC</h3>
<p>Funkce pracuje shodně; s funkcí "Výskyt novotvarů;". Vyhledává však pacienty u nichž byl prokázán výskyt TBC.

<h3 class="nadpis_navod_red">Spotřeba materiálu</h3>
<h3 class="nadpis_navod">Spotřeba léků;</h3>
<p>Z nabídky vybereme požadovaný lék a potvrdíme tlačítkem "Vyhledat". Zobrazí se počet spotřebovaných kusů; zvoleného léku.

<h3 class="nadpis_navod">Ostatní materiál</h3>
<p>Funkce pracuje shodně; s funkcí "Spotřeba léků;". Vyhledává ale zdravotnický materiál vyjma léků;.

</td>
</tr></table>
<CSP:ELSE>

<h3 class="result">Nejste přihlášení!</h3>

Návrat na hlavní stránku.
</CSP:IF>
</BODY>
</HTML>

Soubor Hospitalizace.csp

<HTML>
<HEAD>

<!-- Put your page Title here -->
<TITLE>Hospitalizace - průměrná doba</TITLE>
<link rel="stylesheet" type="text/css" href="style.css">

```

</HEAD>

<BODY bgcolor="white">
<CSP:IF CONDITION=%session.Data("pristup") = "ok">
<SCRIPT language="SQL" name="hodnoty">
select SUBSTRING(NZEpizoda,1,3) AS cislo, NZPropusteniDruh, SUBSTRING(NZAktivitaDatum,1,4) AS
rok,
SUBSTRING(NZAktivitaDatum,5,2) AS mesic, SUBSTRING(NZAktivitaDatum,7,2) AS den,
SUBSTRING(NZAktivitaDatum,9,2) AS hodina, SUBSTRING(NZAktivitaDatum,11,2) AS minuta
from SQLUser.VNZIS
</SCRIPT>
<h3 class="nadpis">Výpočet průměrné doby hospitalizace pacientů v nemocnici</h3>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set cislonove=0 //první načtení čísla pacienta pro porovnání
set roknoy=0 //první načtení pro porovnání
set mesicnoy=0 //první načtení pro porovnání
set dennoy=0 //první načtení pro porovnání
set hodinanova=0 //první načtení pro porovnání
set minutanova=0 //první načtení pro porovnání
set odecetrok=0 //v případě Hospitalizace22 delšího než 1 rok
set soucetdni=0 //delka Hospitalizace22 pro jednoho pacienta ve dnech, pokud je ostereni kratši než 1 rok
set celkovysoucetdni=0 //delka Hospitalizace22 pro všechny pacienty ve dnech, pokud je ostereni kratši než 1
rok
set soucetdni2=0 //delka Hospitalizace22 pro jednoho pacienta ve dnech, pokud je ostereni delší než 1 rok
set celkovysoucetdni2=0 //delka Hospitalizace22 pro všechny pacienty ve dnech, pokud je ostereni delší než 1
rok
set e=0 //počet propuštěných pacientů
set e1=0
set e2=0
set f=0 //průměrná doba Hospitalizace ve dnech
set f1=0
set vysledek=0
set celkovysoucetdni07=0 set celkovysoucetdni207=0 set e07=0 set f07=0

set celkovysoucetdni08=0 set celkovysoucetdni208=0 set e08=0 set f08=0
set cislonove08 = 0 set roknoy08 = 0 set mesicnoy08 = 0 set dennoy08 = 0
set hodinanova08 = 0 set minutanova08 = 0 set e108=0 set e208=0
set odecetrok08=0
set soucetdni08=0
set soucetdni208=0

set celkovysoucetdni09=0 set celkovysoucetdni209=0 set e09=0 set f09=0
set cislonove09 = 0 set roknoy09 = 0 set mesicnoy09 = 0 set dennoy09 = 0
set hodinanova09 = 0 set minutanova09 = 0 set e109=0 set e209=0
set odecetrok09=0
set soucetdni09=0
set soucetdni209=0
</SCRIPT>

<CSP:IF CONDITION=$G(%request.Data("graf",1))="">
<CSP:IF CONDITION=$G(%request.Data("special",1))="">
<form name="form1" method="post" action="Hospitalizace.csp">
<h5 class="popis">Zobraz hospitalizované pacienty propuštěné v roce</h5>
<center><table>
<tr><td><input type="radio" name="special" value="2007" checked>2007</td>
<td><input type="radio" name="special" value="2008">2008</td>
<td><input type="radio" name="special" value="2009">2009</td></tr>
</table>
<br>
<input type="submit" name="Submit" value="Vyhledat"> <form name="form1" method="get"

```

```

action="Hospitalizace.csp">
<input type="submit" name="graf" value="Zobraz celkový graf">
</form>
</center>
</form>

<CSP:ELSE>
<h5 class="popis">Údaje o pacientech s ukončenou hospitalizací v roce
#(%request.Data("special",1))#</h5>
<table class="printout" border="2" align="center">
<tr class="printout_first_row">
<td><b> P&#345;ijmení</b></td>
<td><b> Jméno</b></td>
<td><b> Rodné &#269;íslo</b></td>
<td align="center"><b>Datum p&#345;ijetí</b></td>
<td align="center"><b>Datum propuštění</b></td>
<td align="center"><b>Délka hospitalizace</b></td>
</tr>

<CSP:WHILE CONDITION=hodnoty.Next()>
<SCRIPT LANGUAGE="SQL" NAME="pacient">
SELECT *
From SQLUser.VPacient where PAPACIENT=#(hodnoty.Data("cislo"))#
ORDER by PAJmeno
</SCRIPT>

<CSP:IF CONDITION=(hodnoty.Data("cislo"))=cislonove'> <!-- pokud jsou za sebou dve shodna cisla pacientu
-->
<CSP:IF CONDITION=(hodnoty.Data("NZPropusteniDruh"))="">
<CSP:ELSE>
<CSP:WHILE CONDITION=pacient.Next()>
<CSP:IF CONDITION=(hodnoty.Data("rok"))=roknovy'> <!-- pokud je vysetreni kratši než rok -->
<CSP:IF CONDITION=(hodnoty.Data("rok"))=$G(%request.Data("special",1))'>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni= ((hodnoty.Data("mesic"))*30) + (hodnoty.Data("den")) - (mesicnovy*30 + dennovy)
set celkovysoucetdni= celkovysoucetdni + soucetdni
set e1=e1 + 1 //pocita kolik je pacientu
</SCRIPT>
<CSP:IF CONDITION='soucetdni>0'>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni = soucetdni + 1
set celkovysoucetdni= celkovysoucetdni + 1
</script>
<CSP:ELSE>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni = soucetdni
set celkovysoucetdni= celkovysoucetdni
</script>
</script>
</CSP:IF>
<tr>
<td width="20%">#(pacient.Data("PAPrijmeni"))#</td>
<td width="20%">#(pacient.Data("PAJmeno"))#</td>
<td width="15%">#(pacient.Data("PACisloRodne"))#</td>
<td width="15%">#(dennovy)#.(mesicnovy)#.(roknovy)#</td>
<td width="20%">#(hodnoty.Data("den"))#.(hodnoty.Data("mesic"))#.(hodnoty.Data("rok"))#</td>
<td width="25%">#(soucetdni)#
<!-- testovani kvuli spravnému mluvnickému tvaru slova den -->
<CSP:IF CONDITION='soucetdni=0'>
dni

```



```

<CSP:ELSEIF CONDITION='soucetdni=1'>
den
<CSP:ELSEIF CONDITION='soucetdni<5'>
dny
<CSP:ELSEIF CONDITION='soucetdni>4'>
dni
</CSP:IF>
</td>
</CSP:IF>
<CSP:ELSE>
<CSP:IF CONDITION='(hodnoty.Data("rok"))=$G(%request.Data("special",1))'>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set odecetrok= (hodnoty.Data("rok")) - roknovy //vypocita rozdil dat prijati a propusteni daneho pacienta - rok
set soucetdni2= ((odecetrok*365) + ((hodnoty.Data("mesic"))*30) + (hodnoty.Data("den"))) - (mesicnovy*30 +
dennovy)
set celkovysoucetdni2= celkovysoucetdni2 + soucetdni2
set e2=e2 + 1 //pocita kolik je pacientu
</SCRIPT>
<CSP:IF CONDITION='soucetdni2>0'>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni2 = soucetdni2 + 1
set celkovysoucetdni2= celkovysoucetdni2 + 1
</script>
<CSP:ELSE>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni2 = soucetdni2
set celkovysoucetdni2= celkovysoucetdni2
</script>
</CSP:IF>
<tr>
<td width="20%">#(pacient.Data("PAPrijmeni"))#</td>
<td width="20%">#(pacient.Data("PAJmeno"))#</td>
<td width="15%">#(pacient.Data("PACisloRodne"))#</td>
<td width="15%">#(dennovy)#.(mesicnovy)#.(roknovy)#</td>
<td width="20%">#(hodnoty.Data("den"))#.(hodnoty.Data("mesic"))#.(hodnoty.Data("rok"))#</td>
<td width="25%">#(soucetdni2)#
<!-- testovani kvuli spravnemu mluvnickemu tvaru slova den -->
<CSP:IF CONDITION='soucetdni=0'>
dni
<CSP:ELSEIF CONDITION='soucetdni=1'>
den
<CSP:ELSEIF CONDITION='soucetdni<5'>
dny
<CSP:ELSEIF CONDITION='soucetdni>4'>
dni
</CSP:IF>
</td>
</tr>
</CSP:IF>
</CSP:IF>
</CSP:IF>
</CSP:WHILE>

<script LANGUAGE="CACHE" RUNAT="SERVER">
set e=e1 + e2 //pocita kolik je pacientu
</SCRIPT>
</CSP:IF>
<CSP:ELSE>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set cislonove = (hodnoty.Data("cislo")) //nastavi nove cislo pro porovnaní
set roknovy = (hodnoty.Data("rok")) //nastavi nove datum - rok

```

```

set mesicnovy = (hodnoty.Data("mesic")) //nastavi nove datum - mesic
set dennovy = (hodnoty.Data("den")) //nastavi nove datum - den
set hodinanova = (hodnoty.Data("hodina")) //nastavi nove datum - hodina
set minutanova = (hodnoty.Data("minuta")) //nastavi nove datum - minuta
</script>
</CSP:IF>
</CSP:IF>
</CSP:WHILE></table>

<table align="center">
<tr><td align="center">
<br>
<h3 class="result">Pr&#367;m&#283;rná doba ošet&#345;ení: <b>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set f = (celkovysoucetdni + celkovysoucetdni2)/e
write $FNUMBER(f,"",2)
</script> dní</b></h3><br>
</td>
</tr></table>

<center>
<form name="form1" method="post" action="Hospitalizace.csp">
<input type="hidden" name="special" value="">
<input type="submit" name="zpet" value="Zp&#283;t na zadání">
</form></center>
</CSP:IF>

<CSP:ELSE>
<SCRIPT language="SQL" name="hodnotygraf">
select SUBSTRING(NZEpizoda,1,3) AS cislo, NZPropusteniDruh, SUBSTRING(NZAktivitaDatum,1,4) AS
rok,
SUBSTRING(NZAktivitaDatum,5,2) AS mesic, SUBSTRING(NZAktivitaDatum,7,2) AS den,
SUBSTRING(NZAktivitaDatum,9,2) AS hodina, SUBSTRING(NZAktivitaDatum,11,2) AS minuta
from SQLUser.VNZIS
</SCRIPT>
<CSP:WHILE CONDITION=hodnotygraf.Next()>

<!-- 2007 -->

<CSP:IF CONDITION=(hodnotygraf.Data("cislo"))=cislonove> <!-- pokud jsou za sebou dve shodna cisla
pacientu -->
<CSP:IF CONDITION=(hodnotygraf.Data("NZPropusteniDruh"))="">
<CSP:ELSE>
<CSP:IF CONDITION=(hodnotygraf.Data("rok"))=roknovy'> <!-- pokud je vysetreni kratši nez rok -->
<CSP:IF CONDITION=(hodnotygraf.Data("rok"))="2007">
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni= (((hodnotygraf.Data("mesic"))*30) + (hodnotygraf.Data("den")) - (mesicnovy*30 + dennovy)
set celkovysoucetdni07= celkovysoucetdni07 + soucetdni
set e1=e1 + 1 //pocita kolik je pacientu
</SCRIPT>
<CSP:IF CONDITION='soucetdni>0'>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni = soucetdni + 1
set celkovysoucetdni07= celkovysoucetdni07 + 1
</script>
<CSP:ELSE>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni = soucetdni
set celkovysoucetdni07= celkovysoucetdni07
</script>

