

Univerzita Pardubice
Fakulta ekonomicko-správní

Implementace nového informačního systému: praktický příklad

Petra Franková

Bakalářská práce

2011

ZADÁNÍ BAKALÁŘSKÉ PRÁCE
(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Petra FRANKOVÁ**
Osobní číslo: **E08932**
Studijní program: **B6209 Systémové inženýrství a informatika**
Studijní obor: **Regionální a informační management**
Název tématu: **Implementace nového informačního systému**
Zadávací katedra: **Ústav systémového inženýrství a informatiky**

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

Životního cyklu informačního systému
Popis stávajícího stavu IS u vybraného podniku
Návrh a průběh implementace nového IS
Zhodnocení

Rozsah grafických prací:

Rozsah pracovní zprávy:

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

Seznam odborné literatury:

GÁLA, L., POUR, J., ŠEDIVÁ, Z. Podniková informatika. Praha : Grada Publishing, a.s., 2009. ISBN 978-80-247-2615-1

TVRDÍKOVÁ, M. Zavádění a inovace informačních systémů ve firmách. Praha : Grada Publishing, a.s., 2000. ISBN 80-7169-703-6

VRANA, I. RICHTA, K. Zásady a postupy zavádění podnikových informačních systémů: praktická příručka pro podnikové manažery. Praha : Grada Publishing, a.s., 2005. ISBN 80-247-1103-6

Internetové zdroje


Vedoucí bakalářské práce:


Ing. Hana Jonášová, Ph.D.

Ústav systémového inženýrství a informatiky

Datum zadání bakalářské práce: **4. října 2010**

Termín odevzdání bakalářské práce: **30. dubna 2011**


doc. Ing. Renáta Myšková, Ph.D.

děkanka

L.S.


doc. Ing. Jiří Křupka, Ph.D.

vedoucí ústavu

V Pardubicích dne 4. října 2010

Prohlášení

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

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

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 30. 4. 2011

Petra Franková

Poděkování

Děkuji vedoucí bakalářské práce Ing. Haně Jonášové, Ph.D. za všechny poskytnuté rady, informace a připomínky při vypracování bakalářské práce.

Zároveň bych chtěla ještě poděkovat i mé rodině, přátelům a kolegům za jejich pomoc, trpělivost, pochopení a podporu během celého studia.

Anotace

Bakalářská práce je zaměřena na charakteristiku životního cyklu informačního systému včetně popisu zaváděného informačního systému, prostředí podniku a jeho stávajícího stavu. Práce dále popisuje návrh a samotný průběh implementace informačního systému VETERIS v tomto podniku. Snahou této bakalářské práce je tedy co nejvíce ozřejmit studentům tuto problematiku a poukázat i na chyby a problémy, které v praktickém reálném prostředí mohou nastat.

Klíčová slova

Informační systém, životní cyklus, implementace, pilotní zavádění, souběžné zavádění, veterinární služba.

Title

Implementation of New Information System: practical example

Annotation

The bachelor's thesis is focused on the characterisation of the information system life cycle and includes the description of the information cycle that is being introduced, the enterprise environment and its current conditions. The thesis further on describes a suggestion and the progress of the information system VETERIS implementation in the enterprise. The bachelor's thesis aims to elucidate this issue to students and point out mistakes and problems that may occur in the virtual true environment.

Key words

Information system, life cycle, implementation, pilot implementation, concurrent implementation, vet service.

OBSAH:

1	ÚVOD	11
2	INFORMAČNÍ SYSTÉM	12
2.1	Kvalita informačního systému	12
2.2	Bezpečnost informačního systému	13
2.3	Způsoby řešení informačního systému	13
3	ŽIVOTNÍ CYKLUS INFORMAČNÍHO SYSTÉMU.....	15
3.1	Metodiky.....	15
3.2	Standardy	16
3.3	Základní modely životního cyklu	16
3.4	Etapy životního cyklu	17
3.4.1	Plánování a příprava, stanovení cílů	18
3.4.2	Analýza a návrh	18
3.4.3	Tvorba softwaru, testování.....	19
3.4.4	Implementace do provozu, migrace	20
3.4.5	Rutinní provoz a údržba.....	23
3.4.6	Rozvoj a optimalizace, nový projekt	24
4	POPIS STÁVAJÍCÍHO STAVU A NÁVRH VÝBĚRU INFORMAČNÍHO SYSTÉMU	26
4.1	Veterinární služba Armády České republiky.....	26
4.1.1	Obecná veterinární péče.....	26
4.1.2	Odborná veterinární péče	26
4.1.3	Výkon státní správy	27
4.2	Informační systém veterinární služby Armády České republiky	27
4.2.1	Způsob řešení pořízení nového veterinárního informačního systému	28
4.2.2	Návrh aplikace VETERIS.....	29
4.2.3	Požadavky na hardwarovou a softwarovou infrastrukturu	30
5	POPIS NÁVRHU IMPLEMENTACE APLIKACE VETERIS	31

5.1	Přípravná a zahajovací fáze	31
5.2	Prováděcí fáze.....	31
5.3	Ukončení implementace a předání do provozu	32
6	POPIS REALIZOVANÉ IMPLEMENTACE APLIKACE VETERIS	33
6.1	Fáze přípravná a zahajovací (WBS 1.1)	33
6.1.1	Prověření zabezpečení hardwarové a softwarové infrastruktury (WBS 1.1.1)..	33
6.1.2	Příprava dat (WBS 1.1.2).....	33
6.1.3	Dodávka databázového serveru a notebooků (WBS 1.1.3 a 1.1.4).....	33
6.1.4	Zhodnocení připravenosti zadavatele a dodavatele (WBS 1.1.5 a 1.1.6)	33
6.1.5	Dodávka prototypu aplikace VETERIS (WBS 1.1.7)	34
6.2	Fáze prováděcí (WBS 1.2).....	34
6.2.1	Implementace na pilotní pracoviště – jedna pracovní stanice (WBS 1.2.1)	34
6.2.2	Implementace na pilotní pracoviště – více pracovních stanic (WBS 1.2.2)	35
6.2.3	Migrace dat na databázový server (WBS 1.2.3)	36
6.2.4	Centralizace dat (WBS 1.2.4)	38
6.2.5	Rozšíření na spádová pracoviště (WBS 1.2.5).....	38
6.2.6	Finální optimalizace (WBS 1.2.6)	39
6.3	Fáze ukončení implementace a předání do provozu (WBS 1.3)	39
7	ZHODNOCENÍ IMPLEMENTACE	41
8	ZÁVĚR	44
9	POUŽITÉ ZDROJE	46

Seznam používaných symbolů a zkratek

AČR	Armáda České republiky
AVG	Antivirový systém firmy AVG Technologies
BPMN	Business Process Modeling Notation (Grafická notace určená k popisu podnikových procesů)
CACHE	Objektová databáze firmy InterSystems Česká republika
CADS	Celoarmádní datová síť
CD	Compact Disc (Kompaktní disk)
CPU	Central Processing Unit (Procesor)
ČSN	Česká státní norma
DCOM	Distributed Component Object Model (Protokol definující interakci mezi komponentami firmy Microsoft)
DROGIS	Podsystem informačního systému zdravotnické služby pro sledování drogové závislosti
EPC	Event-driven Process Chain (Diagram procesu řízeného událostmi)
eEPC	extended Event-driven Process Chain (Rozšířený diagram procesu řízeného událostmi)
FoxPro	System řízení báze dat
HW	Hardware
HYGEPID	Podsystem informačního systému zdravotnické služby pro sledování hygienicko-epidemiologické situace
IDEF	Integration DEFinition (Kontextový diagram funkcí)
IEC	Mezinárodní elektrotechnická komise
IS	Informační systém
ISA	Internet Security and Acceleration (Internetové zabezpečení a zrychlení)
ISO	Mezinárodní organizace pro normalizaci
ISVS	Informační systémy veřejné správy
IT	Informační technologie
ITIL	Information Technology Infrastructure Library (Soubor konceptů a postupů pro lepší využití informačních technologií)
LCD	Liquid Crystal Display (Displej z tekutých krystalů)
MIDS	Multidimensional Development of Information System (Metodický základ systémové integrace)
MO	Ministerstvo obrany

MS DOS	Microsoft Disk Operating System (Operační systém firmy Microsoft)
OMT	Object Modeling Technique (Objektově orientované metodiky a technologie)
OOAD	Object-Oriented Analysis & Design (Objektově orientovaná analýza a návrh)
OOSE	Object-Oriented Software Engineering (Objektově orientované softwarové inženýrství)
PC DENT/V	Podsystem informačního systému zdravotnické služby pro zubní lékaře
PC DOKTOR/V	Podsystem informačního systému zdravotnické služby pro praktické lékaře
PRIMIS	Podsystem informačního systému zdravotnické služby pro primární lékařskou péči
RAID	Redundant Array of Independent Disks (Vícenásobné diskové pole nezávislých disků)
RAM	Random-Access Memory (Paměť s přímým přístupem)
SDM	System Development Methodology (Metodika vývoje informačního systému)
SSADM	Structured Systems Analysis and Design Methodology (Strukturovaná metodika vývoje informačního systému)
SW	Software
UML	Unified Modeling Language (Unifikovaný modelovací jazyk)
VETERIS	Informační systém veterinární služby
WBS	Work Breakdown Structure (Struktura rozpisu práce)
Windows XP PRO	Operační systém firmy Microsoft
ZDRAVIS	Informační systém zdravotnické služby
ŽC	Životní cyklus

Seznam obrázků

Obrázek 1 - Životní cyklus IS (zdroj: upraveno na základě [4]).....	17
Obrázek 2 - Souběžné zavádění IS (zdroj: upraveno na základě [14]).....	21
Obrázek 3 - Pilotní zavádění IS (zdroj: upraveno na základě [14]).....	22
Obrázek 4 - Postupné zavádění IS (zdroj: upraveno na základě [14]).....	22
Obrázek 5 - Nárazové zavedení IS (zdroj: upraveno na základě [14])	23
Obrázek 6 - Úrovně veterinární služby AČR (zdroj: vlastní)	27
Obrázek 7 - Schéma ZDRAVIS (zdroj: vlastní)	28
Obrázek 8 - Propojení pracovních stanic peer-to-peer (zdroj: vlastní)	36

Seznam příloh

Příloha A – Modely životního cyklu informačního systému

Příloha B – Návrh plánu implementace

Příloha C – Požadavky na informační systém, popis databázového serveru

Příloha D – Popis aplikace VETERIS

1 ÚVOD

Slovo systém se v dnešní době nadužívá ve všech pádech, tvarech i významech, často i mimo jeho faktický obsah. Nejčastěji užívaným spojením je však Informační systém (dále jen IS). Rozvoj informačních systémů zasahuje do všech oborů, využívají se od státní správy, průmyslu, strojírenství, zemědělství, obchodu, bankovníctví až po zdravotnictví. Přinášejí usnadnění lidské činnosti, rychlost, efektivitu, transparentnost, úsporu finančních prostředků a tím i konkurenční výhody. Jejich vývoj a realizace jsou však velmi specifické a nelze pro ně použít běžné postupy a vzory postupů využívaných v jiných oborech. Proto se v současnosti vytváření nových IS věnuje velká pozornost a vznikají různé modely, metodiky a standardy, které mají tuto velmi komplikovanou, náročnou a důslednou činnost usnadnit. [1]

Mezi základní model tvorby IS patří jeho životní cyklus (dále jen ŽC), který v sobě zahrnuje jednotlivé fáze činností od zrodu první myšlenky na nový či aktualizaci stávajícího IS, přes jeho tvorbu, testování, instalaci, provoz, až po jeho vyřazení. Každá z etap ŽC má svá specifika, která vyžadují, aby se jim věnovala značná pozornost. Mezi nejdůležitější a časově nejnákladnější patří samozřejmě analýza a návrh, včetně precizního programování. Přestože implementace patří také mezi velmi problematické a dá se říci, že i mezi nejkritičtější části vývoje IS, bývá často opomíjená z hlediska podceňování její důležitosti a nevěnuje se jí taková pozornost, jakou by si zasloužila. Problematika implementace IS je velmi významná, protože může mít poměrně výrazný vliv na dodržení termínů zavedení aplikace a celkovou velikost finančních nákladů. V neposlední řadě úspěšný průběh implementace aplikace IS může vést k pozitivnímu přístupu uživatelů v reálném prostředí.

Každý projekt vývoje IS je svým způsobem výjimečný. Unikátnost projektu je dána jeho účelem a cílovým zaměřením. Jejich zvládnutí však může v praxi působit problémy i zkušeným manažerům a informatikům. Proto je cílem této bakalářské práce co nejvíce ozřejmit studentům tuto problematiku a poukázat i na chyby a problémy, které v praktickém reálném prostředí mohou nastat.

Hlavním cílem bakalářské práce je tedy přiblížit studentům na praktickém příkladu problematiku implementace informačního systému ve smyslu zavádění již naprogramované a u dodavatele odladěné aplikace, v tomto případě aplikace pro veterinární péči VETERIS, do reálného prostředí podniku.