```

```

</script>
</CSP:IF>

<script LANGUAGE="CACHE" RUNAT="SERVER">
set odecetrok= (hodnotygraf.Data("rok")) - roknovy //vypocita rozdil dat prijeti a propusteni daneho pacienta -
rok
set soucetdni2= ((odecetrok*365) + ((hodnotygraf.Data("mesic"))*30) + (hodnotygraf.Data("den"))) -
(mesicnovy*30 + dennovy)
set celkovysoucetdni207= celkovysoucetdni207 + soucetdni2
set e2=e2 + 1 //pocita kolik je pacientu
</SCRIPT>
<CSP:IF CONDITION='soucetdni2>0'>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni2 = soucetdni2 + 1
set celkovysoucetdni207= celkovysoucetdni207 + 1
</script>
<CSP:ELSE>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni2 = soucetdni2
set celkovysoucetdni207= celkovysoucetdni207
</script>
</CSP:IF>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set e07=e1 + e2 //pocita kolik je pacientu
</SCRIPT>
</CSP:IF></CSP:IF></CSP:IF>
<CSP:ELSE>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set cislonove = (hodnotygraf.Data("cislo")) //nastavi nove cislo pro porovnani
set roknovy = (hodnotygraf.Data("rok")) //nastavi nove datum - rok
set mesicnovy = (hodnotygraf.Data("mesic")) //nastavi nove datum - mesic
set dennovy = (hodnotygraf.Data("den")) //nastavi nove datum - den
set hodinanova = (hodnotygraf.Data("hodina")) //nastavi nove datum - hodina
set minutanova = (hodnotygraf.Data("minuta")) //nastavi nove datum - minuta
</script>
</CSP:IF>

<!-- 2008 -->

<CSP:IF CONDITION=(hodnotygraf.Data("cislo"))=cislonove08> <!-- pokud jsou za sebou dve shodna cisla
pacientu -->
<CSP:IF CONDITION=(hodnotygraf.Data("NZPropusteniDruh"))="">
<CSP:ELSE>
<CSP:IF CONDITION=(hodnotygraf.Data("rok"))=roknovy08> <!-- pokud je vysetreni kratši než rok -->
<CSP:IF CONDITION=(hodnotygraf.Data("rok"))="2008">
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni08= (((hodnotygraf.Data("mesic"))*30) + (hodnotygraf.Data("den"))) - (mesicnovy08*30 +
dennovy08)
set celkovysoucetdni08= celkovysoucetdni08 + soucetdni08
set e108=e108 + 1 //pocita kolik je pacientu
</SCRIPT>
<CSP:IF CONDITION='soucetdni08>0'>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni08 = soucetdni08 + 1
set celkovysoucetdni08= celkovysoucetdni08 + 1
</script>
<CSP:ELSE>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni08 = soucetdni08
set celkovysoucetdni08= celkovysoucetdni08

```

```

</script>
</script>
</CSP:IF></CSP:IF>

<CSP:ELSE>
<CSP:IF CONDITION='(hodnotygraf.Data("rok"))="2008"'>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set odecetrok08= (hodnotygraf.Data("rok")) - roknoy08 //vypocita rozdil dat prijeti a propusteni daneho
pacienta - rok
set soucetdni208= ((odecetrok08*365) + ((hodnotygraf.Data("mesic"))*30) + (hodnotygraf.Data("den"))) -
(mesicnoy08*30 + dennoy08)
set celkovysoucetdni208= celkovysoucetdni208 + soucetdni208
set e208=e208 + 1 //pocita kolik je pacientu
</SCRIPT>
<CSP:IF CONDITION='soucetdni208>0'>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni208 = soucetdni208 + 1
set celkovysoucetdni208= celkovysoucetdni208 + 1
</script>
<CSP:ELSE>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni208 = soucetdni208
set celkovysoucetdni208= celkovysoucetdni208
</script>
</CSP:IF>
</CSP:IF>
</CSP:IF>

<script LANGUAGE="CACHE" RUNAT="SERVER">
set e08=e108 + e208 //pocita kolik je pacientu
</SCRIPT>
</CSP:IF>
<CSP:ELSE>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set cislonove08 = (hodnotygraf.Data("cislo")) //nastavi nove cislo pro porovnani
set roknoy08 = (hodnotygraf.Data("rok")) //nastavi nove datum - rok
set mesicnoy08 = (hodnotygraf.Data("mesic")) //nastavi nove datum - mesic
set dennoy08 = (hodnotygraf.Data("den")) //nastavi nove datum - den
set hodiny08 = (hodnotygraf.Data("hodina")) //nastavi nove datum - hodina
set minutanova08 = (hodnotygraf.Data("minuta")) //nastavi nove datum - minuta
</script>
</CSP:IF>
</CSP:IF>

<!-- 2009 -->

<CSP:IF CONDITION='(hodnotygraf.Data("cislo"))=cislonove09'> <!-- pokud jsou za sebou dve shodna cisla
pacientu -->
<CSP:IF CONDITION='(hodnotygraf.Data("NZPropusteniDruh"))=""'>
<CSP:ELSE>
<CSP:IF CONDITION='(hodnotygraf.Data("rok"))=roknoy09'> <!-- pokud je vysetreni kratsi nez rok -->
<CSP:IF CONDITION='(hodnotygraf.Data("rok"))="2009"'>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni09= (((hodnotygraf.Data("mesic"))*30) + (hodnotygraf.Data("den"))) - (mesicnoy09*30 +
dennoy09)
set celkovysoucetdni09= celkovysoucetdni09 + soucetdni09
set e109=e109 + 1 //pocita kolik je pacientu
</SCRIPT>
<CSP:IF CONDITION='soucetdni09>0'>
<script LANGUAGE="CACHE" RUNAT="SERVER">

```

```

set soucetdni09 = soucetdni09 + 1
set celkovysoucetdni09= celkovysoucetdni09 + 1
</script>
<CSP:ELSE>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni09 = soucetdni09
set celkovysoucetdni09= celkovysoucetdni09
</script>
</script>
</CSP:IF></CSP:IF>

<CSP:ELSE>
<CSP:IF CONDITION='(hodnotygraf.Data("rok"))="2009"'>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set odecetrok09= (hodnotygraf.Data("rok")) - roknovy09 //vypocita rozdil dat prijeti a propusteni daneho
pacienta - rok
set soucetdni209= ((odecetrok09*365) + ((hodnotygraf.Data("mesic"))*30) + (hodnotygraf.Data("den")) -
(mesicnovy09*30 + dennovy09)
set celkovysoucetdni209= celkovysoucetdni209 + soucetdni209
set e209=e209 + 1 //pocita kolik je pacientu
</SCRIPT>
<CSP:IF CONDITION='soucetdni209>0'>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni209 = soucetdni209 + 1
set celkovysoucetdni209= celkovysoucetdni209 + 1
</script>
<CSP:ELSE>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set soucetdni209 = soucetdni209
set celkovysoucetdni209= celkovysoucetdni209
</script>
</CSP:IF>
</CSP:IF>
</CSP:IF>

<script LANGUAGE="CACHE" RUNAT="SERVER">
set e09=e109 + e209 //pocita kolik je pacientu
</SCRIPT>
</CSP:IF>
<CSP:ELSE>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set cislonove09 = (hodnotygraf.Data("cislo")) //nastavi nove cislo pro porovnaní
set roknovy09 = (hodnotygraf.Data("rok")) //nastavi nove datum - rok
set mesicnovy09 = (hodnotygraf.Data("mesic")) //nastavi nove datum - mesic
set dennovy09 = (hodnotygraf.Data("den")) //nastavi nove datum - den
set hodinanova09 = (hodnotygraf.Data("hodina")) //nastavi nove datum - hodina
set minutanova09 = (hodnotygraf.Data("minuta")) //nastavi nove datum - minuta
</script>
</CSP:IF>
</CSP:IF>
</CSP:WHILE>

<script LANGUAGE="CACHE" RUNAT="SERVER">
set f07 = (celkovysoucetdni07 + celkovysoucetdni207)/e07
set f08 = (celkovysoucetdni08 + celkovysoucetdni208)/e08
set f09 = (celkovysoucetdni09 + celkovysoucetdni209)/e09
set f08procento = (f08/f07)*100 - 100
set f09procento = (f09/f08)*100 - 100
</script>

```

```

<!-- graf -->
<CSP:IF CONDITION='$G(%request.Data("graf",1))["Zobraz"]>
Grafické zobrazení průměrné doby hospitalizace pacientů v jednotlivých letech
<br> <br>

<br><br>
Tabulka konkrétních hodnot pro jednotlivé roky
<center>
<table class="graf" border="1">
<tr class="printout_first_row"><td width="360"></td><td width="150">2007</td><td
width="150">2008</td><td width="150">2009</td></tr>
<tr><td>počet dní</td><td><script LANGUAGE="CACHE" RUNAT="SERVER">
write $FNUMBER(f07,"",2)
</script> </td>
<td> <script LANGUAGE="CACHE" RUNAT="SERVER">
write $FNUMBER(f08,"",2)
</script> </td>
<td><script LANGUAGE="CACHE" RUNAT="SERVER">
write $FNUMBER(f09,"",2)
</script> </td></tr>
<tr><td>zvýšení/snížení oproti předchozímu roku</td><td>--</td>
<td> <script LANGUAGE="CACHE" RUNAT="SERVER">
write $FNUMBER(f08procento,"",2)
</script> %</td>
<td><script LANGUAGE="CACHE" RUNAT="SERVER">
write $FNUMBER(f09procento,"",2)
</script> %</td></tr>
</table></center><br>
<form name="form4" method="post" action="Hospitalizace.csp">
<input type="hidden" name="zvoldiag" value="">
<input type="submit" name="zpet" value="Zpět na zadání roku">
</form>
</CSP:IF>
</CSP:IF>

<CSP:ELSE><br><br>
<h3 class="result">Nejste přihlášení!<b></b></h3><br>
<a href="DPVvelicka.csp"><font color="blue">Návrat na hlavní stránku.</font></a>
</CSP:IF>
</BODY>
</HTML>

```

Soubor Mortalita.csp

```

<HTML>
<HEAD>
<TITLE>Cache Server Page </TITLE>
<link rel="stylesheet" type="text/css" href="styly.css">
</HEAD>

<BODY bgcolor="white">
<CSP:IF CONDITION='%session.Data("pristup") = "ok">
<table align="center" width="100%">
<tr><td align="center">
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a=0
set b=0
set c=0

```

```

set d=0
set umrti=0
set umrti07=0
set umrti08=0
set umrti09=0
</script>
<SCRIPT language="SQL" name="kodyd">
select *
from SQLUser.VNZIS
</SCRIPT>

```

<h3 class="nadpis">Výpočet mortality pacientů v nemocnici</h3>

```

<CSP:WHILE CONDITION=kodyd.Next()>
<!-- Urcení roku které jsou obsaženy v databázi -->
<CSP:IF CONDITION=(kodyd.Data("NZAktivitaDatum"))["2007"]>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a = a + 1
</script>
<CSP:ELSEIF CONDITION=(kodyd.Data("NZAktivitaDatum"))["2008"]>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set b = b + 1
</script>
<CSP:ELSEIF CONDITION=(kodyd.Data("NZAktivitaDatum"))["2009"]>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set c = c + 1
</script>
</CSP:IF>
<CSP:IF CONDITION=(kodyd.Data("NZPropusteniDruh"))>"6">
<CSP:IF CONDITION=(kodyd.Data("NZAktivitaDatum"))[$G(%request.Data("zvolrok",1))]>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set umrti = umrti + 1
</script>
</CSP:IF>
</CSP:IF>
<CSP:IF CONDITION=(kodyd.Data("NZPropusteniDruh"))>"6">
<CSP:IF CONDITION=(kodyd.Data("NZAktivitaDatum"))["2007"]>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set umrti07 = umrti07 + 1
</script>
<CSP:ELSEIF CONDITION=(kodyd.Data("NZAktivitaDatum"))["2008"]>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set umrti08 = umrti08 + 1
</script>
<CSP:ELSEIF CONDITION=(kodyd.Data("NZAktivitaDatum"))["2009"]>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set umrti09 = umrti09 + 1
</script>
</CSP:IF>
</CSP:IF>
</CSP:WHILE>

```

```

<!-- *** Zobrazení selectu a nahrání nabídek jednotlivých roku *** -->
<CSP:IF CONDITION=$G(%request.Data("graf",1))="">
<CSP:IF CONDITION=$G(%request.Data("zvolrok",1))="">
<form name="form1" method="get" action="Mortalita.csp">
<center><table class="year">
<tr><td>

```

<h5 class="popis">Zobraz pacienty zesnulé v nemocnici v roce</h5>

```

<center>
<table>
<tr><td><input type="radio" name="zvolrok" value="2007" checked>2007</td>
<td><input type="radio" name="zvolrok" value="2008">2008</td>
<td><input type="radio" name="zvolrok" value="2009">2009</td></tr>
</table></center>
<br>
<input type="submit" name="Submit" value="Vyhledat">
<form name="form1" method="get" action="Mortalita.csp">
<input type="submit" name="graf" value="Zobraz celkový graf">
</form>
</tr></td>
</form></table></center>

<CSP:ELSE>
<CSP:IF CONDITION=$G(%request.Data("zvolrok"),1)="0">
<h3 class="attention"> Musíte vybrat rok </h3><center>
<form name="form4" method="post" action="Mortalita.csp">
<input type="hidden" name="zvoldiag" value="">
<input type="submit" name="zpet" value="Zp&#283t na zadání roku"></center>
</form>

<CSP:ELSE>
<h3 class="result">po&#269et úmrtí v roce #($G(%request.Data("zvolrok"),1))#: #(umrti)# </h3>
<CSP:IF CONDITION=$G(%request.Data("show_details",1))["Zo">

<h5 class="popis">Údaje o zesnulých pacientech</h5>
<center><table class="printout" border="2" width="100%">
<tr class="printout_first_row">
<td><b> P&#345;ijmení</b></td>
<td><b> Jméno</b></td>
<td align="center"><b>Rodné &#269;íslo</b></td>
<td><b>Datum úmrtí</b></td>
</tr>

<SCRIPT language="SQL" name="kodyd2">
select SUBSTRING(NZEpozoda,1,3) AS Cislo, NZPropusteniDruh, NZAktivitaDatum,
substring (NZAktivitaDatum,1,4) as rok,
substring (NZAktivitaDatum,5,2) as mesic, substring (NZAktivitaDatum,7,2) as den,
substring (NZAktivitaDatum,9,2) as hodina, substring (NZAktivitaDatum,11,2) as minuta
from SQLUser.VNZIS
</SCRIPT>

<CSP:WHILE CONDITION=kodyd2.Next()>
<SCRIPT language="SQL" name="pacient">
select *
From SQLUser.VPacient where PAPACIENT=#(kodyd2.Data("Cislo"))#
order by PAPrijmeni
</SCRIPT>

<CSP:WHILE CONDITION=pacient.Next()>
<CSP:IF CONDITION=(kodyd2.Data("NZAktivitaDatum"))[$G(%request.Data("zvolrok"),1)]>
<CSP:IF CONDITION=(kodyd2.Data("NZPropusteniDruh"))>"6">
<tr>
<td width="20%">&nbsp;#(pacient.Data("PAPrijmeni"))#</td>
<td width="20%">&nbsp;#(pacient.Data("PAJmeno"))#</td>
<td width="20%">#(pacient.Data("PACisloRodne"))#</td>
<td align="center" width="20%">#(kodyd2.Data("den"))#.#(kodyd2.Data("mesic"))#.#
#(kodyd2.Data("rok"))# v #(kodyd2.Data("hodina"))#.#(kodyd2.Data("minuta"))#</td>
</CSP:IF></CSP:IF> </CSP:WHILE></center>

```



```

</CSP:WHILE></table>
<br>
</CSP:IF>

<CSP:IF CONDITION=$G(%request.Data("show_details",1))["Zo"]>
<form name="form7" method="post" action="Mortalita.csp">
<input type="hidden" name="zvolrok" value="#(%request.Data("zvolrok",1))#">
<input type="submit" name="show_details" value="Zav&#345ít údaje o pacientech">
</form><center>
<!-- nabídka zobraz udaje o pacientech pokud neni zobrazeno -->
<CSP:ELSE>
<form name="form6" method="post" action="Mortalita.csp">
<input type="hidden" name="zvolrok" value="#(%request.Data("zvolrok",1))#">
<input type="submit" name="show_details" value="Zobrazit údaje o pacientech">
</form><center>
</CSP:IF>
<br>
<form name="form1" method="post" action="Mortalita.csp">
<input type="hidden" name="zvoldiag" value="">
<input type="submit" name="zpet" value="Zp&#283t na zadání roku">
</form>
</CSP:IF> </CSP:IF>
</td>
</tr></table>

<CSP:ELSE>
<CSP:IF CONDITION=$G(%request.Data("graf",1))["Zobraz"]>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set umrti08procento = (umrti08/1)*100
set umrti09procento = (umrti09/umrti08)*100 - 100
</script>
Grafické zobrazení mortality pacient&#367; v nemocnici v jednotlivých letech
<br> <br>

<br><br>
Tabulka konkrétních hodnot pro jednotlivé roky
<center>
<table class="graf" border="1">
<tr class="printout_first_row"><td width="360"></td><td width="150">2007</td><td
width="150">2008</td><td width="150">2009</td></tr>
<tr><td>po&#269;et pacient&#367;</td><td>
#{umrti07}#</td><td>#{umrti08}#</td><td>#{umrti09}#</td></tr>
<tr><td>zvýšení/snížení oproti p&#345;edchozímu roku</td><td>--</td>
<td> <script LANGUAGE="CACHE" RUNAT="SERVER">
write $FNUMBER(umrti08procento,"",2)
</script> %</td>
<td><script LANGUAGE="CACHE" RUNAT="SERVER">
write $FNUMBER(umrti09procento,"",2)
</script> %</td></tr>
</table></center><br>
<form name="form4" method="post" action="Mortalita.csp">
<input type="hidden" name="zvoldiag" value="">
<input type="submit" name="zpet" value="Zp&#283t na zadání roku">
</form>
</CSP:IF>
</CSP:IF>

<CSP:ELSE><br><br>

```

```

<h3 class="result">Nejste p&#345;ihlášení!<b></b></h3><br>
<a href="DPVelicka.csp"><font color="blue">Návrat na hlavní stránku.</font></a>
</CSP:IF>
</BODY>
</HTML>

```

Soubor vyskytnovotvaru.csp

```

<HTML>
<HEAD>
<TITLE> Cache Server Page </TITLE>
<link rel="stylesheet" type="text/css" href="styly.css">
</HEAD>

<BODY bgcolor="white">
<CSP:IF CONDITION='%session.Data("pristup") = "ok"'>
<SCRIPT LANGUAGE="SQL" NAME="diag">
SELECT *
FROM SQLUser.VDiagnoza
LEFT JOIN SQLUser.VICD10KS ON dgicd10=ICICD10 GROUP by ICICD10 Order by dgicd10
</SCRIPT>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a=0
set b=0
set year=0
set month=0
set day=0
set pomocne=0
</script>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set b = ($ZDate($Horolog,8))
set year = b-10000
set month = b-100
</script>

<h3 class="nadpis">Výpo&#269;et výskytu novotvar&#367;</h3>
<CSP:IF CONDITION='$G(%request.Data("zvoldiagnozu",1))= "">
<center><form name="form1" method="get" action="vyskytnovotvaru.csp">
<h5 class="popis">Zobraz vybranou diagnózu ve zvolenem období:</h5>
&#269;ern&#283; ozna&#269;ené diagnózy - zhodné novotvary<br>
<font color="blue">mod&#345;e ozna&#269;ené diagnózy</font> - nezhodné novotvary<br>
<select name="zvoldiagnozu">
<option style="color: black;" value="0">Zvolte diagnózu</option>
<CSP:WHILE CONDITION=diag.Next()>
<CSP:IF CONDITION='(diag.Data("DGICD10"))[" C"]>
<option style="color: black;" value="#(diag.Data("DGICD10"))#">#(diag.Data("DGICD10"))# -
#(diag.Data("ICPopis"))#</option>
<CSP:ELSE>
<CSP:IF CONDITION='(diag.Data("DGICD10"))[" D"]>
<option style="color: blue;" value="#(diag.Data("DGICD10"))#">#(diag.Data("DGICD10"))# -
#(diag.Data("ICPopis"))#</option>
</CSP:IF></CSP:IF>
</CSP:WHILE>
</select>

<br><br>
<table>
<tr><td><select name="rok">
<option style="color: black;" value="">rok</option>
<option style="color: black;" value="2007">2007</option>

```

```

<option style="color: black;" value="2008">2008</option>
<option style="color: black;" value="2009">2009</option></td>
</select>
<td><select name="mesic">
<option style="color: black;" value="">m&#283;síc</option>
<option style="color: black;" value="01">leden</option>
<option style="color: black;" value="02">unor</option>
<option style="color: black;" value="03">brezen</option>
<option style="color: black;" value="04">duben</option>
<option style="color: black;" value="05">kveten</option>
<option style="color: black;" value="06">cerven</option>
<option style="color: black;" value="07">cervenec</option>
<option style="color: black;" value="08">srpen</option>
<option style="color: black;" value="09">zari</option>
<option style="color: black;" value="10">rijen</option>
<option style="color: black;" value="11">listopad</option>
<option style="color: black;" value="12">prosinec</option></td>
</select>
<td><input type="checkbox" name="rok2" value="vsechny">všechny</td></tr>
</table></center>
<input type="submit" name="Submit" value="Vyhledat"></center>
</form>

```

```

<CSP:ELSE>
<center><table class="printout" border="2">
<h5 class="popis">
Údaje o pacientech se zadanou diagnózou
</h5>
<tr class="printout_first_row">
<td><b> P&#345;ijmení</b></td>
<td><b> Jméno</b></td>
<td align="center"><b>Rodné &#269;íslo</b></td>
<td align="center"><b>Místo narození</b></td>
<td align="center"><b>Národnost</b></td>
<td><b> &#269;íslo diagnózy</b></td>
<td><b> Popis diagnózy</b></td>
<td align="center"><b>Datum</b></td>
</tr>

```

```

<SCRIPT LANGUAGE="SQL" NAME="diagnoz">
SELECT SUBSTRING(DGDIAGNOZA,1,3) AS DGZ, DGSkupinaDiagnoz, DGPoprve, DGICD10, ICICD10,
ICPopis, Aenderungsdatum, geaendert_durch,
SUBSTRING(Aenderungsdatum,1,4) AS rok, SUBSTRING(Aenderungsdatum,5,2) AS mesic,
SUBSTRING(Aenderungsdatum,7,2) AS dden,
SUBSTRING(Aenderungsdatum,1,8) AS datum
FROM SQLUser.VDiagnoza
LEFT JOIN SQLUser.VICD10KS ON dgicd10=ICICD10
LEFT JOIN SQLUser.Diagnosegruppe ON DGSkupinaDiagnoz=Diagnosegruppe
ORDER by ICICD10
</SCRIPT>
<CSP:WHILE CONDITION=diagnoz.Next()>
<SCRIPT LANGUAGE="SQL" NAME="pacient">
SELECT * From SQLUser.VPacient where PAPACIENT=#(diagnoz.Data("DGZ"))#
</SCRIPT>
<CSP:WHILE CONDITION=pacient.Next()>
<CSP:IF CONDITION=(diagnoz.Data("DGICD10"))=$G(%request.Data("zvoldiagnozu",1))>
<CSP:IF CONDITION=(diagnoz.Data("rok"))=$G(%request.Data("rok",1))>
<CSP:IF CONDITION=(diagnoz.Data("mesic"))=$G(%request.Data("mesic",1))>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a = a + 1