2 INFORMAČNÍ SYSTÉM

Každý systém, tedy i IS, je jednoznačně určen svými hranicemi vzhledem ke svému okolí a je sestaven z prvků a jejich vzájemného propojení. Tato propojení se nazývají vazby a určují vztahy mezi prvky, které určují strukturu systému. IS lze chápat z mnoha hledisek, proto je možné se setkat s mnoha definicemi, které se liší, právě podle důvodu, pro který je IS definován. [2]

Dle [2] str. 11 lze IS definovat jako „funkční propojení lidí, dat, procesů, rozhraní, sítí a technologií. Jednotlivé prvky spolupracují tak, aby podporovaly a zlepšovaly každodenní operace v organizaci a zároveň aby podporovaly řešení problémů a proces rozhodování v rámci managementu.“

Takže ve své podstatě IS umožňuje uživateli získávat, zpracovávat, ukládat, přenášet, postupovat, prezentovat, využívat a ochraňovat určitá data jak automatizovaně, tak i ručně. Hlavním úkolem IS je tedy poskytovat uživateli, od nejnižší až po nejvyšší úroveň, informace v přiměřeném množství, ve vhodné formě a především ve správné časové rovině. [2]

IS by měl být dle [2] zejména otevřený, dynamický, podporovaný, komplexní, kompaktní, standardizovaný, stavebnicový, chráněný, kompatibilní, neredundantní, nenákladný, kvalitní a rychle zavedený.

2.1 Kvalita informačního systému

Kvalita patří mezi hlavní vlastnosti IS zejména v současnosti. Využívání IS se významně projevuje ve všech oblastech lidské činnosti a stává se téměř nepostradatelnou nutností. Se vzrůstem závislosti společnosti na IS významně narůstá význam vývoje a výběru vysoce kvalitního systému tak, aby nedocházelo k jeho selhání a následně i k možným ztrátám dat, finančních prostředků i obtížně vyčíslitelného dobrého jména. Maximální kvalitou, kterou lze chápat například jako stabilitu, spolehlivost, funkčnost, efektivnost informačního systému, jsou ušetřeny náklady na pozdější správu, provoz a úpravy problémů vzniklých během provozování systému.

Na kvalitu softwaru vzhledem k jeho specifické povaze nelze používat stejné normy kvality jako pro hmotné výrobky. Mezi nejvýznamnější normy v oblasti kvality IS můžeme zařadit normu ČSN ISO/IEC 9126 Softwarové inženýrství – jakost produktu, spolu se související normou ČSN ISO/IEC 12207 O procesech v životním cyklu softwaru. [3]

2.2 Bezpečnost informačního systému

Bezpečnost IS je v dnešní době považována za jednu z nejdůležitějších problematik. Toto téma je značně rozsáhlé, a proto je zde uvedena jen základní charakteristika bezpečnostních politik IS. Můžeme vycházet ze dvou základních pohledů. První jsou standardy vyžadované zákonnými normami (de iure), které jsou vymahatelné a vyžadovatelné, např. ISO řady 27000. Druhé jsou pak standardy zákony nevyžadované, ale všemi uznávané, respektované a dodržované (de facto) např. ITIL (Information Technology Infrastructure Library). [4]

Při rozboru a řešení problematiky bezpečnosti je velmi důležitý komplexní pohled, opomenutí jedné na první pohled nepodstatné části může vést následně až ke kolapsu celého systému. Z tohoto důvodu by se studie měla zaměřit především na oblast fyzické, personální, logické bezpečnosti, bezpečnosti informačních technologií (dále jen IT) a legislativní záležitosti. Zaměřit průzkum jen na některé konkrétní oblasti může způsobit neodhalení možných reálných problémů (rizik).

Bezpečnostní politika je soubor zásad a pravidel, s jejichž pomocí podnik (firma) chrání svá data. Průběžně se aktualizuje jako odezva na změny ve vnějším prostředí a nové bezpečnostní požadavky. Podle přístupu k problematice zabezpečení a dle [4] jsou rozlišovány čtyři základní typy:

- Promiskuitní – povoleno je naprosto vše (i nebezpečné až rizikové činnosti)
- Liberální – povoleno je naprosto vše (kromě jasně stanovených činností)
- Opatrná – zakázáno je naprosto vše (kromě jasně stanovených povolených činností)
- Paranoidní – zakázáno je naprosto vše (i to, co by nemuselo být zakázáno)

Ekonomické náklady na zabezpečení IS by neměly přesáhnout hodnotu dat v něm obsažených, v opačném případě se systém stává nerentabilním. Přesto je třeba pro dobré zabezpečení IS kombinovat několik různých způsobů zabezpečení (např. přístupová hesla, šifrování, antivirová ochrana, firewall, fyzické zabezpečení, nařízení a směrnice). Příliš velká snaha o zabezpečení může být až kontraproduktivní, uživatelé sice požadavky splní, ale časem začnou hledat cesty, jak si svou činnost usnadnit (hesla na monitoru, „složitá“ hesla splňující požadavky, v extrému až sabotování bezpečnostních politik). Návrh, údržba i aktualizace bezpečnosti IS jsou nedílnou součástí celého průběhu životního cyklu IS.

2.3 Způsoby řešení informačního systému

Vzhledem k různorodosti požadavků kladených na každý IS není možné vymyslet univerzální systém, který bude použitelný ve všech podmínkách a případech. V dnešní době již existuje několik základních variant pořízení IS. Výběr konkrétní varianty by měl být zvolen

až po důkladně provedeném rozboru situace daného podniku, firmy, které hodlají zavést nový IS. Dle [2] a [5] se běžně jedná o volbu některé z následujících variant:

Vlastní vývoj IS na míru včetně implementace

IS se vytváří přímo v podniku vlastními zaměstnanci a dle potřeb podnikových procesů a specifických požadavků daného odvětví. Takto vytvářený systém se může dále rozvíjet a přizpůsobovat nově vzniklým podmínkám. Výhodou je i to, že je zde výrazně menší možnost úniku informací o plusech a minusech tohoto IS. Vývoj je však velmi nákladný a zdlouhavý a je zde i možnost nižší kvality.

Vývoj IS externím dodavatelem na míru a vlastní implementace

U této varianty vytváří nový IS specialisté a analytici externího dodavatele opět na základě potřeb podnikových procesů a specifických požadavků daného odvětví. Oproti předešlé variantě je vývoj obohacen o zkušenosti a znalosti externího dodavatele, avšak právě proto je zde i větší možnost úniku informací.

Nákup hotového IS od externího dodavatele a vlastní implementace

Zde se pořizuje velmi obecný IS, který je spolehlivý a jsou v něm zahrnuty zkušenosti i postřehy všech uživatelů. Jedná se o poměrně rychlé zavedení IS do podniku s nevelkými náklady, ale jeho velkou nevýhodou je, že systém nemusí zcela přesně odpovídat představám zákazníka a specifickým požadavkům daného odvětví, oboru. Je zde i možnost, že tento IS vychází ze zastaralých technologií, které mohou zapříčinit problémy při jeho zavedení.

Nákup IS od externího dodavatele včetně implementace

Jedná se o nejrychlejší způsob s nízkými náklady. Oproti předešlé variantě je zde možné rozložit riziko spojené s implementací IS mezi podnik a dodavatele, stoupá možnost úniku interních informací a závislost podniku na dodavateli.

Tvorba IS na míru dodavatelem včetně outsourcingu

Tato varianta zaručuje, že IS bude vyvinut na míru dle požadavků a potřeb podnikových procesů a specifických požadavků daného odvětví externími analytiky, specialisty. Zároveň s tím je zabezpečen i outsourcing, tedy správa, údržba, obnova, systému externím dodavatelem, čímž je možné snížit počet zaměstnanců zabezpečujících běh IS podniku, a tedy i snížení některých nákladů. Odpovědnost za danou problematiku se tím přenesse na dodavatele. Nejedná se o nejlevnější variantu, vývoj na míru vždy přináší větší náklady a stoupá zde podstatně i závislost na dodavateli a možnost úniku informací.

3 ŽIVOTNÍ CYKLUS INFORMAČNÍHO SYSTÉMU

Vývoj a existence IS od zrodu první myšlenky na nový či aktualizaci stávajícího IS, přes jeho tvorbu, testování, instalaci, provoz, až po jeho vyřazení či opětnou aktualizaci nazýváme životním cyklem IS. Tento cyklus tak udává základní etapy vývoje a běhu IS.

Návrh, vývoj IS, případně jeho úpravy, rozvoj, aktualizace, povýšení (upgrade) lze provádět podle velkého množství metodik, standardů, postupů a modelů, které se od sebe odlišují podle typu možného řešení systému, druhu IS i dle oboru, předmětu podnikání firmy, podniku, kteří IS vlastní, případně pořizují. [4]

3.1 Metodiky

Doporučené postupy se označují jako metodiky. Jedná se o návod, který popisuje všechny činnosti, které se vztahují k návrhu IS. Tyto činnosti jsou dále rozpracovány na jednotlivé pořadí kroků, úkolů, jakým způsobem mají být zpracovány i kdo z vývojového týmu je má zpracovávat. Lze konstatovat, že každá firma zabývající se vývojem a distribucí a správou (údržbou) IS má své vlastní metodiky, přizpůsobené potřebám firmy. Tyto firemní metodiky jsou buď vlastní, nebo většinou vycházejí ze standardních metodik. [4]

Rozdíl mezi jednotlivými metodikami spočívá většinou v tom, že každá řeší vznik možných rizik jiným způsobem, a proto není možné používat dvě metodiky pro stejnou etapu vývoje. Bylo by to vytržené z kontextu a ve výsledku by celek nefungoval tak, jak bylo původně metodikou zamýšleno. Hlavním cílem metodik je tedy snížení rizika vzniku chyb. [1]

Dalším příkladem jsou metodiky univerzitní, jako je například metodika MIDS Multidimensional Development of Information System aneb Metodický základ systémové integrace. Tato metodika je vyvíjena Doc. Ing. Jiřím Voříškem, CSc. na Vysoké škole ekonomické, Katedře IT. Základem této metodiky při návrhu a tvorbě IS je, že v každé etapě navrhování a analyzování IS je postupováno v souladu se všemi dimenzemi a jejich vzájemnými vazbami. [6] Dle [6] se jednotlivé etapy odlišují úrovní abstrakce:

Strategická úroveň	globální strategie podniku informační strategie podniku
Konceptuální úroveň	úvodní studie globální analýza a návrh detailní analýza a návrh
Implementační úroveň	tvorba softwaru implementace provoz a údržba

Nezbytností je tedy účast nejen kvalitních odborných pracovníků jako programátorů, analytiků, ale i zadavatele a budoucích uživatelů v celém průběhu projektu z důvodu stanovení podnikových cílů, specifik oboru podniku, nástin podkladů pro analýzy a návrhy a především následné testování. [6]

Mezi další známé strukturované metodiky patří např. holandská metodika SDM (System Development Methodology) a SSADM (Structured Systems Analysis and Design Methodology) z Velké Británie. Z objektových metodik to jsou např. UML (Unified Modeling Language) unifikovaný modelovací jazyk, OMT (Object Modeling Technique) Objektově orientované metodiky a technologie, OOAD (Object-Oriented Analysis & Design) Objektově orientovaná analýza a návrh a OOSE (Object-Oriented Software Engineering) Objektově orientované softwarové inženýrství. Každá z těchto metod měla svoji vlastní hodnotu a důraz. OMT (Rumbaugh) zdůrazňuje analýzu, OOAD (Booch) návrh a OOSE (Jacobson) analýzu chování. [7]

3.2 Standardy

Českým standardem životního cyklu, který vydal v roce 2002 Úřad pro veřejné informační systémy, je Standard ISVS 005/02.01 pro náležitosti životního cyklu IS. Stejně jako každý jiný standard stanovuje veškeré základní etapy a procesy životního cyklu a dále přesně určuje veškerou dokumentaci v celém jeho průběhu. Standardem je stanovena povinnost provádět dokumentaci životního cyklu informačního systému včetně jeho dílčích projektů a údržby bez určení detailních požadavků její realizace. To znamená, že žádná metodika ani model životního cyklu není preferován pro použití při vývoji informačního systému. Rozhodnutí o volbě konkrétního modelu, a tím i zodpovědnost za vývoj, je plně v rukou vývojového týmu a správců. Pouze použití prvků ve standardu označených termínem „musí“ je pro tým povinné, aby bylo vyhověno požadavkům standardu. [8]

Tento český standard životního cyklu odkazuje i na mezinárodní standard ČSN ISO/IEC 12207:1997 Informační technologie. Procesy v životním cyklu software (Software Life Cycle Processes), jehož návrh pochází již z roku 1988.