```

```

</script>
<tr>
<td width="20%">&nbsp;#(pacient.Data("PAPrijmeni"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PAJmeno"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PACisloRodne"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PANarozeniMisto"))#</td>
<td width="5%">&nbsp;#(pacient.Data("PANarodnost"))#</td>
<td width="10%">#(diagnoz.Data("DGICD10"))#</td>
<td width="40%">#(diagnoz.Data("ICPopis"))#</td>
<td width="10%">#(diagnoz.Data("dden"))#.#(diagnoz.Data("mesic"))#.#(diagnoz.Data("rok"))#</td>
</tr>
</CSP:IF>
</CSP:IF>
</CSP:IF>
<CSP:IF CONDITION=""=$G(%request.Data("rok",1))'>
<CSP:IF CONDITION=""=$G(%request.Data("mesic",1))'>
<CSP:IF CONDITION=$G(%request.Data("rok2",1))="vsechny">
<CSP:IF CONDITION=(diagnoz.Data("DGICD10"))=$G(%request.Data("zvoldiagnozu",1))'>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a = a + 1
</script>
<tr>
<td width="20%">&nbsp;#(pacient.Data("PAPrijmeni"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PAJmeno"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PACisloRodne"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PANarozeniMisto"))#</td>
<td width="5%">&nbsp;#(pacient.Data("PANarodnost"))#</td>
<td width="10%">#(diagnoz.Data("DGICD10"))#</td>
<td width="40%">#(diagnoz.Data("ICPopis"))#</td>
<td width="10%">#(diagnoz.Data("dden"))#.#(diagnoz.Data("mesic"))#.#(diagnoz.Data("rok"))#</td>
</tr>
</CSP:IF>
</CSP:IF>
</CSP:IF>
</CSP:IF>
</CSP:WHILE>
</CSP:WHILE>
</table>Po&#269;et pacient&#367; s danou diagnózou:<font color="red"><b> #(a)# </font></b>
<br><br>
<form name="form1" method="post" action="vyskytnovotvaru.csp">
<input type="hidden" name="zvoldiag" value="">
<input type="submit" name="zpet" value="Zp&#283;t na zadání">
</form></center>
</CSP:IF>

<CSP:ELSE><br><br>
<h3 class="result">Nejste p&#345;ihlášení!<b></b></h3><br>
<a href="DPVelicka.csp"><font color="blue">Návrat na hlavní stránku.</font></a>
</CSP:IF>
</BODY>
</HTML>

```

Soubor hlaseinovotvaru.csp

```

<HTML>
<HEAD>
<TITLE>Cache Server Page </TITLE>
<link rel="stylesheet" type="text/css" href="styly.css">
</HEAD>

```

```

<BODY bgcolor="white">
<CSP:IF CONDITION='%session.Data("pristup") = "ok"'>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a=0
set a1=0
set a2=0
set b=0
set year=0
set month=0
set day=0
set pomocne=0
</script>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set b = ($ZDate($Horolog,8))
set year = b-10000
set month = b-100
</script>
<SCRIPT LANGUAGE="SQL" NAME="diagnoz">
SELECT SUBSTRING(DGDIAGNOZA,1,3) AS DGZ, DGSkupinaDiagnoz, DGPoprve, DGICD10, ICICD10,
ICPopis, Aenderungsdatum, geaendert_durch,
SUBSTRING(Aenderungsdatum,1,4) AS rok, SUBSTRING(Aenderungsdatum,5,2) AS mesic,
SUBSTRING(Aenderungsdatum,7,2) AS dden,
SUBSTRING(Aenderungsdatum,1,8) AS datum
FROM SQLUser.VDiagnoza
LEFT JOIN SQLUser.VICD10KS ON dgicd10=ICICD10
LEFT JOIN SQLUser.Diagnosegruppe ON DGSkupinaDiagnoz=Diagnosegruppe
ORDER by ICICD10
</SCRIPT>

<CSP:IF CONDITION='$G(%request.Data("vyber",1))="">
<CSP:IF CONDITION='$G(%request.Data("special",1))="">
<h3 class="nadpis">Výpočet výskytu novotvarů; z důvodu hlášení</h3>
<form name="form1" method="post" action="hlaseninovotvaru.csp">
<h5 class="popis">Zobraz všechny diagnózy ve zvoleném období:</h5>
<center><table>
<tr><td><input type="radio" name="special" value="poslrok" checked>poslední rok</td>
<td><input type="radio" name="special" value="poslmesic">poslední měsíc</td>
<td><input type="radio" name="special" value="poslden">poslední den</td></tr>
</table>
<br>
<input type="submit" name="Submit" value="Vyhledat"></center>
</form>

<CSP:ELSE>
<center><table class="printout" border="2">
<h3 class="nadpis">Výpočet výskytu novotvarů; z důvodu hlášení</h3>
<h5 class="popis"><center>
<CSP:IF CONDITION='$G(%request.Data("special",1))="poslrok"'>
Údaje o pacientech s nalezenou diagnózou pro období:<br>
#($extract(year,7,8))#.#($extract(year,5,6))#.#($extract(year,1,4))# -
#($extract(b,7,8))#.#($extract(b,5,6))#.#($extract(b,1,4))#
<CSP:ELSEIF CONDITION='$G(%request.Data("special",1))="poslmesic"'>
Údaje o pacientech s nalezenou diagnózou pro období:<br>
#($extract(month,7,8))#.#($extract(month,5,6))#.#($extract(month,1,4))# -
#($extract(b,7,8))#.#($extract(b,5,6))#.#($extract(b,1,4))#
<CSP:ELSEIF CONDITION='$G(%request.Data("special",1))="poslden"'>
Údaje o pacientech s nalezenou diagnózou pro období:<br>
#($extract(b,7,8))#.#($extract(b,5,6))#.#($extract(b,1,4))#
</CSP:IF></center></h5>
<tr class="printout_first_row">

```

```

<td><b> P&#345;ijmení</b></td>
<td><b> Jméno</b></td>
<td align="center"><b>Rodné &#269;íslo</b></td>
<td align="center"><b>Místo narození</b></td>
<td align="center"><b>Národnost</b></td>
<td><b> &#269;íslo diagnózy</b></td>
<td><b> Popis diagnózy</b></td>
<td align="center"><b>Datum</b></td>
</tr>

<CSP:WHILE CONDITION=diagnoz.Next()>
<SCRIPT LANGUAGE="SQL" NAME="pacient">
SELECT *
From SQLUser.VPacient where PAPACIENT=#(diagnoz.Data("DGZ"))#
</SCRIPT>
<CSP:WHILE CONDITION=pacient.Next()>

<CSP:IF CONDITION=$G(%request.Data("special",1))="poslrok">
<CSP:IF CONDITION='(diagnoz.Data("datum"))>year'>
<CSP:IF CONDITION='(diagnoz.Data("DGICD10"))[" C"]>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a1 = a1 + 1
</script>
<tr>
<td width="20%">&nbsp;#(pacient.Data("PAPrijmeni"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PAJmeno"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PACisloRodne"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PANarozeniMisto"))#</td>
<td width="5%">&nbsp;#(pacient.Data("PANarodnost"))#</td>
<td width="10%">#(diagnoz.Data("DGICD10"))#</td>
<td width="40%">#(diagnoz.Data("ICPopis"))#</td>
<td width="10%">#(diagnoz.Data("dden"))#.#(diagnoz.Data("mesic"))#.#(diagnoz.Data("rok"))#</td>
<form name="form6" method="get" action="hlaseninovotvaru.csp">
<input type="hidden" name="vyber" value="#(pacient.Data("PAPACIENT"))#"><input type="hidden"
name="vyber2" value="#(diagnoz.Data("DGICD10"))#">
<td width="5%"><input type="submit" name="Submit2" value="Formulá&#345;"></td></tr>
</form>

</CSP:IF>
<CSP:IF CONDITION='(diagnoz.Data("DGICD10"))[" D"]>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a2 = a2 + 1
</script>
<tr>
<td width="20%">&nbsp;#(pacient.Data("PAPrijmeni"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PAJmeno"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PACisloRodne"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PANarozeniMisto"))#</td>
<td width="5%">&nbsp;#(pacient.Data("PANarodnost"))#</td>
<td width="10%">#(diagnoz.Data("DGICD10"))#</td>
<td width="40%">#(diagnoz.Data("ICPopis"))#</td>
<td width="10%">#(diagnoz.Data("dden"))#.#(diagnoz.Data("mesic"))#.#(diagnoz.Data("rok"))#</td>
<td width="5%"></td>
</CSP:IF>
</CSP:IF>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a = a1 + a2
</script>
</CSP:IF>

```

```

<CSP:IF CONDITION='$G(%request.Data("special",1))="poslmesic">
<CSP:IF CONDITION='(diagnoz.Data("datum"))>month'>
<CSP:IF CONDITION=(diagnoz.Data("DGICD10"))[" C"]>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a1 = a1 + 1
</script>
<tr>
<td width="20%">&nbsp;#(pacient.Data("PAPrijmeni"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PAJmeno"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PACisloRodne"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PANarozeniMisto"))#</td>
<td width="5%">&nbsp;#(pacient.Data("PANarodnost"))#</td>
<td width="10%">#(diagnoz.Data("DGICD10"))#</td>
<td width="40%">#(diagnoz.Data("ICPopis"))#</td>
<td width="10%">#(diagnoz.Data("dden"))#.#(diagnoz.Data("mesic"))#.#(diagnoz.Data("rok"))#</td>
<form name="form6" method="get" action="hlaseninovotvaru.csp">
<input type="hidden" name="vyber" value="#(pacient.Data("PAPACIENT"))#"><input type="hidden"
name="vyber2" value="#(diagnoz.Data("DGICD10"))#">
<td width="5%"><input type="submit" name="Submit2" value="Formulá&#345;"></td></tr>
</form>
</CSP:IF>
<CSP:IF CONDITION='(diagnoz.Data("DGICD10"))[" D"]>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a2 = a2 + 1
</script>
<tr>
<td width="20%">&nbsp;#(pacient.Data("PAPrijmeni"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PAJmeno"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PACisloRodne"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PANarozeniMisto"))#</td>
<td width="5%">&nbsp;#(pacient.Data("PANarodnost"))#</td>
<td width="10%">#(diagnoz.Data("DGICD10"))#</td>
<td width="40%">#(diagnoz.Data("ICPopis"))#</td>
<td width="10%">#(diagnoz.Data("dden"))#.#(diagnoz.Data("mesic"))#.#(diagnoz.Data("rok"))#</td>
<td width="5%"></td>
</CSP:IF>
</CSP:IF>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a = a1 + a2
</script>
</CSP:IF>

<CSP:IF CONDITION='$G(%request.Data("special",1))="posliden">
<CSP:IF CONDITION='(diagnoz.Data("datum"))=b'>
<CSP:IF CONDITION=(diagnoz.Data("DGICD10"))[" C"]>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a1 = a1 + 1
</script>
<tr>
<td width="20%">&nbsp;#(pacient.Data("PAPrijmeni"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PAJmeno"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PACisloRodne"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PANarozeniMisto"))#</td>
<td width="5%">&nbsp;#(pacient.Data("PANarodnost"))#</td>
<td width="10%">#(diagnoz.Data("DGICD10"))#</td>
<td width="40%">#(diagnoz.Data("ICPopis"))#</td>
<td width="10%">#(diagnoz.Data("dden"))#.#(diagnoz.Data("mesic"))#.#(diagnoz.Data("rok"))#</td>
<form name="form6" method="get" action="hlaseninovotvaru.csp">
<input type="hidden" name="vyber" value="#(pacient.Data("PAPACIENT"))#"><input type="hidden"
name="vyber2" value="#(diagnoz.Data("DGICD10"))#">

```

```

<td width="5%"><input type="submit" name="Submit2" value="Formulář"></td></tr>
</form>
</CSP:IF>
<CSP:IF CONDITION=(diagnoz.Data("DGICD10"))[" D"]>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a2 = a2 + 1
</script>
<tr>
<td width="20%">&nbsp;#(pacient.Data("PAPrijmeni"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PAJmeno"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PACisloRodne"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PANarozeniMisto"))#</td>
<td width="5%">&nbsp;#(pacient.Data("PANarodnost"))#</td>
<td width="10%">#(diagnoz.Data("DGICD10"))#</td>
<td width="40%">#(diagnoz.Data("ICPopis"))#</td>
<td width="10%">#(diagnoz.Data("dden"))#.#(diagnoz.Data("mesic"))#.#(diagnoz.Data("rok"))#</td>
<td></td>
</CSP:IF>
</CSP:IF>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a = a1 + a2
</script>
</CSP:IF>
</CSP:WHILE>
</CSP:WHILE>
</table>Počet pacientů s diagnózou novotvaru:<font color="red"><b> #(a)# </font></b>
<br><br>
<form name="form1" method="post" action="hlaseninovotvaru.csp">
<input type="hidden" name="zvoldiag" value="">
<input type="submit" name="zpet" value="Zpět na zadání">
</form></center>
</CSP:IF>

<!-- ***** Formular s vyplnenymi udaji ***** -->

<CSP:else>
<SCRIPT LANGUAGE="SQL" NAME="pacient2">
SELECT * From SQLUser.VPacient where PAPACIENT =#($G(%request.Data("vyber",1)))#
</SCRIPT>
<a href="help.csp" class="help"
onclick="window.open('help.csp','_blank','width=310,height=180,top=256,left=412'); return false">help</a>
<CSP:WHILE CONDITION=pacient2.Next()>
<SCRIPT LANGUAGE="SQL" NAME="pacient3">
SELECT SUBSTRING(PANarozeniDatum,1,4) AS rokn, SUBSTRING(PANarozeniDatum,5,2) AS mesicn,
SUBSTRING(PANarozeniDatum,7,2) AS denn, PAPACIENT
From SQLUser.VPacient where PAPACIENT =#($G(%request.Data("vyber",1)))#
</SCRIPT>

<CSP:WHILE CONDITION=pacient3.Next()>
<SCRIPT LANGUAGE="SQL" NAME="diagnoza2">
SELECT SUBSTRING(DGDIAGNOZA,1,3) AS DGZ2, DGICD10, ICICD10, ICPopis, Datum,
SUBSTRING(Datum,1,4) AS rok2, SUBSTRING(Datum,5,2) AS mesic2, SUBSTRING(Datum,7,2) AS den2
FROM SQLUser.VDiagnoza
LEFT JOIN SQLUser.VICD10KS ON dgicd10=ICICD10
LEFT JOIN SQLUser.Diagnosegruppe ON DGSkupinaDiagnoz=Diagnosegruppe ORDER by ICICD10
</SCRIPT>

<CSP:WHILE CONDITION=diagnoza2.Next()>
<CSP:IF CONDITION=(pacient2.Data("PAPACIENT"))=#G(%request.Data("vyber",1))>
<CSP:IF CONDITION=(diagnoza2.Data("DGZ2"))=#G(%request.Data("vyber",1))>

```



```

<CSP:IF CONDITION='(diagnoza2.Data("DGICD10"))=$G(%request.Data("vyber2",1))'>
<center><form name="form5" method="get" action="tisk.csp" target="blank">
<center><table cellpadding="20"><tr><td> <!-- Tabulka kvuli odsazení od okraju stránky -->
<table class="rakovina_nadpis">
<tr><td><h2><b>Incidence zhoubného novotvaru</b></h2></td>
</TABLE>
<table class="rakovina_nadpis" cellpadding="10">
<tr><td width="200">Rodné &#269;íslo: <TABLE
class="rakovina"><tr><td>#(pacient2.Data("PACisloRodne"))#</tr></td></TABLE></td>
<tr><td width="200">P&#345;ijmení: <TABLE
class="rakovina"><tr><td>#(pacient2.Data("PAPrijmeni"))#</tr></td></TABLE></td>
<tr><td width="200">Jméno: <TABLE
class="rakovina"><tr><td>#(pacient2.Data("PAJmeno"))#</tr></td></TABLE></td>
<tr><td width="200">Místo narození: <TABLE
class="rakovina"><tr><td>#(pacient2.Data("PANarozeniMisto"))#</tr></td></TABLE></td>
<CSP:IF CONDITION='(pacient2.Data("PAPohlavi"))="1">
<tr><td width="30">Pohlaví:<TABLE class="rakovina"><tr><td>žena</tr></td></TABLE></td>
<CSP:ELSE>
<tr><td width="30">Pohlaví:<TABLE class="rakovina"><tr><td>muž</tr></td></TABLE></td>
</CSP:IF></table>

<table class="rakovina_nadpis" cellpadding="10">
<tr><td width="500">Trvalé bydliště&#283;; <TABLE class="rakovina"><tr><td>#(pacient2.Data("PAMesto"))#,
#(pacient2.Data("PAUlice"))#, #(pacient2.Data("PAPsc"))#</tr></td></TABLE></td>
<tr><td width="200">Datum narození: <TABLE
class="rakovina"><tr><td>#(pacient3.Data("denn"))#.#(pacient3.Data("mesicn"))#.#(pacient3.Data("rokn"))#</t
r></td></TABLE></td>
</table>

<table class="rakovina_nadpis" cellpadding="10">
<tr><td width="400">Stav (žijící): <TABLE class="rakovina"><tr><td>1<input type="checkbox" name="y1"
value="checked">v rodin&#283;; 2<input type="checkbox" name="y2" value="checked">osamocen&#283;;
9<input type="checkbox" name="y3" value="checked">neznámo</tr></td></TABLE></td>
<tr><td width="40">Národnost: <TABLE
class="rakovina"><tr><td>#(pacient2.Data("PANarodnost"))#</tr></td></TABLE></td>
</table>

<table class="rakovina_nadpis" cellpadding="10">
<tr><td width="400">Zjiště&#283;no p&#345;i: <TABLE class="rakovina"><tr><td>1<input type="checkbox"
name="z1" value="checked">skríning</td> <td> 3<input type="checkbox" name="z3"
value="checked">klin.manif.</td> <td> 5<input type="checkbox" name="z5" value="checked">jiný
zp&#367;sob</td></tr>
<tr><td>2<input type="checkbox" name="z2" value="checked">prev. prohl. </td> <td> 4<input
type="checkbox" name="z4" value="checked">pitva</td> <td> 9<input type="checkbox" name="z6"
value="checked">neznámo</td></tr> </TABLE></td>
<tr><td width="200">Datum stanovení diagnózy: <TABLE
class="rakovina"><tr><td>#(diagnoza2.Data("den2"))#.#(diagnoza2.Data("mesic2"))#.#(diagnoza2.Data("rok2"
))#</tr></td></TABLE></td>
</table>

<table class="rakovina_nadpis" cellpadding="10">
<tr><td width="170">Kou&#345;ení: <TABLE class="rakovina"><tr><td>1<input type="checkbox"
name="x1" value="checked">ano</td> <td> 3<input type="checkbox" name="x3" value="checked">bývalý
ku&#345;ák</tr></td>
<tr><td>2<input type="checkbox" name="x2" value="checked">ne </td> <td> 9<input type="checkbox"
name="x4" value="checked">neznámo</tr></td></TABLE></td>
<tr><td width="180">Lateralita: <TABLE class="rakovina"><tr><td>1<input type="checkbox" name="x5"
value="checked">vpravo</td> <td> 3<input type="checkbox" name="x7"
value="checked">oboustran&#283;</td> <td> 9<input type="checkbox" name="x9"
value="checked">neznámo</td></tr>

```

```
<tr><td>2<input type="checkbox" name="x6" value="checked">vlevo </td> <td> 4<input type="checkbox"
name="x8" value="checked">odpadá</tr></TABLE></td>
</table>
```

```
<table class="rakovina_nadpis" cellspacing="10">
<td width="380">K dg. vedla tato vyšet&#345;ení: <TABLE class="rakovina"><tr><td>00<input
type="checkbox" name="x10" value="checked">klinicky jasně</td> <td> 04<input type="checkbox"
name="x13" value="checked">cytologie</td><td> 32<input type="checkbox" name="x16"
value="checked">lab.vyš.,markery</td></tr>
<tr><td>01<input type="checkbox" name="x11" value="checked">histologie </td> <td> 08<input
type="checkbox" name="x14" value="checked">RTG,zobr.m.</td><td> 64<input type="checkbox"
name="x17" value="checked">pitva</td></tr>
<tr><td>02<input type="checkbox" name="x12" value="checked">operace </td><td> 16<input
type="checkbox" name="x15" value="checked">endoskopie</td><td> 99<input type="checkbox" name="x18"
value="checked">DCO</td></tr></TABLE></td>
</table>
```

```
<table class="rakovina_nadpis" cellspacing="10">
<td width="600">Diagnóza (slovn&#283;): <TABLE
class="rakovina"><tr><td>#(diagnoza2.Data("ICPopis"))#</tr></td></TABLE></td>
<td width="200">kód MKN-10: <TABLE
class="rakovina"><tr><td>#(diagnoza2.Data("DGICD10"))#</tr></td></TABLE></td>
</table>
```

```
<table class="rakovina_nadpis" cellspacing="10">
<td width="220">Riziková kategorie (trofooblast): <input type="checkbox" name="w1" value="checked"></td>
<td width="280">Sérové nádorové markery (jen u varlat): <input type="checkbox" name="w2"
value="checked"></td>
<td width="300">Onemocn&#283;ní: <TABLE class="rakovina"><tr><td>1<input type="checkbox"
name="w3" value="checked">lokalizovaně</td> <td> 2<input type="checkbox" name="w4"
value="checked">pokro&#269;ilé</td><td> 9<input type="checkbox" name="w5"
value="checked">neznámo</td></tr></TABLE></td>
</table>
```

```
<table class="rakovina_nadpis" cellspacing="10">
<td width="200">Datum hlášení: <TABLE class="rakovina"><tr><td
width="200">#($extract(b,7,8))#.#($extract(b,5,6))#.#($extract(b,1,4))#</td></TABLE></td>
<td width="350">Jméno a podpis hlásícího léka&#345;e: <TABLE class="rakovina"><tr><td><input
type="text" size="100" name="jmeno_lekare" value=""></tr></td></TABLE>
</table> <!--rakovina_nadpis -->
</tr></table> <!-- prvni rakovina_nadpis -->
</TABLE>
</tr></td>
</table>
```

```
<input type="hidden" name="id" value="#"($G(%request.Data("vyber",1)))#>
<input type="hidden" name="kodd" value="#"($G(%request.Data("vyber2",1)))#>
<input type="submit" name="tisk" value="Verze pro tisk">
</form></CSP:IF>
```

```
</CSP:IF>
</CSP:IF>
</CSP:WHILE>
</CSP:WHILE>
</CSP:WHILE>
<form name="form4" method="post" action="hlaseninovotvaru.csp">
<input type="hidden" name="vyber" value="">
<input type="submit" name="zpet" value="Zp&#283;t na zadání">
</form></center>
```

```
</CSP:IF>
<CSP:ELSE><br><br>
```

```

<h3 class="result">Nejste p&#345;ihlášení!<b></b></h3><br>
<a href="DPVelicka.csp"><font color="blue">Návrat na hlavní stránku.</font></a>
</CSP:IF>
</BODY>
</HTML>