3.3 Základní modely životního cyklu

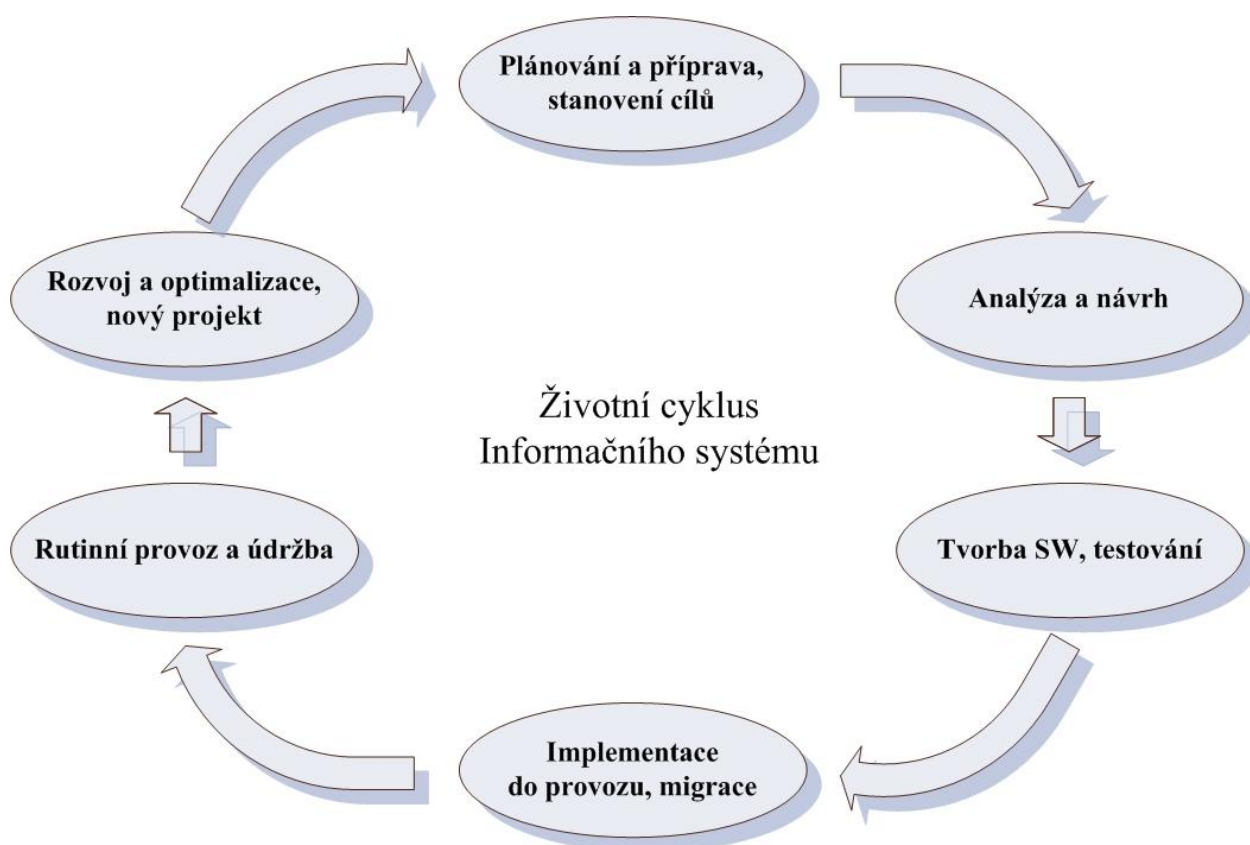
Jednotlivé fáze životního cyklu neprobíhají ve skutečnosti postupně, ale mohou se opakovat, nebo i překrývat. V dnešní době je vytvořeno několik potvrzených modelů, mezi základní patří model vodopád, iterativní, výzkumník, prototypový a spirálový. Dále pak to jsou například v-model, aplikační balíčky, vývoj uživatelem, objektově orientovaný apod. V praxi se však často využívají kombinace těchto modelů, ne jen jeden konkrétní model. [9]

Každý z těchto modelů má svá specifika, výhody i nevýhody. Jejich podrobnější popis je uveden v příloze A, aby byl studijní materiál úplný.

3.4 Etapy životního cyklu

Mnohé metodiky, zdroje a modely tedy uvádějí různé etapy od těch nejzákladnějších jako je výběr, implementace, provoz a inovace přes rozšířenější základní pohled například podle modelu životního cyklu vodopád (viz příloha A) na definici požadavků, návrh IS a softwaru, implementace a testování modulů, testování systému a provoz a údržba až po dopodrobna rozpracované metodiky jako je například MIDS.

Ve své podstatě se však dá říci, že ze všech výše uvedených postupů, metodik a modelů lze určit tyto základní etapy životního cyklu: plánování a příprava, analýza a návrh, implementace a testování, zavedení do provozu, rutinní provoz a údržba a nakonec rozvoj a optimalizace. Názorně je zobrazen sled těchto etap na následujícím obrázku 1.



Obrázek 1 - Životní cyklus IS (zdroj: upraveno na základě [4])

3.4.1 Plánování a příprava, stanovení cílů

V této etapě je hlavním cílem rozhodnout, zda je opravdu potřeba nový IS. K odpovědi na tuto otázku, na kterou odpovídá vrcholový management podniku, který zavedení nového IS zvažuje, je nutné zpracovat podkladové materiály. Provádí se vstupní analýza a její výsledky jsou zapracovány do tzv. úvodní studie, která slouží nejen jako podklad pro rozhodnutí vrcholovému managementu, ale i jako podklad pro výběrová řízení v případě řešení dodavatelskou formou.

Zpracování úvodní studie může ovlivňovat v některých směrech i chod podniku. Především zaneprázdněním odborných pracovníků z oblasti IT a speciálních odvětví, kterých se nový IS týká, nebo v případě zpracování externími specialisty i zvýšenými náklady. [10]

Pokud se vedení podniku rozhodne pro zpracování úvodní studie, měla by především obsahovat nejen důvody pro realizaci projektu, ale i stanovení strategických cílů podniku. Dále by měly být její součástí tyto části: návrh nového řešení, srovnání původního řešení s novým, stanovení konkrétních požadavků jako potřeby uživatelů včetně odůvodnění, analýza požadovaných funkcí, procesů a vazeb, analýza možných rizik včetně návrhu jejich řešení, bezpečnostní strategie, požadavky na lidské zdroje, chybějící kvalifikace, školení, motivace pracovníků, potřebné a podpůrné programové vybavení, hardware a potřebné úpravy síťové infrastruktury, předběžný plán realizace a samozřejmě i ekonomické zhodnocení projektu, tedy náklady, doba návratnosti, zhodnocení přínosů atd. [10]

Nejedná se o přesné detailní zpracování, to se provádí až v následující etapě životního cyklu. Zpracování úvodní studie by nemělo být vedením podniku bráno na lehkou váhu (ignorováno), protože její neexistence nebo nekvalitní zpracování obvykle vede v dalších fázích životního cyklu k podstatnému snížení efektivity, a tím i k podstatně větším nákladům.

3.4.2 Analýza a návrh

Analýza je detailnějším a podrobnějším rozpracováním předchozí etapy. Je jí potřeba věnovat abnormálně velkou pozornost, protože veškerá opomenutí a chyby jsou při pozdějším odhalení těžko odstranitelné a zvyšují náklady. Provádí se především analýza podnikových procesů, stávajících databází, stávajících aplikací, funkcionality a nemělo by se zapomínat na vzájemné vazby všech procesů, bezpečnost apod. Analýza i návrh neboli projektová studie se musí zpracovat dostatečně podrobně a srozumitelně pro veškeré zúčastněné složky, jak pro dodavatele IS (ať z vlastních řad, či z externí firmy na základě výsledků výběrového řízení), tak i pro vrcholové vedení podniku zvažující pořízení IS. Využívají se k tomu metody datového modelování, které se řeší na konceptuální, logické a fyzické úrovni, dále pak procesní modelování. U procesních modelů se využívá různých možností zápisu, jak slovního, matematického, tak i grafického. [4], [9], [11]

Existuje celá řada těchto modelů od všem známého vývojového diagramu, který znázorňuje logickou strukturu procesu, přes Diagram hierarchie (funkční strom) všech procesů, funkční model systému na úrovních IDEF0, IDEF1, IDEF2, IDEF3 (Integration DEFinition), dále model kontextu procesu, který znázorňuje vazby systému k jeho okolí, model EPC a rozšířený eEPC (extended Event-driven Process Chain), znázorňující toky jednotlivých procesů. Dalším příkladem modelu znázorňujícím toky procesů a také zodpovědnost je model BPMN (Business Process Modeling Notation) a v neposlední řadě model Případu užití znázorňující funkčnost systému, který stejně jako model BPMN je založen na jazyku UML.

Pokud vrcholový management podniku souhlasí s realizací, slouží výsledky této etapy jako podklad pro smlouvu s dodavatelem nového IS i jako podklad pro jeho implementaci, a tedy i realizaci.

3.4.3 Tvorba softwaru, testování

V této etapě již dochází k vlastní tvorbě IS, tedy jeho programování na základě specifikací z předešlých etap životního cyklu, a testování všech modulů, funkcí, fází programování. Programování probíhá na základě získaných informací z úvodních studií, analýz a fyzického návrhu. Definují se vstupy a výstupy jednotlivých operací a určí se způsob jejich modifikace. Naprogramují se veškeré funkce a doladí se jejich vzájemné propojení, výpočtové postupy, nastavení parametrů jednotlivých modulů, detailní struktura, vzhled a obsah jednotlivých obrazovek se přizpůsobuje zákazníkovi apod. Dále se připravuje převod dat z původní verze IS, pokud existuje, případně tvorba nových databází, včetně aktualizace a tvorby základních číselníků. [4], [11]

Interní testování tvůrcem softwaru rovněž patří k velmi důležité části vývoje systému a nemělo by se opět podceňovat. V praxi se často stává, že bývá zkracováno, v nejhorším případě i vynecháno, z důvodu možného překročení termínu. Toto testování se neprovádí u zadavatele, ale u dodavatele na testovacích datech, která by měla obsahovat maximální množství konečných reálných dat, aby nedošlo například k nenahraditelným ztrátám dat u zadavatele z důvodu chyby vyvíjeného IS. V průběhu tohoto testování se systém postupně přeměňuje z rozpracovaného systému na hotový. Základní typy testování jsou funkční, uživatelské a výkonnostní, zátěžové a bezpečnostní. [1]

Veškeré postupy a vývoj IS se musí zaznamenávat do dokumentace k pozdějšímu využití pro potřebu školení, zajištění provozu i budoucích úprav a upgrade. [9]

3.4.4 Implementace do provozu, migrace

Proces implementace by měl probíhat podle naplánovaného harmonogramu, aby přechod od starého systému k novému byl uskutečněn v krátké době bez zbytečných průtahů. Implementace IS se skládá převážně z instalace, nastavení, ověřování a testování jak hardwaru, tak i softwaru potřebného k provozu nového IS, dále pak z převodu původní datové báze do nového systému, provádění zátěžových a akceptačních testů, ověřování celého IS. Zároveň probíhá přidělování přihlašovacích jmen a hesel, poskytnutí manuálů, školení uživatelů k obsluze, údržbě systému i bezpečnosti. Závěrem této etapy je předávací řízení.

Před provedením instalace se většinou provádí upgrade výpočetní techniky, kterou je třeba přizpůsobit potřebám zaváděného IS (kapacitně, výkonnostně...), poté probíhá instalace na servery, jednotlivé uživatelské počítače, terminály. Dále pak se zprovozňuje komunikace mezi externími systémy. [4]

Převod dat může probíhat různými způsoby. Rozhodnutí, jaký způsob bude zvolen, většinou vychází z analýzy stávající databáze, pokud existuje. Vytvořit databáze je možné převodem stávajících, pokud jsou v dobrém stavu (vyčištěné od duplicit, nesmyslných údajů apod.), anebo ručním zadáním nových databází, což je velmi časově náročný postup a závisí od rozsahu a stavu stávající databáze. Dále je možno provést dle požadavku zadavatele i sloučení databází z různých pracovišť podniku pro centrální evidenci dat.

Školení je vhodné věnovat dostatečnou pozornost, aby se u uživatelů nevytvořila nechuť k novému IS a předešlo se sabotování průběhu implementace a tím i nedodržení harmonogramu. Školení je možné provést dodavatelem u všech uživatelů. Tento typ školení se moc nevyužívá, je dost nákladný a náročný na organizaci a běh podniku i na množství možných dotazů. Většinou se využívá školení dodavatelem, ale jen vybraných klíčových uživatelů. Tito poté proškolí koncové uživatele systému. V praxi se většinou toto školení provádí těsně před zavedením na konkrétních datech podniku a využívá se i k předběžnému testování, čímž se může i zkrátit doba doladění. [10]

Během implementace je dobré naslouchat připomínkám a potřebám uživatelů a dle potřeby operativně přizpůsobit plán implementace. Implementace končí kladným vyhodnocením akceptačních testů. Poté vchází nový systém do fáze rutinního provozu. [9], [11], [12]

V závislosti na implementaci nového informačního systému probíhají zároveň i určité změny v podniku jako například zavedení nových interních normativních aktů, organizačních a bezpečnostních směrnic, norem provozu i manuálů informačního systému, dále pak i možné personální změny apod. [9], [13]

Ve zkušebním provozu již pracují s implementovaným systémem uživatelé, kteří s ním budou pracovat každodenně po jeho úspěšném zavedení a přijetí. Během zkušebního provozu se odstraňují nalezené chyby a je možné zapracovat ještě dodatečné požadavky na úpravy od uživatelů. Pokud se aplikace chová dle požadavků a je plně funkční s minimem nevýznamných, jen zřídka se vyskytujících chyb, zkušební provoz můžeme prohlásit za ukončený. Jedná se o velmi důležitou část vývoje, která je velmi závislá na vzájemné spolupráci zadavatele a vývojového týmu dodavatele informačního systému. Jeho zdárný průběh většinou zajišťuje vyšší kvalitu systému. Na závěr zkušebního provozu by měly proběhnout akceptační testy, které slouží jako podklad pro předávací řízení. [1]

V závěru této etapy probíhá předávací řízení. Probíhá po realizaci testování systému dodavatelem, provedení akceptačních, výkonnostních testů a po úspěšném průběhu zkušebního provozu. Poté dodavatel se zadavatelem provádějí odsouhlasení funkcionality, výkonnosti, spolehlivosti, dokumentace celého průběhu i samotného IS formou předávacího protokolu.

Implementaci lze řešit některým z následujících postupů:

Souběžné zavádění

Je založeno na principu souběžného chodu nového IS na všech pracovištích a původního, stávajícího systému po určitou dobu, což je značně náročné na kapacitu. Současný provoz obou systémů většinou trvá do té doby, dokud nový systém nepracuje spolehlivě a jeho uživatelé jsou dostatečně zapracováni. Poté dojde k odstavení a uložení původního systému pro omezený počet uživatelů jako možný podklad pro zpětnou analýzu za určitá období, statistické vyhodnocení, plánování, rozhodování apod. Od tohoto bodu se nadále pracuje pouze s novým systémem, viz níže uvedený obrázek 2. [11]

Jedná se o bezpečný princip (postup). Nevýhodou je, že po dobu souběžného chodu obou IS musí uživatelé provádět stejné operace dvakrát. U větších podniků je možné tuto nevýhodu řešit v tomto období externími pracovníky. [11]



Obrázek 2 - Souběžné zavádění IS (zdroj: upraveno na základě [14])

Pilotní zavádění

U tohoto zavedení spočívá princip na zavedení nového IS na jednom vybraném, tzv. pilotním pracovišti, na kterém probíhá zkušební provoz. Ověřuje a doladuje se zde funkčnost nového IS. Jako pilotní pracoviště by mělo být vybráno takové, u kterého se dá ověřit co nejvíce problémových oblastí. Na tomto pracovišti pak dále může po odladění probíhat i zaškolování uživatelů dalších pracovišť. Po ukončení zkušebního provozu na pilotním pracovišti se nový systém dále zavádí na ostatní potřebná a již připravená pracoviště, viz obrázek 3. [11]

Tento princip implementace je výhodný při implementaci výrazně odlišných IS, které vyžadují rozsáhlé testování v provozních podmínkách, a umožňuje také postupný převod původní datové základny. [11], [14]



Obrázek 3 - Pilotní zavádění IS (zdroj: upraveno na základě [14])

Postupné zavádění

U postupného zavádění nedochází k implementaci IS najednou, ale po jeho částech. Nejdříve se zavádějí primární části IS, na kterých ostatní části závisí, po jejich ověření a odstranění nedostatků se podobným postupem zavádí ostatní části až po zavedení celého systému. Využívá se především u velmi složitých systémů, které mají komplikované vnitřní vazby a bylo by obtížné ověření systému jako celku. [11] Názorné zobrazení postupného zavádění je na obrázku 4.



Obrázek 4 - Postupné zavádění IS (zdroj: upraveno na základě [14])

Nárazové zavedení

Princip nárazového zavedení je v ukončení a odstranění starého IS a následné implementaci nového IS bez či s minimálním přechodem, jak ukazuje níže uvedený obrázek 5. Tento postup je velmi rizikový, proto se používá pouze výjimečně a to většinou tam, kde není možné zajistit souběžné zavádění nového IS. Není však náročný na čas a pracovní síly. [11], [14]



Obrázek 5 - Nárazové zavedení IS (zdroj: upraveno na základě [14])

Implementace nového IS do podniku, firmy bývá většinou problematické, a proto je velmi důležitá volba vhodného způsobu jeho zavedení. Tato volba by měla vyplývat z posouzení například typu a funkce předchozího systému, objemu změn, způsobu ovládání, připravenosti pracovišť a uživatelů na zavedení IS apod. [11]

V praxi se však více využívá kombinací těchto způsobů, nejčastěji nárazového a postupného zavádění IS. [14]

3.4.5 Rutinní provoz a údržba

Tato etapa by měla trvat co možná nejdéle, aby se při provozu a užívání IS naplnily nejen veškeré předpoklady, které se od nového systému očekávaly, ale aby se vrátily i vynaložené náklady. Jinak by nemělo ani význam o novém IS uvažovat. Tato etapa začíná předáním systému do provozu po potvrzení předávacího protokolu oběma stranami, které se účastní tohoto projektu (většinou dodavatel, programátoři, pracovníci IT... a zákazník, uživatel...). Patří sem běžný provoz systému, údržbové operace, provozní servis jak hardwaru, tak i softwaru, dále pak podpora uživatelů a stálé monitorování IS včetně jeho bezpečnosti.

Do provozního servisu patří zabezpečení chodu, údržba, správa a kontrola veškeré infrastruktury jako počítačové podnikové sítě, správa databází, využití úložných kapacit, zálohování a správa databází a podobně. V praxi se většinou s dodavatelem uzavírá smlouva o podpoře a údržbě IS. Veškeré potřebné opravy a navrhované úpravy IS se konzultují s dodavatelem. Hodnotí se, zda je úprava či oprava nad rámec původního zadání, na základě toho se řeší i finanční stránka. Veškeré tyto realizované změny musí být však řádně otestovány a zaznamenávány do projektové dokumentace pro další možný upgrade systému. [4]

V průběhu činnosti IS je třeba provádět neustálé monitorování veškerých výpadků, poruch, vzniklých chyb IS, ale i jeho výkonnost a závady na technickém vybavení. Toto sledování je pak možné statisticky zpracovat a slouží jako podklad pro aktualizaci a návrhy na změnu IS, případně jeho technického zázemí, jako třeba rozšíření o záložní server.

Podporou uživatelů jsou míněny průběžné konzultační služby uživatelům, takzvaný helpdesk. Pracovníci helpdesku pomáhají operativně řešit uživatelům jejich problémy či požadavky

související s činností IS. V případě závažnějšího problému se řešení většinou postupuje o úroveň výše na programátory a analytiky systému. Zde se opět provádí sledování a vyhodnocování řešených problémů a zpracovávají se jako podklady pro možné úpravy, případně update či upgrade systému. [4]

Bezpečnosti by měla být věnována značná pozornost, a to již v průběhu prvních fází životního cyklu, tedy už při samotném zvažování pořízení či aktualizace IS. V průběhu provozu je nutné se zaměřit na oprávněné přístupy, opětovná školení uživatelů, ukončení platnosti přístupových práv po změně zařazení či odchodu uživatelů z podniku apod. Součástí je samozřejmě i vyhodnocování a řešení bezpečnostních rizik. [4]

3.4.6 Rozvoj a optimalizace, nový projekt

Rozvoj a optimalizace, ukončení stávajícího IS nebo rozhodnutí pro nový IS je poslední etapou životního cyklu. Podměty a návrhy pro úpravy a změny stávajícího systému vznikají během rutinního provozu, při kterém se provádí monitorování všech poruch, výpadků, závad a nefunkčností včetně poznatků uživatelů, které by neměly zůstat stranou. Tyto požadavky je nutné posoudit a vyhodnotit v rámci změnového řízení určenými oprávněnými zástupci podniku. Ti musí posoudit a určit, zda je výhodnější (únosná) ještě renovace stávajícího systému, nebo už je načase přejít k novému IS, čímž se vracíme k první etapě životního cyklu. Vše začíná znovu od začátku a cyklus se tím uzavírá. Zjednodušeně lze říci, že životní cyklus je něco jako stoupající spirála, vír, který s výškou získává i na objemu s rostoucím vědeckým pokrokem a vývojem v této oblasti.

Důvody pro přistoupení k novému IS mohou být různé, od těch zásadních, jako jsou změny strategických cílů a procesů, organizační, prodej podniku, přes změny technologické, vývojové, výrobní, změny legislativy, podnikových předpisů až ke změnám z důvodu mohutného rozsahu úprav, které přesahují rámec update či upgrade. Důvodem může být i doba provozu systému a rozsah již provedených úprav, kde dalšími úpravami by se značně zvýšilo riziko vzniku závažných chyb ve funkčnosti. [4]

K ukončení IS dochází ze stejných důvodů jako pro přistoupení k novému IS, jak je uvedeno výše. Pokud dochází k náhradě rušeného IS novým, postupuje se podle jednoho z dříve uvedených postupů implementace. Poté stejně jako v případě ukončení IS bez náhrady se provede zálohování veškerých dat na kvalitní média s dlouhodobou životností pro možné obnovení a přístupy k potřebným datům, případně se i vygenerují a vytisknou veškeré tiskové sestavy v obecně dostupných formátech. Jejich další ošetřování a udržování je prováděno v souladu s archivačními, zálohovacími předpisy a postupy organizace. Jako zajímavé řešení se jeví vytvoření virtuálního

systemu s funkčním IS včetně jeho dat, který je následně zálohován výše uvedenými postupy. Podrobnější zpracování této problematiky je natolik rozsáhlé a přímo nesouvisející s tématem bakalářské práce, proto není její součástí.

4 POPIS STÁVAJÍCÍHO STAVU A NÁVRH VÝBĚRU INFORMAČNÍHO SYSTÉMU

Tuto problematiku jsem si vybrala z důvodu, protože jsem byla zaměstnavatelem začleněna jako jeden ze zástupců veterinární služby Armády České republiky (AČR) do realizačního týmu, který začal plánovat a realizovat implementaci nového veterinárního IS nazvaného VETERIS. Studenti tedy budou seznámeni s průběhem realizace implementace tohoto IS, včetně uvedení problémů, které mohou v praxi nastat. To znamená, že v praxi ne vždy vše probíhá podle návrhu a plánu.

4.1 Veterinární služba Armády České republiky

Veterinární služba AČR je organizačně začleněna do struktur zdravotnické služby AČR. Veškerá činnost veterinární služby AČR musí být v souladu s právní legislativou a interními normativními akty. Mezi nejdůležitější patří Zákon 166/99 Sb. o veterinární péči ve znění pozdějších předpisů, Zákon 246/92 Sb. o ochraně zvířat proti týrání ve znění pozdějších předpisů, Zákon 110/1997 Sb. o potravinách ve znění pozdějších předpisů a z interních normativních aktů například Vojenský předpis Vet-1-2 Veterinární zabezpečení.

Veterinární služba AČR odborně spolupracuje a vzájemně si vyměňuje informace s ostatními složkami rezortu MO, mimo rezort pak s dalšími odbornými složkami na úseku veterinární péče. Plní úkoly v oblasti rozvoje vojenské kynologie u vojenských útvarů a zařízení, zajišťuje biologickou ochranu letišť, veterinární asanaci apod. Veterinární péče se dělí na odbornou část, obecnou část a na výkon státní správy. [15]

4.1.1 Obecná veterinární péče

Obsah obecné veterinární péče zahrnuje specifické úkoly veterinární služby AČR, a to zejména organizaci, odborné řízení a výkon vojenské kynologie, organizaci, odborné řízení a výkon biologické ochrany letišť a péče a zabezpečení veterinárním materiálem. [15]