```

Soubor TBC.csp

```

<HTML>
<HEAD>
<TITLE>Cache Server Page </TITLE>
<link rel="stylesheet" type="text/css" href="styly.css">
</HEAD>

<BODY bgcolor="white">
<CSP:IF CONDITION='%session.Data("pristup") = "ok"'>

<SCRIPT LANGUAGE="SQL" NAME="diag">
SELECT *
FROM SQLUser.VDiagnoza
LEFT JOIN SQLUser.VICD10KS ON dgicd10=ICICD10 GROUP by ICICD10 Order by dgicd10
</SCRIPT>

<script LANGUAGE="CACHE" RUNAT="SERVER">
set a=0
set b=0
set year=0
set month=0
set day=0
set pomocne=0
</script>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set b = ($ZDate($Horolog,8))
set year = b-10000
set month = b-100
</script>

<h3 class="nadpis">Výpo&#269;et výskytu TBC</h3>
<CSP:IF CONDITION='$G(%request.Data("zvoldiagnozu"),1)="">
<center><form name="form1" method="get" action="TBC.csp">
<h5 class="popis">Zobraz vybranou diagnózu ve zvolenem období:</h5>
<select name="zvoldiagnozu">
<option style="color: black;" value="0">Vyberte diagnozu</option>
<CSP:WHILE CONDITION=diag.Next()>
<CSP:IF CONDITION='(diag.Data("DGICD10"))[" A15"]>
<option style="color: black;" value="#(diag.Data("DGICD10"))#">#(diag.Data("DGICD10"))# -
#(diag.Data("ICPopis"))#</option>
<CSP:ELSE>
<CSP:IF CONDITION='(diag.Data("DGICD10"))[" A16"]>
<option style="color: black;" value="#(diag.Data("DGICD10"))#">#(diag.Data("DGICD10"))# -
#(diag.Data("ICPopis"))#</option>
<CSP:ELSE>
<CSP:IF CONDITION='(diag.Data("DGICD10"))[" A17"]>
<option style="color: black;" value="#(diag.Data("DGICD10"))#">#(diag.Data("DGICD10"))# -
#(diag.Data("ICPopis"))#</option>
<CSP:ELSE>
<CSP:IF CONDITION='(diag.Data("DGICD10"))[" A18"]>
<option style="color: black;" value="#(diag.Data("DGICD10"))#">#(diag.Data("DGICD10"))# -
#(diag.Data("ICPopis"))#</option>
<CSP:ELSE>

```

```

<CSP:IF CONDITION='(diag.Data("DGICD10"))[" A19"]'>
<option style="color: black;" value="#(diag.Data("DGICD10"))#">#(diag.Data("DGICD10"))# -
#(diag.Data("ICPopis"))#</option>
</CSP:IF></CSP:IF></CSP:IF></CSP:IF></CSP:IF>
</CSP:WHILE>
</select>

<br><br>
<table>
<tr><td><select name="rok">
<option style="color: black;" value="">rok</option>
<option style="color: black;" value="2007">2007</option>
<option style="color: black;" value="2008">2008</option>
<option style="color: black;" value="2009">2009</option></td>
</select>
<td><select name="mesic">
<option style="color: black;" value="">m&#283;síc</option>
<option style="color: black;" value="01">leden</option>
<option style="color: black;" value="02">unor</option>
<option style="color: black;" value="03">brezen</option>
<option style="color: black;" value="04">duben</option>
<option style="color: black;" value="05">kveten</option>
<option style="color: black;" value="06">cerven</option>
<option style="color: black;" value="07">cervenec</option>
<option style="color: black;" value="08">srpen</option>
<option style="color: black;" value="09">zari</option>
<option style="color: black;" value="10">rijen</option>
<option style="color: black;" value="11">listopad</option>
<option style="color: black;" value="12">prosinec</option></td>
</select>
<td><input type="checkbox" name="rok2" value="vsechny">všechny</td></tr>
</table></center>
<br>
<input type="submit" name="Submit" value="Vyhledat"></center>
</form>

<CSP:ELSE>
<center><table class="printout" border="2">
<h5 class="popis">
Údaje o pacientech se zadanou diagnózou
</h5>
<tr class="printout_first_row">
<td><b> P&#345;ijmení</b></td>
<td><b> Jméno</b></td>
<td align="center"><b>Rodné &#269;íslo</b></td>
<td align="center"><b>Místo narození</b></td>
<td align="center"><b>Národnost</b></td>
<td><b> &#269;íslo diagnózy</b></td>
<td><b> Popis diagnózy</b></td>
<td align="center"><b>Datum</b></td>
</tr>

<SCRIPT LANGUAGE="SQL" NAME="diagnoz">
SELECT SUBSTRING(DGDIAGNOZA,1,3) AS DGZ, DGSkupinaDiagnoz, DGPoprve, DGICD10, ICICD10,
ICPopis, Aenderungsdatum, geaendert_durch,
SUBSTRING(Aenderungsdatum,1,4) AS rok, SUBSTRING(Aenderungsdatum,5,2) AS mesic,
SUBSTRING(Aenderungsdatum,7,2) AS dden,
SUBSTRING(Aenderungsdatum,1,8) AS datum
FROM SQLUser.VDiagnoza
LEFT JOIN SQLUser.VICD10KS ON dgicd10=ICICD10

```

```

LEFT JOIN SQLUser.Diagnosegruppe ON DGSkupinaDiagnoz=Diagnosegruppe
ORDER by ICICD10
</SCRIPT>
<CSP:WHILE CONDITION=diagnoz.Next()>
<SCRIPT LANGUAGE="SQL" NAME="pacient">
SELECT * From SQLUser.VPacient where PAPACIENT=#(diagnoz.Data("DGZ"))#
</SCRIPT>
<CSP:WHILE CONDITION=pacient.Next()>
<CSP:IF CONDITION=(diagnoz.Data("DGICD10"))=$G(%request.Data("zvoldiagnozu",1))'>
<CSP:IF CONDITION=(diagnoz.Data("rok"))=$G(%request.Data("rok",1))'>
<CSP:IF CONDITION=(diagnoz.Data("mesic"))=$G(%request.Data("mesic",1))'>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a = a + 1
</script>
<tr>
<td width="20%">&nbsp;#(pacient.Data("PAPrijmeni"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PAJmeno"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PACisloRodne"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PANarozeniMisto"))#</td>
<td width="5%">&nbsp;#(pacient.Data("PANarodnost"))#</td>
<td width="10%">#(diagnoz.Data("DGICD10"))#</td>
<td width="40%">#(diagnoz.Data("ICPopis"))#</td>
<td width="10%">#(diagnoz.Data("dden"))#.#(diagnoz.Data("mesic"))#.#(diagnoz.Data("rok"))#</td>
</tr>
</CSP:IF>
</CSP:IF>
</CSP:IF>
<CSP:IF CONDITION=""=$G(%request.Data("rok",1))'>
<CSP:IF CONDITION=""=$G(%request.Data("mesic",1))'>
<CSP:IF CONDITION=$G(%request.Data("rok2",1))="vsechny">
<CSP:IF CONDITION=(diagnoz.Data("DGICD10"))=$G(%request.Data("zvoldiagnozu",1))'>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a = a + 1
</script>
<tr>
<td width="20%">&nbsp;#(pacient.Data("PAPrijmeni"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PAJmeno"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PACisloRodne"))#</td>
<td width="10%">&nbsp;#(pacient.Data("PANarozeniMisto"))#</td>
<td width="5%">&nbsp;#(pacient.Data("PANarodnost"))#</td>
<td width="10%">#(diagnoz.Data("DGICD10"))#</td>
<td width="40%">#(diagnoz.Data("ICPopis"))#</td>
<td width="10%">#(diagnoz.Data("dden"))#.#(diagnoz.Data("mesic"))#.#(diagnoz.Data("rok"))#</td>
</tr>
</CSP:IF>
</CSP:IF>
</CSP:IF>
</CSP:IF>
</CSP:WHILE>
</CSP:WHILE>
</table>Po&#269;et pacient&#367; s danou diagnózou:<font color="red"><b> #(a)# </font></b>
<br><br>
<form name="form1" method="post" action="TBC.csp">
<input type="hidden" name="zvoldiag" value="">
<input type="submit" name="zpet" value="Zp&#283;t na zadání">
</form></center>
</CSP:IF>

<CSP:ELSE><br><br>
<h3 class="result">Nejste p&#345;ihlášení!<b></b></h3><br>

```

```

<a href="DPVelicka.csp"><font color="blue">Návrat na hlavní stránku.</font></a>
</CSP:IF>
</BODY>
</HTML>

```

Soubor leky.csp

```

<HTML>
<HEAD>
<TITLE>Cache Server Page </TITLE>
<link rel="stylesheet" type="text/css" href="styly.css">
</HEAD>

<BODY>
<CSP:IF CONDITION='%session.Data("pristup") = "ok"'>
<h3 class="nadpis">Výpočet spotřeby léků</h3>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a = 0
set b = 0
</script>
<CSP:IF CONDITION='$G(%request.Data("zvolleky"),1)'>
<center><form name="form1" method="post" action="leky.csp">
<select name="zvolleky">
<option style="color: black;" value="0">Vyberte lek</option>
<SCRIPT LANGUAGE="SQL" NAME="lek">
SELECT *
FROM SQLUser.NLeistung
GROUP by Beschreibung Order by Beschreibung
</SCRIPT>
<CSP:WHILE CONDITION=lek.Next()>
<CSP:IF CONDITION=(lek.Data("Leistung"))["MH"]>
<option style="color: black;" value="#(lek.Data("Leistung"))#">#(lek.Data("Beschreibung"))# ,
#(lek.Data("Beschreibung_Englisch"))#
</option>
</CSP:WHILE>
</CSP:IF>
</CSP:WHILE>
</select>
<input type="submit" name="Submit" value="Vyhledat">
</form></center>

<CSP:ELSE>
<SCRIPT LANGUAGE="SQL" NAME="leky2">
SELECT *
FROM SQLUser.ErbrachteNL
</SCRIPT>
<CSP:WHILE CONDITION=leky2.Next()>
<CSP:IF CONDITION='(leky2.Data("Leistung"))=$G(%request.Data("zvolleky"),1)'>

<script LANGUAGE="CACHE" RUNAT="SERVER">
set a = a + (leky2.Data("Menge"))
</script></CSP:IF>
</CSP:WHILE>

<SCRIPT LANGUAGE="SQL" NAME="leky">
SELECT *
FROM SQLUser.NLeistung
</SCRIPT>

<CSP:WHILE CONDITION=leky.Next()>

```

```

<CSP:IF CONDITION='(leky.Data("Leistung"))=$G(%request.Data("zvolleky",1))'>
<h5 class="popis">Název léku:<b><font color="red"> #(leky.Data("Beschreibung"))# ,
#(leky.Data("Beschreibung_Englisch"))#</font></b></h5>
<h5 class="popis">Kód léku:<b><font color="red"> #(leky.Data("Leistung"))#</font></b></h5>
</CSP:IF>
</CSP:WHILE>
<br><h3 class="result">Spot&#345;ebované množství léku: <b> #(a)# ks</b></h3><br>
<center><form name="form2" method="post" action="leky.csp">
<input type="hidden" name="zvolleky" value="">
<input type="submit" name="zpet" value="Zp&#283t na zadání">
</form></center>
</CSP:IF>
<CSP:ELSE><br><br>
<h3 class="result">Nejste p&#345;ihlášení!<b></b></h3><br>
<a href="DPVelicka.csp"><font color="blue">Návrat na hlavní stránku.</font></a>
</CSP:IF>
</BODY>
</HTML>