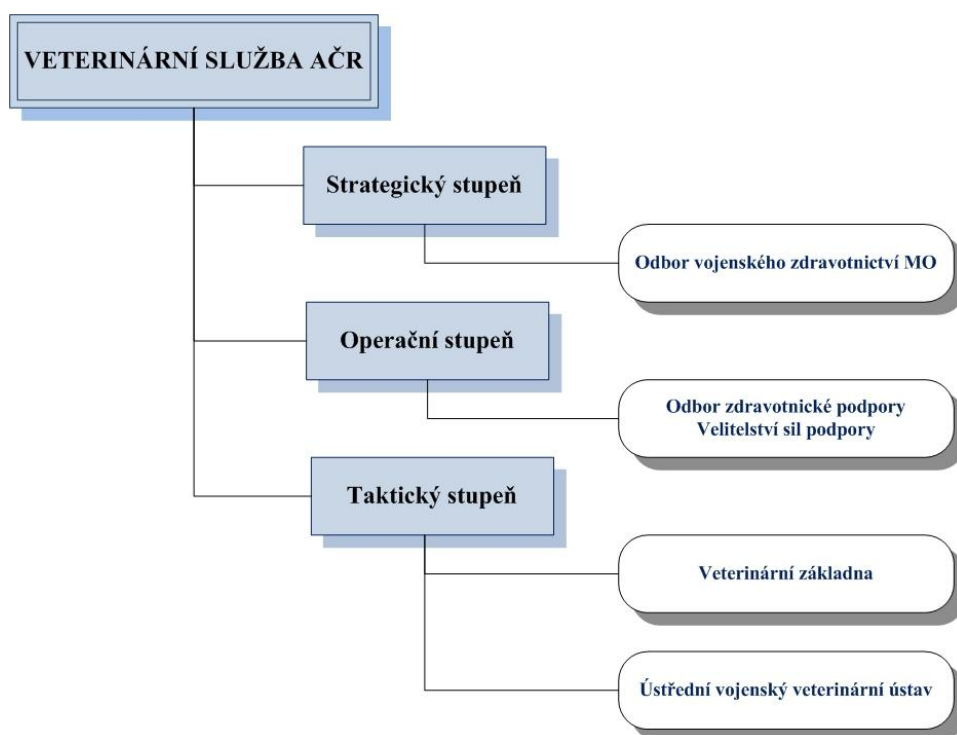
4.1.2 Odborná veterinární péče

Oblast odborné veterinární péče je komplexním souborem veterinárních, léčebně preventivních, protinákazových, hygienických, diagnostických, profylaktických, zásobovacích a dalších veterinárních činností, které v rezortu MO zajišťují orgány veterinární služby AČR. Má za cíl vytváření optimálních životních podmínek, zabezpečení pohody zvířat a jejich ochrany proti týrání, které zahrnují vhodné ustájení, krmení, čištění, péči o výstroj, správné využívání v souladu s biologickým založením zvířete apod. [15]

4.1.3 Výkon státní správy

Výkon státní správy ve věcech veterinární péče vykonává veterinární služba AČR v souladu s veterinárním zákonem a správním řádem. Správní řízení umožňuje orgánům státní správy rozhodovat o právech chráněných zájmech nebo povinnostech v oblasti státní správy. Další důležitou oblastí činnosti je úzká spolupráce se Státní veterinární správou České republiky při ochraně území České republiky právě před zavlečením nákaz zvířat a nemocí přenosných ze zvířat na člověka a před dovozem zdravotně závadných živočišných produktů a krmiv za zahraničí. [15]

Jak je znázorněno na následujícím obrázku 6, je v současné době veterinární služba AČR z pohledu řízení rozdělena do tří úrovní.



Obrázek 6 - Úrovně veterinární služby AČR (zdroj: vlastní)

4.2 Informační systém veterinární služby Armády České republiky

Původní aplikace na evidenci zdravotní dokumentace služebních zvířat veterinární služby je již z pohledu dnešních aplikací značně zastaralá. Byla vyvinuta ještě jako 16bitová aplikace pro operační prostředí MS DOS s využitím databázového prostředí FoxPro. V současné době již je údržba a další rozvoj této aplikace s ohledem na zastaralost vývojového a operačního prostředí a pokrok v metodikách a technologiích vývoje softwaru (objektové technologie, síťová

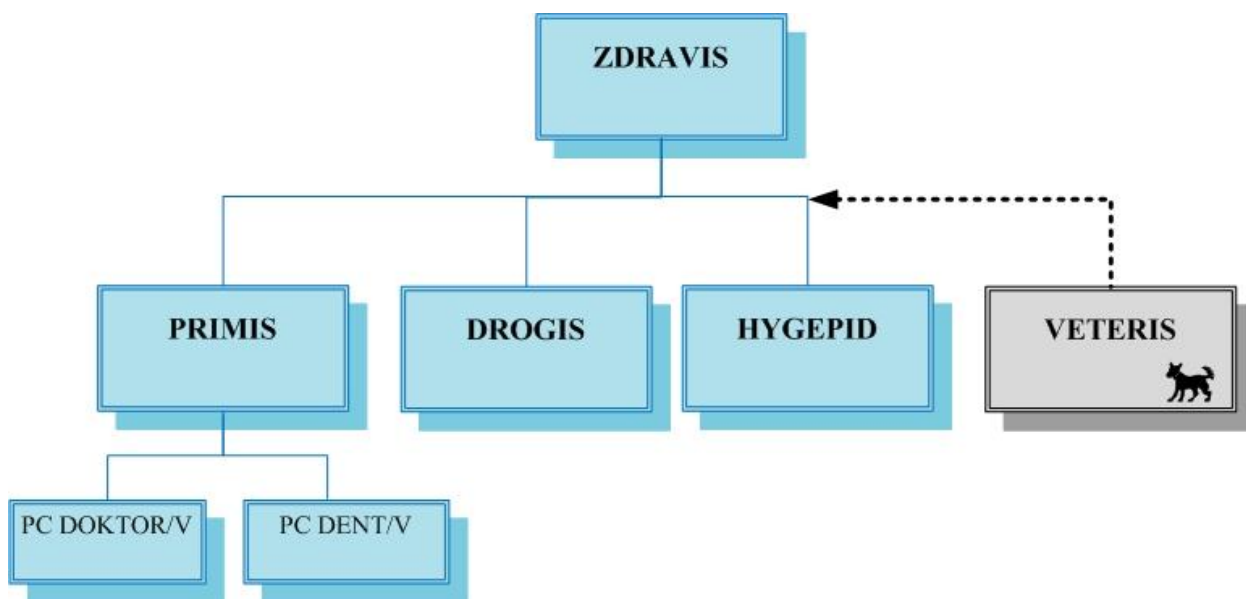
komunikace, zabezpečení, zálohování, apod.) vysoce nákladná, komplikovaná, a tím i nerentabilní. Náklady na potřebné úpravy začaly být ve značném nepoměru k výsledku.

4.2.1 Způsob řešení pořízení nového veterinárního informačního systému

Z výše uvedených důvodů se postupně začalo zvažovat opuštění této aplikace a vývoj nového IS. Rozhodovalo se, zda využít nabídky firem, distribuujících IS na evidenci veterinární péče o zvířata pro veterinární ošetrovny v soukromém sektoru, či vývoj úplně nové aplikace.

Způsob řešení pořízení nové aplikace podpořilo sloučení veterinární a zdravotnické služby AČR, na základě kterého bylo rozhodnuto o integraci stávajícího veterinárního IS do zdravotnického IS ZDRAVIS. Zastaralý veterinární IS však naprosto nevyhovoval požadavkům na integraci a připojení vzhledem k aplikační logice a zásadním rozdílům ve struktuře dat. Proto bylo rozhodnuto o vývoji nové aplikace VETERIS.

Na obrázku 7 je uvedeno schéma ZDRAVIS včetně jeho funkcionalit s nově integrovanou aplikací VETERIS.



Obrázek 7 - Schéma ZDRAVIS (zdroj: vlastní)

PRIMIS – primární lékařská humánní péče v rámci AČR. Obsahuje vedení zdravotnické dokumentace, tisk receptů, vystavování neschopenek a dalších poukazů na různá vyšetření. Dále se využívá k vedení statistiky v rámci aplikace a vyúčtování pojišťoven. Je tvořena dílčími podsystémy PC DOKTOR/V pro praktické lékaře a PC DENT/V pro zubní lékaře.

DROGIS – aplikace, která na základě diagnóz a dalších ukazatelů hodnotí drogovou závislost.

HYGEPID – vedení hygienicko-epidemiologické situace v rámci posádkových ošetřoven a systém hlášení začínajících epidemií a pandemií, zatím v rozpracovaném stavu.

VETERIS – nově implementovaná samostatná aplikace, vycházející ze standardu PC-DOKTOR/V pro veterinární část zdravotnické služby.

4.2.2 Návrh aplikace VETERIS

Jak vyplývá z výše uvedené kapitoly a z důvodu charakteru vojenských dat, byl jako vhodný způsob řešení, po zvážení všech možných variant, vybrán vývoj IS externím dodavatelem na míru a vlastní zavedení. Model jednotlivých etap ŽC aplikace VETERIS se při úvodních úpravách studií neřešil. Způsob tvorby návrhu a programování tohoto nového IS, který je tvořen na základě podobné již užívané aplikace PC-DOKTOR/V, sám předurčil, spolu se zkušenostmi dodavatele a pracovníků oddělení IT s realizací nových IS, typ modelu ŽC. Jde o kombinaci modelu ŽC prototypového a výzkumníka (viz příloha A), které v tomto případě tvorby, programování a úprav aplikace VETERIS byly nejvhodnější. Důvodem je především možnost implementace prototypů, případně jednotlivých verzí IS, které se upravují na základě praktických zkušeností s aplikací. Odstraňují se tím především nedostatky zjištěné při zkušebním použití, jejichž vznik by se těžko odhaloval od „zeleného“ stolu.

Tento IS byl navrhován a programován na základě dokumentace k ZDRAVIS (PC-DOKTOR/V), protože ve své podstatě se jedná o společný základ lékařských humánních dat, která byla aktualizována na základě odlišných specifik veterinární medicíny a služby. Značnou výhodou byla úspora času a finančních prostředků při tvorbě prvotních analýz, neboť se nemusela vytvářet celá dokumentace od prvopočátku, ale byla doplněna pouze o veterinární specifika. Zároveň byl ušetřen čas při výběru dodavatelské firmy, jelikož aplikace VETERIS je podsystémem ZDRAVIS, jedná se o stejnou dodavatelskou firmu, jejíž výhodou byla především důvěrná znalost stávajícího IS a částečná znalost specifického prostředí, infrastruktury, systémového prostředí apod. Dále bylo využito stávajícího hardwarového vybavení současné serverovny, která byla dlouhodobě budována a upgradována pro potřeby celého systému ZDRAVIS.

Vedení projektu bylo doplněno o zástupce veterinární služby, který byl gestorem úprav, oprávněnosti požadavků na změny a doplnění úprav z odborného hlediska. Pro rozvoj systému v oblasti veterinární služby jsou využívány zkušenosti vojenských veterinárních lékařů, pro rozvoj technického a programového vybavení se vychází ze zkušeností a znalostí pracovníků IT AČR a civilní dodavatelské firmy.

Požadavky na bezpečnost a kvalitu zůstaly nezměněné, stejně tak požadavky na základní funkcionalitu IS. Jedinečným identifikátorem, kterým je v PC DOKTOR/V rodné číslo pacienta, je

u aplikace VETERIS číslo čipu, které má zvíře umístěno pod kůží. Mezi hlavní funkce patří kartotéka služebních zvířat, kde ke každému služebnímu zvířeti jsou zobrazovány jeho základní informace. Složka zdravotních záznamů zvířete, tzv. *dekurz*, je tvořena nekonečným pásem záznamů o prodělaných prohlídkách a vyšetřeních včetně účtované péče. Z tohoto *dekurzu* se dále spouštějí další funkce. Bylo nutné změnit a doplnit číselníky diagnóz, léčebných výkonů včetně cen pro vykazování a vyúčtování poskytnuté veterinární péče. Rovněž byly zpracovány číselníky druhů a plemen zvířat, výcvikových kategorií služebních psů, důvodů vyřazení, vojenských útvarů a zařízení se služebními zvířaty, dále pak laboratorní profily včetně doplnění veterinárních specifik neexistujících v humánní verzi PC-DOKTOR/V. Mezi další funkce, které bylo třeba změnit, patří formuláře, a to zejména Zápisy o vyřazování, úhynu, odprodeji služebních zvířat, Protokol o vyšetření zdravotního stavu zvířete, které poranilo člověka apod. Na rozdíl od humánní verze byla aplikace VETERIS rozšířena o sklad veterinárních léků a materiálů. Naopak odbourány byly funkce spojené se zdravotními pojišťovnami apod.

4.2.3 Požadavky na hardwarovou a softwarovou infrastrukturu

Při analýze a návrhu bylo stanoveno jako systémové minimum použití Windows XP PRO Service pack 2 a vyšší a minimální hardwarová konfigurace byla stanovena podle stávajících počítačů, je to 2,4 GHz CPU, 2 GB RAM a 160 GB disk, vypalovačka CD, LCD monitor. Tiskárny se mohou používat libovolné, protože v aplikaci lze přizpůsobit tisk jednotlivých formulářů pro libovolnou tiskárnu. Na základě plánovaných nákupů byly také pořízeny pro spádové veterinární lékaře přenosné počítače (notebooky) o hardwarovém minimu 1,83 GHz CPU a 512 MB RAM.

Pro potřeby záloh a přístupu k datům byl pořízen nový databázový server. Databázovým prostředím je platforma databáze CACHE firmy Intersystem. Popis databázového serveru je uveden v příloze B.

5 POPIS NÁVRHU IMPLEMENTACE APLIKACE VETERIS

Na základě rozhodnutí o vlastní implementaci IS je vedením projektu určen realizační tým pro zavedení aplikace VETERIS včetně stanovení pravomocí a zodpovědností jednotlivých pracovníků tohoto týmu. Skládá se z pracovníků oddělení IT, veterinárního lékaře pilotního pracoviště především jako odborného konzultanta, zástupců nadřízeného stupně veterinární služby AČR a představitele dodavatele aplikace VETERIS.

Na základě hrubého plánu implementace vypracovaného ve fázi analýzy a návrhu je členy realizačního týmu navržen a zpracován plán implementace, ve kterém jsou upřesněny jednotlivé etapy a kroky včetně termínů jejich provedení s ohledem na určený typ implementace, a to kombinace pilotního a souběžného zavádění.