```

Soubor material.csp

```

<HTML>
<HEAD>
<TITLE>Cache Server Page </TITLE>
<link rel="stylesheet" type="text/css" href="styly.css">
</HEAD>

<BODY bgcolor="white">
<CSP:IF CONDITION=%session.Data("pristup") = "ok">
<h3 class="nadpis">Výpo&#269et spot&#345;eby zdravotnického materiálu</h3>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set a = 0
set b = 0
</script>
<CSP:IF CONDITION=$G(%request.Data("zvolmaterial",1))="">
<center><form name="form1" method="post" action="material.csp">
<select name="zvolmaterial">
<option style="color: black;" value="0">Vyberte material</option>
<SCRIPT LANGUAGE="SQL" NAME="material">
SELECT *
FROM SQLUser.NLeistung
GROUP by Beschreibung Order by Beschreibung
</SCRIPT>

<CSP:WHILE CONDITION=material.Next()>
<CSP:IF CONDITION=(material.Data("Leistung"))["MH">
<CSP:ELSE>
<option style="color: black;" value="#(material.Data("Leistung"))#">#(material.Data("Beschreibung"))# ,
#(material.Data("Beschreibung_Englisch"))#
</option>
</CSP:IF>
</CSP:WHILE>
</select>
<input type="submit" name="Submit" value="Vyhledat">
</form></center>

<CSP:ELSE>
<SCRIPT LANGUAGE="SQL" NAME="material2">
SELECT *
FROM SQLUser.ErbrachteNL

```

```

</SCRIPT>

<CSP:WHILE CONDITION=material2.Next()>
<CSP:IF CONDITION=(material2.Data("Leistung"))=$G(%request.Data("zvolmaterial",1))>

<script LANGUAGE="CACHE" RUNAT="SERVER">
set a = a + (material2.Data("Menge"))
</script></CSP:IF>
</CSP:WHILE>

<SCRIPT LANGUAGE="SQL" NAME="material3">
SELECT *
FROM SQLUser.NLeistung
</SCRIPT>

<CSP:WHILE CONDITION=material3.Next()>
<CSP:IF CONDITION=(material3.Data("Leistung"))=$G(%request.Data("zvolmaterial",1))>
<h5 class="popis">Název materiálu:<b><font color="red"> #(material3.Data("Beschreibung"))# ,
#(material3.Data("Beschreibung_Englisch"))#</font></b></h5>
<h5 class="popis">Kód materiálu:<b><font color="red"> #(material3.Data("Leistung"))#</font></b></h5>
</CSP:IF>
</CSP:WHILE>

<br><h3 class="result">Spotřeba: ebované množství materiálu: <b> #(a)# ks</b></h3><br>
<center><form name="form2" method="post" action="material.csp">
<input type="hidden" name="zvolmaterial" value="">
<input type="submit" name="zpet" value="Zpět na zadání">
</form></center>
</CSP:IF>
<CSP:ELSE><br><br>
<h3 class="result">Nejste přihlášení!<b></b></h3><br>
<a href="DPVelicka.csp"><font color="blue">Návrat na hlavní stránku.</font></a>
</CSP:IF>
</BODY>
</HTML>

```

Soubor tisk.csp

```

<HTML>
<HEAD>
<TITLE>Incidence zhoubného novotvaru - tisk </TITLE>
<link rel="stylesheet" type="text/css" href="styly.css">
</HEAD>

<BODY>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set f=($G(%request.Data("id",1)))
set e=($G(%request.Data("kodd",1)))
</SCRIPT>
<script LANGUAGE="CACHE" RUNAT="SERVER">
set b = ($ZDate($Horolog,8))
</script>
<SCRIPT LANGUAGE="SQL" NAME="formular1">
SELECT SUBSTRING(PANarozeniDatum,1,4) AS rokn, SUBSTRING(PANarozeniDatum,5,2) AS mesicn,
SUBSTRING(PANarozeniDatum,7,2) AS denn, PAPACIENT,
PACisloRodne, PAPrijmeni, PANarozeniMisto, PAMesto, PANarodnost, PAJmeno, PAPohlavi, PAUlice,
PAPsc
From SQLUser.VPacient where PAPACIENT =#(f)#
</SCRIPT>
<CSP:WHILE CONDITION=formular1.Next()>

```



```

<SCRIPT LANGUAGE="SQL" NAME="formular2">
SELECT SUBSTRING(DGDIAGNOZA,1,3) AS DGZ2, DGICD10, ICICD10, ICPopis, Datum,
SUBSTRING(Datum,1,4) AS rok2, SUBSTRING(Datum,5,2) AS mesic2, SUBSTRING(Datum,7,2) AS den2
FROM SQLUser.VDiagnoza
LEFT JOIN SQLUser.VICD10KS ON dgicd10=ICICD10
LEFT JOIN SQLUser.Diagnosegruppe ON DGSkupinaDiagnoz=Diagnosegruppe ORDER by ICICD10
</SCRIPT>
<CSP:WHILE CONDITION=formular2.Next()>
<CSP:IF CONDITION=(formular2.Data("DGZ2"))=f'>
<CSP:IF CONDITION=(formular2.Data("DGICD10"))=e'>
<center><table cellpadding="20"><tr><td> <!-- Tabulka kvuli odsazení od okraju stránky -->
<table class="rakovina_nadpis">
<tr><td><b>Incidence zhoubného novotvaru</b></td></tr></table>
<table class="rakovina_nadpis" cellpadding="10">
<tr><td width="200">Rodné číslo: <TABLE
class="rakovina"><tr><td>#(formular1.Data("PACisloRodne"))#</tr></table></td>
<tr><td width="200">Příjmení: <TABLE
class="rakovina"><tr><td>#(formular1.Data("PAPrijmeni"))#</tr></table></td>
<tr><td width="200">Jméno: <TABLE
class="rakovina"><tr><td>#(formular1.Data("PAJmeno"))#</tr></table></td>
<tr><td width="200">Místo narození: <TABLE
class="rakovina"><tr><td>#(formular1.Data("PANarozeniMisto"))#</tr></table></td>
<CSP:IF CONDITION=(formular1.Data("PAPohlavi"))="1">
<tr><td width="30">Pohlaví:<TABLE class="rakovina"><tr><td>žena</tr></table></td>
<CSP:ELSE>
<tr><td width="30">Pohlaví:<TABLE class="rakovina"><tr><td>muž</tr></table></td>
</CSP:IF></table>
<table class="rakovina_nadpis" cellpadding="10">
<tr><td width="500">Trvalé bydliště: <TABLE class="rakovina"><tr><td>#(formular1.Data("PAMesto"))#,
#(formular1.Data("PAUlice"))#, #(formular1.Data("PAPsc"))#</tr></table></td>
<tr><td width="200">Datum narození: <TABLE
class="rakovina"><tr><td>#(formular1.Data("denn"))#.#(formular1.Data("mesic"))#.#(formular1.Data("rokn"))
#</tr></table></td>
</table>

<table class="rakovina_nadpis" cellpadding="10">
<tr><td width="400">Stav (žijící): <TABLE class="rakovina"><tr><td>1<input type="checkbox"
#($G(%request.Data("y1",1)))# name="y1" value="checked">v rodině; 2<input type="checkbox"
#($G(%request.Data("y2",1)))# name="y2" value="checked">osamocen; 9<input type="checkbox"
#($G(%request.Data("y3",1)))# name="y3" value="checked">neznámo</tr></table></td>
<tr><td width="40">Národnost: <TABLE
class="rakovina"><tr><td>#(formular1.Data("PANarodnost"))#</tr></table></td>
</table>

<table class="rakovina_nadpis" cellpadding="10">
<tr><td width="400">Zjištěno příjmení: <TABLE class="rakovina"><tr><td>1<input type="checkbox"
#($G(%request.Data("z1",1)))# name="z1" value="checked">skrining</td> <td> 3<input type="checkbox"
#($G(%request.Data("z3",1)))# name="z3" value="checked">klin.manif.</td> <td> 5<input type="checkbox"
#($G(%request.Data("z5",1)))# name="z5" value="checked">jiný způsob; </tr>
<tr><td>2<input type="checkbox" name="z2" #($G(%request.Data("z2",1)))# value="checked">prev. prohl.
</td> <td> 4<input type="checkbox" #($G(%request.Data("z4",1)))# name="z4"
value="checked">pitva</td> <td> 9<input type="checkbox" #($G(%request.Data("z6",1)))# name="z6"
value="checked">neznámo</td></tr> </table></td>
<tr><td width="200">Datum stanovení diagnózy: <TABLE
class="rakovina"><tr><td>#(formular2.Data("den2"))#.#(formular2.Data("mesic2"))#.#(formular2.Data("rok2"))
#</tr></table></td>
</table>

<table class="rakovina_nadpis" cellpadding="10">

```

```

<td width="200">kou&#345;ení: <TABLE class="rakovina"><tr><td>1<input type="checkbox"
#($G(%request.Data("x1",1)))# name="x1" value="checked">ano</td> <td> 3<input type="checkbox"
name="x3" #($G(%request.Data("x3",1)))# value="checked">bývalý ku&#345;ák</tr></td>
<tr><td>2<input type="checkbox" #($G(%request.Data("x2",1)))# name="x2" value="checked">ne </td> <td>
9<input type="checkbox" name="x4" #($G(%request.Data("x4",1)))#
value="checked">neznámo</tr></td></TABLE></td>
<td width="180">Lateralita: <TABLE class="rakovina"><tr><td>1<input type="checkbox"
#($G(%request.Data("x5",1)))# name="x5" value="checked">vpravo</td> <td> 3<input type="checkbox"
name="x7" #($G(%request.Data("x7",1)))# value="checked">oboustran&#283;</td><td> 9<input
type="checkbox" #($G(%request.Data("x9",1)))# name="x9" value="checked">neznámo</td></tr>
<tr><td>2<input type="checkbox" #($G(%request.Data("x6",1)))# name="x6" value="checked">vlevo </td>
<td> 4<input type="checkbox" name="x8" #($G(%request.Data("x8",1)))#
value="checked">odpadá</tr></td></TABLE></td>
</table>

```

```

<table class="rakovina_nadpis" cellspacing="10">
<td width="380">K dg. vedla tato vyšet&#345;ení: <TABLE class="rakovina"><tr><td>00<input
type="checkbox" #($G(%request.Data("x10",1)))# name="x10" value="checked">klinicky jasn&#283;</td> <td>
04<input type="checkbox" #($G(%request.Data("x13",1)))# name="x13" value="checked">cytologie</td><td>
32<input type="checkbox" #($G(%request.Data("x16",1)))# name="x16"
value="checked">lab.vyš.,markery</td></tr>
<tr><td>01<input type="checkbox" #($G(%request.Data("x11",1)))# name="x11" value="checked">histologie
</td> <td> 08<input type="checkbox" #($G(%request.Data("x14",1)))# name="x14"
value="checked">RTG,zobr.m.</td><td> 64<input type="checkbox" #($G(%request.Data("x17",1)))#
name="x17" value="checked">pitva</td></tr>
<tr><td>02<input type="checkbox" #($G(%request.Data("x12",1)))# name="x12" value="checked">operace
</td><td> 16<input type="checkbox" #($G(%request.Data("x15",1)))# name="x15"
value="checked">endoskopie</td><td> 99<input type="checkbox" #($G(%request.Data("x18",1)))#
name="x18" value="checked">DCO</td></tr></TABLE></td>
</table>

```

```

<table class="rakovina_nadpis" cellspacing="10">
<td width="600">Diagn&#283;za (slovn&#283;): <TABLE
class="rakovina"><tr><td>#(formular2.Data("ICPopis"))#</tr></td></TABLE></td>
<td width="200">k&#282;d MKN-10: <TABLE
class="rakovina"><tr><td>#(formular2.Data("DGICD10"))#</tr></td></TABLE></td>
</table>

```

```

<table class="rakovina_nadpis" cellspacing="10">
<td width="220">Rizik&#282;ov&#282;a kategorie (trofooblast): <input type="checkbox" #($G(%request.Data("w1",1)))#
name="w1" value="checked"></td>
<td width="220">S&#282;erov&#282;e n&#282;adorov&#282;e markery (jen u varlat): <input type="checkbox"
#($G(%request.Data("w2",1)))# name="w2" value="checked"></td>
<td width="350">Onemocn&#283;n&#283;: <TABLE class="rakovina"><tr><td>1<input type="checkbox"
#($G(%request.Data("w3",1)))# name="w3" value="checked">lokalizovan&#283;</td> <td> 2<input
type="checkbox" #($G(%request.Data("w4",1)))# name="w4" value="checked">pokro&#269;il&#283;</td><td>
9<input type="checkbox" #($G(%request.Data("w5",1)))# name="w5"
value="checked">nezn&#282;amo</td></tr></TABLE></td>
</table>