Plán implementace je rozdělen do tří základních fází, sestavených z jednotlivých činností související s implementací aplikace VETERIS, a to přípravné a zahajovací, prováděcí a ukončení zavádění a předání do provozu. Vzor harmonogramu implementace aplikace VETERIS je uveden v příloze C.

5.1 Přípravná a zahajovací fáze

Tato fáze je navržena tak, aby začala probíhat souběžně s tvorbou softwaru. Jsou zde zařazeny činnosti, které mají zabezpečit připravenost k implementaci nové aplikace. Mezi přípravné činnosti patří především prověření softwarové a hardwarové infrastruktury stávajícího vybavení jednotlivých veterinárních pracovišť na základě minimálních požadavků stanovených při analýze a návrhu včetně dodání a instalace nového databázového serveru a notebooků pro spádové veterinární lékaře. Tento materiál byl objednáno již při analýze a návrhu IS. Dále zde jsou uvedeny činnosti k přípravě původních dat k přenosu do nové aplikace z obou původních pracovišť. Nedílnou součástí je i upřesnění přidělení jednotlivých rolí uživatelům na základě návrhu zpracovaného ve fázi analýzy a návrhu ŽC tohoto IS. Závěrečnou a nutnou etapou této fáze je převzetí aktuálního odzkoušeného prototypu aplikace od dodavatele, kterou je v další fázi možno implementovat do zkušebního provozu.

5.2 Prováděcí fáze

Tato fáze je odstartována dodáním funkčního prototypu aplikace k instalaci. Je nejrozsáhlejší a je rozdělena na šest etap, které postupně zabezpečují instalaci a odzkoušení veterinární aplikace ve všech jejích základních režimech provozu (viz příloha C).

Zahájením je samostatná instalace na jedné pracovní stanici pilotního pracoviště (Veterinární základna), na kterém je provedeno testování a ověřování funkcionalit jednotlivých prototypů aplikace. Poté je na tomto pilotním pracovišti instalace rozšířena na více propojených pracovních stanic (peer-to-peer). Další etapou je pak přenos jednotlivých databází z obou pracovišť na centrální server. Nakonec je realizováno sloučení databází obou pracovišť a připojení všech veterinárních pracovišť včetně spádových k této centrální databázi. Na závěr je provedeno odzkoušení vytvořené funkcionality *Odloučený počítač*.

Na pilotním pracovišti je naplánován cyklus s dostatečnou časovou rezervou k ověřování a doplňování potřebné chybějící funkčnosti jednotlivých prototypů. Aktualizace prototypu vychází z ověřených nahlášených připomínek uživatelů k aplikaci a na základě vyhodnocení řízených rozhovorů s těmito uživateli. Každá etapa podléhá důslednému testování a ověřování funkcionality, připojení zálohování jak uživateli, tak i pracovníky informačních technologií. Po dokončení testování je provedena optimalizace nastavení s update prototypu aplikace dodavatelem. V průběhu všech etap postupně probíhají školení uživatelů. Prováděcí fáze je zakončena finálním nastavením a optimalizací aplikace včetně update.

5.3 Ukončení implementace a předání do provozu

Zde jsou naplánovány činnosti, které jsou spojené s posledními aktivitami před předáním aplikace do rutinního provozu. Je zde zařazeno zapracování posledních potřebných změn do interních normativních aktů a celkové zhodnocení způsobilosti aplikace pro rutinní provoz včetně zhodnocení procesu zavádění. Příprava a podpis smlouvy s dodavatelem o poskytování softwarových služeb, jako pozáruční servis, zpracování potřebných úprav, helpdesk. Zároveň dochází k dopracování zprávy a dokumentace zavádění aplikace.

Posledním bodem je právě ukončení zkušebního a přechod do rutinního provozu na všech pracovištích veterinární služby včetně odloučených, spádových, veterinárních pracovišť na základě podpisu závěrečného akceptačního protokolu, který je nedílnou součástí fakturace za implementaci aplikace VETERIS dodavatelem.

6 POPIS REALIZOVANÉ IMPLEMENTACE APLIKACE VETERIS

Pro přehlednost jsou jednotlivé nadpisy v této kapitole doplněny o odkaz na jednotlivé kódy WBS (Work Breakdown Structure), které jsou uvedeny na vzoru harmonogramu aplikace VETERIS v příloze C. Tyto kódy udávají osnovu jednotlivých etap a fází činností.

6.1 Fáze přípravná a zahajovací (WBS 1.1)

Přípravná a zahajovací fáze je spuštěna v průběhu tvorby softwaru a provádějí se tyto následující činnosti.

6.1.1 Prověření zabezpečení hardwarové a softwarové infrastruktury (WBS 1.1.1.)

Na základě implementačního projektu je prověřena a zabezpečena hardwarová i softwarová infrastruktura potřebná k zavedení a chodu aplikace. Je prováděno případné doplnění a modifikace požadavků i v průběhu implementace. Uskuteční se kontrola všech pracovních stanic, přenosných počítačů a případně provedena instalace operačního systému na stanovenou minimální standardní úroveň, případně vyšší.

6.1.2 Příprava dat (WBS 1.1.2)

Před zahájením prováděcí fáze aplikace je provedena kontrola a příprava ostrých dat z předešlé, již dlouho zastaralé aplikace. Datová báze je prověřena především na duplicitní a neúplné záznamy. Je realizováno doplnění a odstranění přebytečných záznamů. Příprava původních dat k přenosu je uskutečněna na obou veterinárních pracovištích. Možnost přenosu původní báze dat do nové aplikace byla jedním z hlavních požadavků při tvorbě analýzy. Na přípravě dat se podílejí jak uživatelé, pracovníci IT, tak i dodavatel aplikace.

6.1.3 Dodávka databázového serveru a notebooků (WBS 1.1.3 a 1.1.4)

Databázový server na zálohování dat včetně notebooků pro spádové veterinární pracoviště je dodán ve stanoveném termínu. Pracovníky IT je provedena instalace na potřebnou softwarovou úroveň, která je ve shodě s požadavky, případně vyšší. Databázový server je zapojen a umístěn v serverovně ZDRAVIS a notebooky jsou předány k užívání na spádová veterinární pracoviště.

6.1.4 Zhodnocení připravenosti zadavatele a dodavatele (WBS 1.1.5 a 1.1.6)

Stávající hardware a software jsou zabezpečeny v požadovaném minimu, případně i vyšším. Nový materiál je dodán v termínu a jeho instalace je bez závad. Pracovníci jsou zařazeni do jednotlivých uživatelských rolí od nejnižších pravomocí příslušných rolí jako *ostatní-*

administrativa přes řadového lékaře, vedoucího lékaře až po velitelství s nejvyššími pravomocemi. Pilotní pracoviště je vyzkoušeno o termínu zavedení aplikace VETERIS včetně termínu školení, je zabezpečena maximální účast uživatelů a klíčových pracovníků. U dodavatele je předveden připravený prototyp k implementaci, potvrzen termín školení klíčových uživatelů.

6.1.5 Dodávka prototypu aplikace VETERIS (WBS 1.1.7)

V další etapě je převzata u dodavatele odzkoušená a na cvičných datech prověřená aktuální verze aplikace VETERIS včetně licencí a vytvoření jednotlivých rolí uživatelů, které jsou rozdílné podle oprávnění přístupu od pouhého nahlížení do vymezených částí aplikace až po neomezenou působnost ve všech funkčních částech aplikace. Termín dodání prototypu aplikace je milníkem a od tohoto data začíná implementace jako taková.

6.2 Fáze prováděcí (WBS 1.2)

Tato fáze je odstartována převzetím aktuální verze aplikace VETERIS od dodavatele a jsou realizovány tyto etapy.

6.2.1 Implementace na pilotní pracoviště – jedna pracovní stanice (WBS 1.2.1)

Instalace – v daném termínu je provedena pracovníky IT na pilotním pracovišti instalace aplikace VETERIS na jednu pracovní stanici v ambulanci veterinárního lékaře v lokálním umístění, zatím bez připojení k centrálnímu serveru.

Migrace dat – po kontrole instalace je provedena migrace již připravených ostrých dat z původní zastaralé aplikace.

Školení klíčových uživatelů – po provedené instalaci je realizováno dodavatelskou firmou prvotní školení a mimo pracovníků pilotního pracoviště se jej účastní i členové realizačního týmu spolu se zástupcem druhého pracoviště (Ústřední vojenský veterinární ústav). Struktura školení je přizpůsobena dle přiřazených uživatelských rolí pracovníků.

Obsahem je prvotní poučení uživatelů o bezpečnostních pravidlech užívání aplikace VETERIS s povinnostmi dodržovat ustanovení související s prací uživatele ve ZDRAVIS a pravidly práce s uživatelským účtem (přístupová jména, hesla, tvorba a životnost bezpečných hesel apod.). Každý uživatel ZDRAVIS externí i interní musí mít prověrku pro práci s utajovanými skutečnostmi pro stupeň vyhrazené, i když nejsou zpracovávána klasifikovaná data. Podrobnější pokyny pro práci uživatele jsou pak uvedeny v směrnici pro manipulaci s uživatelskými účty a provozní bezpečnostní směrnici ZDRAVIS. Dále pak se školení již zaměřuje na seznámení s aplikací, k čemu je určena, její přínosy a pravomoci jednotlivých uživatelských rolí. Samozřejmostí je proškolení ve veškeré funkcionalitě aplikace od zavedení nového pacienta

(zvířete), zápis do *dekurzu*, výdej léku ze skladu přes vytištění dokumentace až po výkazy činnosti za jednotlivá období včetně jejího spuštění, ukončení a zálohování.

Testování, analýza funkčnosti a zálohování – v rámci pilotního zavádění je prováděno postupné odzkoušení funkcionality všech prototypů na ostrých datech tohoto pilotního pracoviště. Uživatelé jsou zpracovány hlášení s požadavky na potřebné změny a úpravy prototypu aplikace a zároveň s nimi jsou prováděny řízené rozhovory. Oprávněnost požadavků je nejprve posouzena zástupci veterinární služby a poté i pracovníky IT. Realizovatelnost akceptovaných požadavků a připomínek je řešena s dodavatelem. Zároveň probíhá i odzkoušení schopnosti aplikace běžet lokálně.

Optimalizace nastavení systému a update prototypu – na základě předaných sumarizovaných požadavků je zapracována dodavatelem aktualizovaná verze prototypu, která po dokončení je opět nainstalována a opět testována. Celý cyklus se takto opakuje ve stanoveném termínu do zapracování všech potřebných funkcionalit a úprav uživatelského rozhraní na pilotním pracovišti.

Zhodnocení etapy:

Tato etapa trvá poměrně dlouho z důvodu doladování a upřesňování všech funkcí a možností aplikace včetně individuálního přizpůsobení uživateli, a tím i výrazného zlepšení komfortu užívání aplikace VETERIS. Přínosné jsou řízené rozhovory s uživateli, které přinášejí i úpravy, které původně nejsou zaznamenány do jejich hlášení. Na každý prototyp je zpracován akceptační protokol se záznamem všech funkčních úprav, který je schválen jak řešitelským týmem, tak dodavatelem. Nevýhodou je nestanovení konečného počtu prototypů. Časový interval není překročen. Úpravy však pokračují v dalších etapách.

6.2.2 Implementace na pilotní pracoviště – více pracovních stanic (WBS 1.2.2)

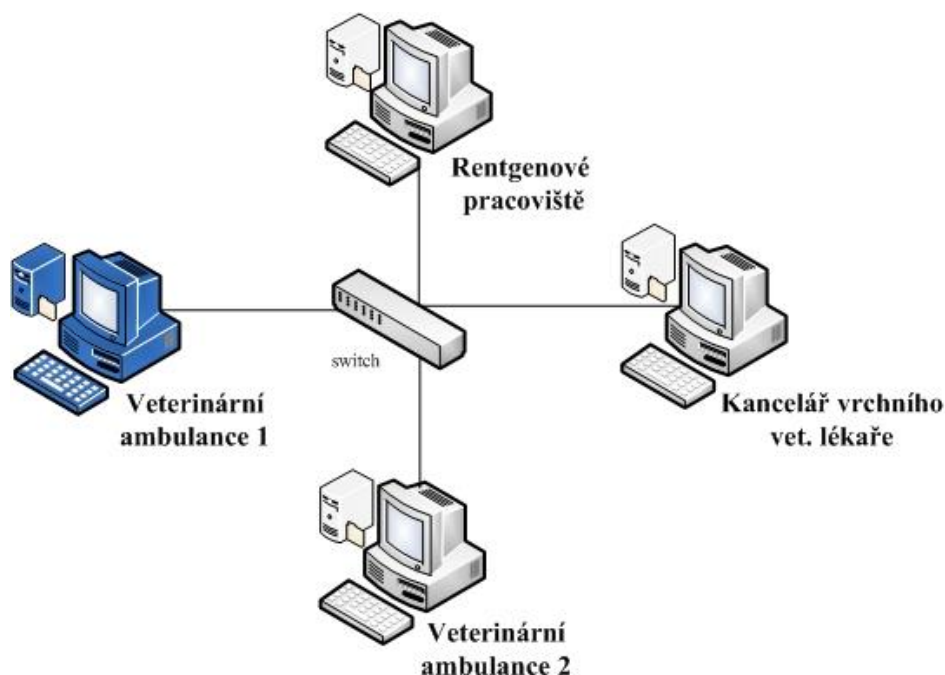
Instalace – zde dochází již k instalaci na další pracovní stanice v druhé ambulanci veterinární kliniky pilotního pracoviště a dále ještě na pracoviště rentgenu a kancelář vedoucího veterinárního lékaře. Pracovní stanice jsou propojeny v síti peer-to-peer s využitím síťových služeb založených na Windows DCOM. Každý počítač může být jak pracovní stanicí, tak i serverem. V případě výpadku veterinární ambulance 1 se přenesou záloha například na veterinární ambulanci 2 a ostatní pracovní stanice se hlásí k databázi tohoto počítače. Všechny stanice mají stejné vybavení. Tato aplikace má umět pracovat lokálně i dokázat obsloužit až 10 uživatelů, kteří využívají databázi na jedné pracovní stanici. Není zde potřeba centrálního serveru. Zapojení pracovních stanic je schematicky znázorněno na obrázku 8.

Testování, analýza funkčnosti a zálohování – probíhá otestování funkčnosti připojení jednotlivých pracovních stanic k této jedné databázi na jedné z těchto čtyřech pracovních stanic. Pokračuje se dále v ověřování funkcionality prototypů. Jsou zpracována hlášení s požadavky na potřebné změny a úpravy uživatelského rozhraní aplikace spolu s výsledky řízených rozhovorů opět procházejí ověřením oprávněnosti a řešitelnosti.

Optimalizace nastavení systému a update prototypu – akceptované připomínky jsou opět dodavatelem zapracovány do nového prototypu.

Zhodnocení etapy:

Zavedení a testování je splněno v požadovaném čase, propojení pracuje bez závad. Po pilotním zavedení je aplikace VETERIS připravena k implementaci na další pracoviště a je ukončeno souběžné zpracování v nové a původní aplikaci. Od tohoto okamžiku je nová aplikace VETERIS brána jako primární.



Obrázek 8 - Propojení pracovních stanic peer-to-peer na pilotním pracovišti (zdroj: vlastní)

6.2.3 Migrace dat na databázový server (WBS 1.2.3)

Migrace dat pracovišť a připojení – na centrální databázový server CACHE jsou umístěna data pilotního pracoviště. Pomocí celoarmádní datové sítě (CADS) je pilotní pracoviště připojeno na tento centrální server. CADS je vlastní datová a komunikační síť AČR, tzv. intranet, která využívá vlastní kabeláže a částečně pronájmu od telekomunikačních firem. Podobné datové sítě mají i České dráhy nebo složky Ministerstva vnitra. Poté jsou na tento datový server přenesena

data z druhého pracoviště, která jsou opět před vložením zkontrolována, doplněna a vyčištěna. K těmto datům je připojeno druhé pracoviště pomocí modemového připojení.

Školení všech koncových uživatelů – součástí jsou i další školení, jak základní pro budoucí uživatele, tak i zdokonalovací pro současné uživatele. Zdokonalovací školení jsou opět řízena dodavatelem IS. Základní školení uživatelů již provádí realizační tým.

Zátěžové, ověřovací, testy připojení a síťové funkčnosti – otestování propojení pomocí architektury klient – server. Zkušebním režimem vzdáleného připojení je prověřena rychlost připojení pilotního pracoviště, zda vyhovuje běžnému zpracovávání dat v běžném provozu. Doba odezvy je zjištěna do 20 ms. Tato doba je postačující pro běžný provoz pilotního pracoviště a splňuje nedefinované požadavky uživatelem, a tím i dobrý uživatelský komfort. Testováním vzdáleného přístupu druhého pracoviště při běžném provozu a samozřejmě po proškolení a zařazení příslušných uživatelů je však zjištěno, že i přes původní plánované předpoklady dostačujícího datového přenosu je zjištěna doba odezvy kolem 40 ms. Tato odezva sítě již znemožňuje pohodlnou interaktivní práci s aplikací a snižuje tím značně i uživatelský komfort aplikace na minimum.

Testování, analýza funkčnosti a zálohování – stejný postup jako v předchozích etapách, sběr požadavků, ověření, předání. Dochází k rozšíření počtu uživatelů.

Optimalizace nastavení systému a update prototypu – akceptované připomínky jsou opět zapracovány do nové verze aplikace VETERIS.

Zhodnocení etapy:

Přínosem je rozšíření počtu uživatelů, a tím i návrhů z jiného úhlu pohledu, především pak uživatelského rozhraní. Odhaleno je nepříjemné zjištění, které značně prodlužuje a komplikuje dokončení celé implementace aplikace. Je nutné zabezpečit větší přenosovou rychlost a kapacitu pro připojení veterinární ambulance druhého pracoviště pomocí výkonnějších a rychlejších modemů, a to k celoarmádní síti CADS. Je zahájeno jednání o připojení veterinárního pracoviště k CADS, ale z důvodu předem neplánovaných finančních prostředků a úsporných opatření v rámci resortu se připojení proti plánovanému harmonogramu značně opozdilo. Je nutná aktualizace (posun) harmonogramu implementace, protože ani plánované časové rezervy nestačí pokrýt toto zpoždění způsobené i vysokou administrativní náročností. Tento čas je využit k doladování aplikace. Do připojení na server pracuje druhé pracoviště s aplikací lokálně. Po nápravě jsou migrována aktuální data a testy provedeny u druhého pracoviště znovu, tentokrát již s vyhovujícím výsledkem.

6.2.4 Centralizace dat (WBS 1.2.4)

Příprava sloučení dat – v průběhu jednání a realizace připojení druhého pracoviště k CADS probíhá převod a sloučení původní datové báze jednotlivých pracovišť. Na sloučení se podílí především dodavatel. Databáze služebních zvířat jsou sjednoceny na základě jedinečného identifikátoru, kterým je číslo čipu umístěného pod kůží zvířete a snímaného pomocí čteček. Záznamy a veškeré údaje jsou datově a časově jednoznačně určeny, a protože jsou obě pracoviště od sebe vzdálena mnoho kilometrů, není ani fyzicky možné, aby v jeden okamžik bylo uvedené zvíře ošetřováno na obou pracovištích zároveň. Aby bylo ihned zřetelné, které pracoviště má právě služební zvíře v péči, byla už v průběhu programování aplikace VETERIS zabudována funkce *registrace*.

Sloučení bází dat, připojení pracovišť – je nutné odstavit obě pracoviště v jeden termín a v tomto okamžiku ve spolupráci s dodavatelem je uskutečněno překlopení na centrální databázi.

Zátěžové, ověřovací testy připojení a síťové funkčnosti – opět jsou provedeny zátěžové testy připojení obou pracovišť k centrální databázi. Do této doby obě pracoviště pracovala se svou konkrétní oddělenou verzí databáze na serveru.

Zhodnocení etapy:

V průběhu slučování nedochází k žádné nesrovnalosti. Nepředpokládá se a ani nenastává žádná kolize při slučování. Sloučením těchto databází je vytvořen centrální registr všech služebních zvířat, která jsou v péči veterinární služby AČR.

6.2.5 Rozšíření na spádová pracoviště (WBS 1.2.5)

V praxi se stává, že je zapotřebí provozovat program nejen v základní ordinaci připojené na centrální databázi, ale i na jiných ordinačních místech (v terénu). Tato situace je řešena mobilním počítačem (notebookem) spolu s funkcí *odloučený počítač*. Tato funkce tedy umožňuje veterinárním lékařům pracovat s aplikací VETERIS i na vzdáleném pracovišti, útvaru, zařízení se služebními zvířaty bez připojení k centrálnímu registru služebních zvířat. Zde mohou provést potřebné veterinární zákroky, prevenci apod. a zároveň je ihned zaznamenat do aplikace, která je nainstalována na jejich mobilních počítačích. Je tedy možné prostřednictvím této funkce zajistit zkompletování dat v centrální databázi nejen pro potřeby společného vykazování péče, ale i pro účely sledování výkonnosti celého zařízení.

Instalace – na notebooky spádových veterinárních lékařů se instaluje poslední, aktuální verze aplikace VETERIS.

Školení – opět probíhá základní školení pro budoucí uživatele i zdokonalovací pro současné uživatele. Specifickou částí tohoto školení je především připojování k centrálnímu registru služebních zvířat a pro spádové veterinární lékaře připojení pomocí funkce *odloučeného počítače*.

Zátěžové, ověřovací testy připojení – prověření funkce odloučený počítač, která je nezbytná pro potřeby přenosu dat, a tím aktualizace centrální databáze.

Zhodnocení etapy:

Implementace i školení jsou realizovány ve stanoveném aktualizovaném termínu, aplikace VETERIS je v tomto okamžiku implementována na všechna pracoviště veterinární služby AČR. Podceněna však je motivace uživatelů. Jedná se ve větší míře o starší veterinární lékaře bez pozitivního vztahu k IT, kteří v nové aplikaci spatřují spíše nepřítele než pomocníka, který jim posléze usnadní jejich činnost.

6.2.6 Finální optimalizace (WBS 1.2.6)

V této etapě jsou zapracovány poslední změny v aplikaci a jejím nastavení na základě ověřených a schválených připomínek uživatelů před uzavřením procesu implementace a přechodu na rutinní provoz. Opět dochází ke kontrole zapracovaných úprav.

Zhodnocení etapy:

Veškerá požadovaná funkcionalita aplikace VETERIS je zapracována, po finální optimalizaci je aplikace připravena na přechod do rutinního provozu. Přechází se do závěrečné fáze ukončení implementace.

6.3 Fáze ukončení implementace a předání do provozu (WBS 1.3)

Během celého procesu postupné implementace od pilotního pracoviště až k spádovým veterinárním lékařům je vše zaznamenáváno do prováděcí dokumentace a souběžně probíhají pravidelná konzultační jednání realizačního týmu s vedením projektu a zástupci dodavatele. Jsou řešeny výsledky zátěžových testů připojení, testy funkčnosti, obnovy a akceptační testy včetně zpracování protokolů.

Zároveň je prováděno dopracování a zapracování posledních potřebných změn do interních normativních aktů veterinární služby AČR, bezpečnostních směrnic administrátora a uživatele, provozně bezpečnostních směrnic, provozního řádu serverovny, směrnice pro manipulaci s uživatelskými účty, směrnice technického provozu. Všem uživatelům jsou předány a distribuovány poslední aktuální verze manuálů aplikace VETERIS.

Po odzkoušení všech funkcionalit aplikace VETERIS je posouzena její způsobilost a připravenost na přechod ze zkušebního do rutinního provozu. Je zhodnocen proces implementace

a vše je zpracováno do průběžné dokumentace řešení implementace. Na základě zjištěných skutečností je předpřipravena smlouva o poskytování softwarových služeb. Při závěrečném předávacím řízení je sepsán zápis o průběhu celého řízení, jsou potvrzeny finální akceptační protokoly, průběžné akceptační protokoly byly řešeny a potvrzovány v průběhu celé implementace po provedených vyhovujících testech. Zároveň jsou s dodavatelem dohodnuty servisní podmínky IS a podepsána smlouva o poskytování softwarových služeb a podpory uživatelů, která zahrnuje například:

- čtvrtletní upgrade aplikace
- 14denní odzkoušení tohoto upgrade
- požadavek na řešení případné reklamace do jednoho týdne od podání
- vytvoření systému helpdesk, kdy prvotní hlášení problému s aplikací je nahlášeno na pracoviště IT uživatele a v případě jeho nevyřešení je dále předáno k řešení dodavateli

Poté je ukončen zkušební provoz a je odstartován rutinní provoz na všech pracovištích včetně spádových veterinárních lékařů. Posledním krokem je dopracování závěrečné zprávy o implementaci aplikace VETERIS.

Spuštěním rutinního provozu se přešlo z etapy ŽC implementace do etapy rutinního provozu a údržby. Jejím hlavním úkolem bude provozovat aplikaci na stanovených veterinárních pracovištích s cílem zajistit požadované služby s požadovanou provozní a nákladovou efektivností, kvalitou a spolehlivostí. Dále budou pracovníci oddělení IT jak vlastními silami, tak i ve spolupráci s dodavatelem zajišťovat logistické, servisní a zabezpečovací služby umožňující nepřetržitý provoz systému, monitorovat provoz, zabezpečovat sběr uživatelských požadavků na zefektivnění provozního zabezpečení a provozní úpravy, realizovat průběžnou přípravu uživatelů a evidovat a vyhodnocovat mimořádné události a poruchy.