```

```

<table class="rakovina_nadpis" cellspacing="10">
<td width="150">Datum hl&#282;šení: <TABLE
class="rakovina"><tr><td>#($extract(b,7,8))#.#($extract(b,5,6))#.#($extract(b,1,4))#</tr></td></TABLE></td>
<td width="400">Jm&#282;eno a podpis hl&#282;si&#283;eho l&#282;ka&#345;e: <TABLE class="rakovina"><tr><td>
#($G(%request.Data("jmeno_lekare",1)))#</tr></td></TABLE>
</table> <!--rakovina_nadpis -->
</tr></table> <!-- prvni rakovina_nadpis -->
</TABLE>
</tr></td>
</table></center>

```

```
</CSP:IF>  
</CSP:IF>  
</CSP:WHILE>  
</CSP:WHILE>  
</BODY>  
</HTML>
```

Příloha č.2 – Protokol „Incidence a léčba zhoubného novotvaru“

Incidence a léčba zhoubného novotvaru		Razítka zdravot. pracoviště, které diagnostikuje novotvar	
Vypíšte trojčíslo silně orámovanou část (? lze zaškrtnout více možností)		Pořadí ZN <input type="text"/>	Evid. č. ZN <input type="text"/>
Rodné číslo <input type="text"/>		Příjmení a jméno <input type="text"/>	
Trvalé bydliště: Obec, okres, PSČ <input type="text"/>		Datum narození <input type="text"/>	
Sociální postavení: 1 <input type="checkbox"/> zaměstnanec 2 <input type="checkbox"/> samost. pracující 3 <input type="checkbox"/> nepracující 4 <input type="checkbox"/> nezaměstnaný 5 <input type="checkbox"/> bezdomovec 9 <input type="checkbox"/> neznámo		IČ <input type="text"/>	PČZ <input type="text"/>
Stav (žijící) 1 <input type="checkbox"/> v rodině 2 <input type="checkbox"/> osamocené 9 <input type="checkbox"/> neznámo	Země původu (slovně) <input type="text"/>		kód <input type="text"/>
Zjištěno při: 1 <input type="checkbox"/> skrining 3 <input type="checkbox"/> klin. manif. 5 <input type="checkbox"/> jiný způsob 2 <input type="checkbox"/> prev. prohl. 4 <input type="checkbox"/> pitva 9 <input type="checkbox"/> neznámo	Datum 1. návštěvy lékaře <input type="text"/>		Datum stanovení diagnózy <input type="text"/>
Kouření: 1 <input type="checkbox"/> ano 3 <input type="checkbox"/> bývalý kuřák 2 <input type="checkbox"/> ne 9 <input type="checkbox"/> neznámo	Laterality: 1 <input type="checkbox"/> vpravo 2 <input type="checkbox"/> vlevo 3 <input type="checkbox"/> oboustranné 4 <input type="checkbox"/> odpadá 9 <input type="checkbox"/> neznámo	K dg. vedla tato vyšetř.? ¹ 02 <input type="checkbox"/> operace 16 <input type="checkbox"/> endoskopie 00 <input type="checkbox"/> klin. jasné 04 <input type="checkbox"/> cytologie 32 <input type="checkbox"/> lab. vys., markery 01 <input type="checkbox"/> histologie 08 <input type="checkbox"/> RTG, zobr. m. 64 <input type="checkbox"/> pitva 99 <input type="checkbox"/> DCO	
Diagnóza (slovně) <input type="text"/>		kód MKN-10 <input type="text"/>	
Morfologie hist. <input type="checkbox"/> cyt. <input type="checkbox"/>	kód topograf. C <input type="text"/>		
TNM T <input type="text"/> mm <input type="text"/> N <input type="text"/> M <input type="text"/>	pTNM y <input type="text"/> pT <input type="text"/> mm <input type="text"/>	pN <input type="text"/>	pN vyš. <input type="text"/>
Riziková kategorie (trofoblast) <input type="checkbox"/>	Sérové nádorové markery (jen u varlat) <input type="checkbox"/>	Onemocnění	
Klinické stadium (0,I,II,III,IV) 6 metastázy u nezn. prim. 7 neuvádí se 9 neznámo	1 <input type="checkbox"/> lokalizované 2 <input type="checkbox"/> pokročilé 9 <input type="checkbox"/> neznámo		
Praktický lékař, u kterého je pacient registrován <input type="text"/>	Datum hlášení <input type="text"/>		
Léčebný postup navržený pracovištěm zodpovědným za léčbu: <input type="text"/>		Razítka zdravot. pracoviště zodpovědného za léčbu	
Datum převzetí do péče <input type="text"/>		IČ <input type="text"/>	
Jméno a podpis hlásícího lékaře <input type="text"/>		Datum hlášení <input type="text"/>	
Operace: operační výkon (slovně) zdr. zařízení a oddělení <input type="text"/>		Datum operace <input type="text"/>	
Druh operace 1 <input type="checkbox"/> extirpace nádoru 2 <input type="checkbox"/> odstranění orgánu s nádorem 3 <input type="checkbox"/> radik. operace s odstr. regionálních uzlin 4 <input type="checkbox"/> radik. odstr. uzlin bez zásahu na prim. nádor 7 <input type="checkbox"/> pokus a/nebo odlehčovací operace 9 <input type="checkbox"/> neznámo	0 <input type="checkbox"/> neoperován		
Odstranění nádoru 0 <input type="checkbox"/> R0 žádný reziduální nádor 1 <input type="checkbox"/> R1 mikroskop. rezid. nádor 2 <input type="checkbox"/> R2 makroskop. rezid. nádor 3 <input type="checkbox"/> nádor ponechán 9 <input type="checkbox"/> neznámo	Endoskopická operace 1 <input type="checkbox"/> ano 2 <input type="checkbox"/> ne 9 <input type="checkbox"/> neznámo		
Radioterapie (slovně) zdr. zařízení a oddělení <input type="text"/>		Datum zahájení radioterapie <input type="text"/>	
Druh záření ¹ 01 <input type="checkbox"/> RTG terapie 02 <input type="checkbox"/> teleterapie Co 60, Cs 137, Gama nůž 04 <input type="checkbox"/> terapie korpuskulárním zářením 08 <input type="checkbox"/> lineární urychlovač 16 <input type="checkbox"/> brachyterapie 32 <input type="checkbox"/> terapie otevřeným zářením 00 <input type="checkbox"/> neozařován	kód <input type="text"/>		
Forma léčby 1 <input type="checkbox"/> předoperační 2 <input type="checkbox"/> pooperační 3 <input type="checkbox"/> před i pooperační 4 <input type="checkbox"/> samostatná (bez operace) 9 <input type="checkbox"/> neznámo	Konkomitance s chemoterapií 1 <input type="checkbox"/> ano 2 <input type="checkbox"/> ne		
Chemoterapie (slovně) zdr. zařízení a oddělení <input type="text"/>		Datum zahájení chemoterapie <input type="text"/>	
Forma léčby 1 <input type="checkbox"/> předoperační 2 <input type="checkbox"/> pooperační 3 <input type="checkbox"/> před i pooperační 4 <input type="checkbox"/> samostatná (bez operace) 9 <input type="checkbox"/> neznámo 0 <input type="checkbox"/> nepodána	Typ léčby ¹ 01 <input type="checkbox"/> systémová 02 <input type="checkbox"/> regionální 08 <input type="checkbox"/> lokální		
Hormonální léčba (slovně) zdr. zařízení a oddělení <input type="text"/>		Datum zahájení hormonální léčby <input type="text"/>	
Druh léčby ¹ 01 <input type="checkbox"/> chirurgická 02 <input type="checkbox"/> medikamentózní 04 <input type="checkbox"/> radiční 00 <input type="checkbox"/> neprovedena		kód <input type="text"/>	
Jiná léčba (slovně) zdr. zařízení a oddělení <input type="text"/>		Datum zahájení jiné léčby <input type="text"/>	
Druh léčby ¹ 01 <input type="checkbox"/> imunoterapie 02 <input type="checkbox"/> hypertermie 04 <input type="checkbox"/> kryoterapie 08 <input type="checkbox"/> transplantace kostní dřeně 16 <input type="checkbox"/> fotodynamická 32 <input type="checkbox"/> regulační terapie 64 <input type="checkbox"/> jiná 00 <input type="checkbox"/> neprovedeno	kód <input type="text"/>		
Protinád. léčba nepodána pro 1 <input type="checkbox"/> stav nevyžaduje 2 <input type="checkbox"/> místní rozsah 3 <input type="checkbox"/> generalizaci 4 <input type="checkbox"/> kontraind. nes. s nád. 5 <input type="checkbox"/> odmítnutí 6 <input type="checkbox"/> úmrtí 9 <input type="checkbox"/> neznámo			
Příčina smrti z Listu o prohlídce mrtvého (MKN-10) Ia <input type="text"/>		Oprava Ia <input type="text"/>	
Datum úmrtí <input type="text"/>		Pitva 1 <input type="checkbox"/> ano 2 <input type="checkbox"/> ne Předchozí novotvary: <input type="text"/>	
Počet novotvarů u této nemocné <input type="text"/>		Zařízení dispensární péče IČ <input type="text"/>	
		PČZ <input type="text"/>	
		PČDP <input type="text"/>	
		odd. <input type="text"/>	

Tisk: KOVARIK, Č. Trebová
Dodávka DITIS s.r.o., PS 31, Odrfovcova 24, 562 06 Ústí n. L., tel.: 465 524 027, fax: 465 524 701, e-mail: ditis@ditis.cz, http://www.ditis.cz

verze 2006/02

Příloha č.3 – Protokol „Incidence zhoubného novotvaru“

Incidence zhoubného novotvaru				
Rodné číslo: 8661041356	Příjmení: Nováková	Jméno: Ilda	Místo narození: Brno	Pohlaví: žena
Trvalé bydliště: Hostěradice, Hostěradice, 671 71			Datum narození: 04.11.1986	
Stav (žijící): 1 <input checked="" type="checkbox"/> v rodině 2 <input type="checkbox"/> osamoceně 9 <input type="checkbox"/> neznámo				Národnost: CZ
Zjištěno při: 1 <input type="checkbox"/> skrining 3 <input type="checkbox"/> klin.manif. 5 <input checked="" type="checkbox"/> jiný způsob 2 <input type="checkbox"/> prev. prohl. 4 <input type="checkbox"/> pitva 9 <input type="checkbox"/> neznámo			Datum stanovení diagnózy: 08.05.2009	
kouření: 1 <input type="checkbox"/> ano 3 <input type="checkbox"/> bývalý kuřák 2 <input checked="" type="checkbox"/> ne 9 <input type="checkbox"/> neznámo		Lateralita: 1 <input type="checkbox"/> vpravo 3 <input type="checkbox"/> oboustraně 9 <input checked="" type="checkbox"/> neznámo 2 <input type="checkbox"/> vlevo 4 <input type="checkbox"/> odpadá		
K dg. vedla tato vyšetření:				
00 <input type="checkbox"/> klinicky jasné		04 <input type="checkbox"/> cytologie		32 <input type="checkbox"/> lab.vyš. ,markery
01 <input type="checkbox"/> histologie		08 <input checked="" type="checkbox"/> RTG,zobr.m.		64 <input type="checkbox"/> pitva
02 <input type="checkbox"/> operace		16 <input type="checkbox"/> endoskopie		99 <input type="checkbox"/> DCO
Diagnóza (slovně): ZN- sestupný tračník - colon descendens			kód MKN-10: C18.6	
Riziková kategorie (trofooblast): <input checked="" type="checkbox"/>		Sérové nádorové markery (jen u varlat): <input type="checkbox"/>		Onemocnění: 1 <input checked="" type="checkbox"/> lokalizované 2 <input type="checkbox"/> pokročilé 9 <input type="checkbox"/> neznámo
Datum hlášení: 17.05.2009		Jméno a podpis hlásícího lékaře: MUDr. Karel Hlaváček		