7 ZHODNOCENÍ IMPLEMENTACE

Použití prototypového modelu ŽC se ukázalo jako velmi vhodné pro vývoj aplikace ve vybraném prostředí. Postupně se prototyp zdokonaloval a upravoval na základě připomínek uživatelů jak v uživatelském rozhraní, tak i pro přínosné úpravy (doplnění některých funkcí). Řešitelský tým akceptoval po dohodě s konciliem uživatelů i zapracování funkcionalit, které původně nebyly navrženy. Zabránilo se tím zdlouhavému čekání na hotovou aplikaci a možnosti odhalení závažných funkčních nedostatků až ve zkušebním běhu úplné verze aplikace, což by bylo podstatně časově i finančně nákladnější. Převážná většina funkčních úprav probíhala v rámci pilotního zavádění na pilotním pracovišti. Na ostatní pracoviště už byla implementována téměř dokončená aplikace VETERIS. Dále docházelo převážně k úpravám uživatelského rozhraní.

Jak již bylo dříve uvedeno, vhodnou volbou dodavatele se podařilo vyvarovat značných nejasností z neznalosti specifického prostředí podniku. Spolupráce s dodavatelem při analýze, tvorbě, implementaci i při rutinním provozu a údržbě probíhala a nadále probíhá k oboustranné spokojenosti. Nedochovalo a ani zatím nedochází k žádnému neetickému jednání, ani k zneužívání vzájemné důvěry. To bylo podpořeno systémovým přístupem ve zpracovávání veškeré dokumentace včetně zaznamenávání průběžných změn. Přesto se při zpracovávání závěrečných dokumentů projevila spolupráce jako ne zcela dostatečná, neboť některé potřebné informace bylo nutné pracně dohledávat.

Jako pozitivní se ve spolupráci s dodavatelem ukázalo i velmi dobré zpracování harmonogramu implementace. Jednotlivé fáze a etapy na sebe postupně navazují a byl v nich ponechán dostatečný časový prostor především na otestování a prověření všech funkcionalit v rámci celého průběhu implementace až na nedostatečnou časovou rezervu. Plánovaná časová rezerva se ukázala jako nedostačující z důvodu řešení neuspokojivé kapacity připojení druhého pracoviště. V časovém harmonogramu muselo proto dojít k úpravám, a to k posunu všech navazujících operací o tuto prodlevu. Velkým nedostatkem se totiž ukázalo, že při analýze bylo chybně odhadnuto síťové připojení, čímž došlo k odhalení malé přenosové rychlosti (kapacity) až v průběhu testování při zkušebním provozu. Tím vzniklo jak odchýlení (zpoždění) od časového plánu z důvodu administrativního vyřizování žádostí o připojení na CADS a o přidělení neplánovaných finančních prostředků, tak i k navýšení nákladů celé etapy implementace z důvodu realizace tohoto zvýšení kapacity síťového připojení.

Delší časový horizont byl vhodný i z důvodu souběžného zavádění, kterým se do určité míry snížilo vytížení testujících pracovníků při zpracovávání údajů jak ve stávajícím, tak i v zaváděném

IS na pilotním pracovišti. Vhodným řešením bylo i zapojení uživatelů do testování a připomínkování aplikace v celém průběhu implementace od pilotního pracoviště až po spádové veterinární lékaře.

Kladně je možno hodnotit přístup ke školení uživatelů, které probíhalo ve více etapách při postupném rozšiřování implementace aplikace. Dále v rozdílných specifikacích zaměřených na rozdílné role uživatelů a v rámci úspor bylo základní školení prováděno formou realizačního týmu a nová specifika pak dodavatelskou firmou aplikace VETERIS.

Naproti tomu byla však především podceněna motivace a informovanost budoucích uživatelů této aplikace. V jejím využívání sice viděli usnadnění své práce a zvýšení efektivity, ale cítili i obavy ze zvýšené možnosti dohledu nadřízené složky nad jejich činností. U některých pracovníků tyto pocity vedly až k pasivnímu přístupu. Na místo podnětných, konstruktivních připomínek přicházeli s množstvím formálních až „hnidopišských“ připomínek a požadavků a hledali důvody, proč tuto aplikaci nezavést. Naštěstí pro realizační tým bylo těchto pracovníků minimum.

Podpora vývoje a implementace informačního systému byla značně závislá i na osobních preferencích řídicích pracovníků, kterých se během implementace, testování a rutinního provozu aplikace VETERIS vystřídal několik.

Přes termínové a finanční potíže z důvodu nutnosti zabezpečit větší přenosovou rychlost a kapacitu se aplikaci VETERIS podařilo úspěšně implementovat bez dalších větších komplikací. Přínosem je transparentnost odborné činnosti veterinární služby, podpora řízení, ale i velká úspora času při vyhodnocování, zpracovávání podkladů apod., které bez softwarové podpory bylo velmi náročné, a to nejen časově. Implementací aplikace byla tedy především zvýšena efektivnost a snížena administrativní zátěž jak pracovníků veterinárních pracovišť, tak i nadřízených složek. VETERIS slouží v současné době u AČR k vedení zdravotní dokumentace ošetřovaných zvířat, vyplňování a tisku dokladů, které se při veterinární péči obvykle vystavují a které jsou předepsány interními předpisy. Stručný popis funkcionality a ukázka uživatelského rozhraní této aplikace je uvedena v příloze D.

Pro další vývoj a rozvoj aplikace bych doporučila opět provést načtení aktuálních ceníků veterinárních úkonů. Postupně zdokonalovat funkčnost skladového hospodářství s léky a zdravotnickým materiálem. Vhodné by bylo i doplnění klasických manuálů aplikace o interaktivní manuál, který by do značné míry omezil zatížení pracovníků IT zabývajících se uživatelskou podporou řešením banálních a rutinních uživatelských problémů s provozem aplikace. Vývoj aplikace je však v současné době omezen jen na základní udržovací a nutné úpravy funkčnosti

z důvodu úsporných finančních opatření, a to nejen v rezortu MO. Toto úsporné opatření však může mít naopak za následek po delším časovém období podstatně větší náklady na úpravy a aktualizaci aplikace, která se v dnešním sprinterském tempu vývoje IT bez průběžné modifikace stává velmi rychle morálně zastaralou.

8 ZÁVĚR

„Šedá, můj příteli, je všechna teorie, a žití zlatý strom se zelená," zní výrok klasika J. W. Goetheho.

Studenti v rámci své odborné přípravy nemají možnost se reálně seznámit s praktickou problematikou vývoje a implementace rozsáhlejšího IS vzhledem k časové a finanční náročnosti těchto projektů. Běžný rutinní „projekt“ v rámci studia spočívá prakticky pouze v instalaci a konfiguraci vlastního osobního počítače v rozsahu maximálně jednoho dne. Z tohoto důvodu jsem se rozhodla zpracovat tuto problematiku podrobněji z pohledu teorie i reálných zkušeností člena realizačního týmu. Forma zpracování je volena tak, aby vzniklý materiál byl přístupný a srozumitelný studentům denního studia při jejich odborné přípravě v této problematice.

Implementace IS patří mezi časově nejrozsáhlejší a nejkomplicovanější etapy ŽC a také mezi nejrizikovější. Cílem této práce bylo popsat reálný postup implementace IS v reálném prostředí nejen z pohledu teoretického modelování těchto procesů, ale zejména s důrazem na jejich aplikaci v praktickém řešení této problematiky. Rovněž bylo snahou poukázat i na rizika a problémy, které vznikají až při skutečné realizaci, a do jaké míry je lze či nelze v rámci plánovacích procesů předpokládat.

Po zralé úvaze byl použit model pilotního a souběžného zavádění, protože nebylo možné přerušit rutinní provoz veterinární služby AČR. Pilotní způsob umožnil odhalení nedostatků vzniklých při návrhu aplikace zejména v oblasti uživatelského rozhraní a změn v legislativě, které proběhly až po ukončení analytické fáze návrhu. Zároveň byly zpracovány připomínky běžných uživatelů na chybějící funkcionality nebo příliš těžkopádnou dostupnost některých rutinních funkcí pomocí dílčích upgrade jednotlivých prototypů aplikace aktualizovaných na základě těchto akceptovaných a realizačním týmem posouzených připomínek uživatelů.

Souběžný provoz stávající aplikace umožnil upravit v rámci připomínkových řízení uživatelské rozhraní tak, aby přechod do nového IS byl pro běžné uživatele co nejsnazší.

Pro objektivnější posouzení kvality implementace zavedení nového IS by bylo vhodné zhodnocení celého procesu externím, nezaujatým a nezainteresovaným pozorovatelem.

V práci je rovněž poukázáno na problémy vzniklé se získáním finančních prostředků na pořízení neplánovaných kapacit a zdrojů, jejichž potřeba byla zjištěna až v pokročilé fázi realizace. Tyto formální postupy jsou u státních rozpočtových organizací komplikované a značně zdlouhavé vzhledem k nutnosti delšího časového horizontu plánování finančních prostředků a vyjádření celé řady nadřízených a spolupracujících složek rezortu MO.

IS se podařilo zavést do reálného provozu s určitým časovým zpožděním způsobeným nutností zabezpečit zvýšenou přenosovou kapacitu datové linky připojením druhého pracoviště. Tento požadavek vyplynul až v průběhu realizace, vůči veškerým analytickým předpokladům týkajících se kapacity těchto linek.

Obvyklé běžné problémy ohledně uživatelského přístupu způsobené převážně nechutí uživatelů učit se novou aplikaci byly překonány právě maximálním přizpůsobením uživatelského rozhraní aplikace VETERIS původnímu IS včetně zapracování značné části jejich požadavků. Tato nechuť byla do značné míry způsobena nedostatečnou motivací všech zainteresovaných pracovníků, kteří tuto činnost vykonávali navíc v rámci své běžné pracovní náplně.

Zadostiučiněním vývojovému i implementačnímu týmu bylo, že v rámci následných rutinních aktualizací uživatelé kvitovali, že s novou aplikací jsou schopni své běžné povinnosti zvládnout rychleji při zachování kvality. Toto neformální sdělení nejvíce potěšilo od největších odpůrců nové aplikace VETERIS.

Po vyhodnocení celého zavádění bylo konstatováno, že je vhodné provádět průběžné kontroly i za účasti nezaujatého externího odborníka, který se orientuje jak v oblasti IT, tak v profesní oblasti dané služby. Rovněž se ukázalo jako nezbytné velmi pečlivě vést průběžnou dokumentaci o zavádění, byť v rámci zavádění tohoto projektu byla vedena na odpovídající úrovni. Přesto při zpracovávání závěrečných protokolů bylo nutné některé podrobnější informace pracně dohledávat.

9 POUŽITÉ ZDROJE

- [1] PALETA, Petr. *Co programátory ve škole neučí aneb softwarové inženýrství v reálné praxi*. Brno : Computer Press, a. s., 2003. 360 s. ISBN 80-251-0073-1.
- [2] KOMÁRKOVÁ, Jitka. *Úvod do informačních systémů : pro kombinovanou formu studia*. Pardubice: Univerzita Pardubice, 2006. 85 s. ISBN 80-7194-870-5.
- [3] Česká technická norma ČSN ISO/IEC 9126-1 *Softwarové inženýrství – Jakost produktu – Část 1: Model jakosti* [online]. 2010 [cit. 2010-10-31]. Dostupné z WWW: <<http://shop.normy.biz/d.php?k=65526>>
- [4] GÁLA, Libor; POUR, Jan; ŠEDIVÁ, Zuzana. *Podniková informatika 2. Přepřacované a aktualizované vydání*. Praha : Grada Publishing, a. s., 2009. 496 s. ISBN 978-80-247-2615-1.
- [5] TVRDÍKOVÁ, Milena. *Zavádění a inovace informačních systémů ve firmách*. Praha : Grada Publishing, a.s., 2000. 110 s. ISBN 80-7169-703-6.
- [6] ŠMÍD, V. *Metodika MDIS* [online]. 2007 [cit. 2010-09-26]. Metodika MDIS. Dostupné z WWW: <<http://www.fi.muni.cz/~smid/mis-mdis.htm>>.
- [7] BENEŠ, M. *Metodiky a notace UML* [online]. 2008 [cit. 2010-10-31]. Dostupné z WWW: <<http://objekty.vse.cz/Objekty/MetodikyANotace-UML>>.
- [8] *Standard ISVS 005/02.01 pro náležitosti životního cyklu informačního systému* [online]. 2008 [cit. 2010-10-31]. Dostupné z WWW: <http://aplikace.mvcr.cz/archiv2008/micr/files/457/uvis_s005.02.01_v2002c5_20021218.pdf>.
- [9] *Informační systémy* [online], 2009 [cit. 2010-07-28]. Informační systémy. Dostupné na WWW: <<https://akela.mendelu.cz/~rybicka/prez/infssystem.pdf>>.
- [10] VYMĚTAL, Dominik. *Projekty informačních systémů v podnicích a jejich realizace*. Opava : Slezská univerzita v Opavě, 2008. 122 s. ISBN 978-80-7248-477-5.
- [11] ŠMÍD, V. *Vladimír Šmíd : Úvod* [online]. 2007 [cit. 2010-07-29]. Životní cyklus informačního systému. Dostupné z WWW: <<http://www.fi.muni.cz/~smid/mis-zivcyk.htm>>.

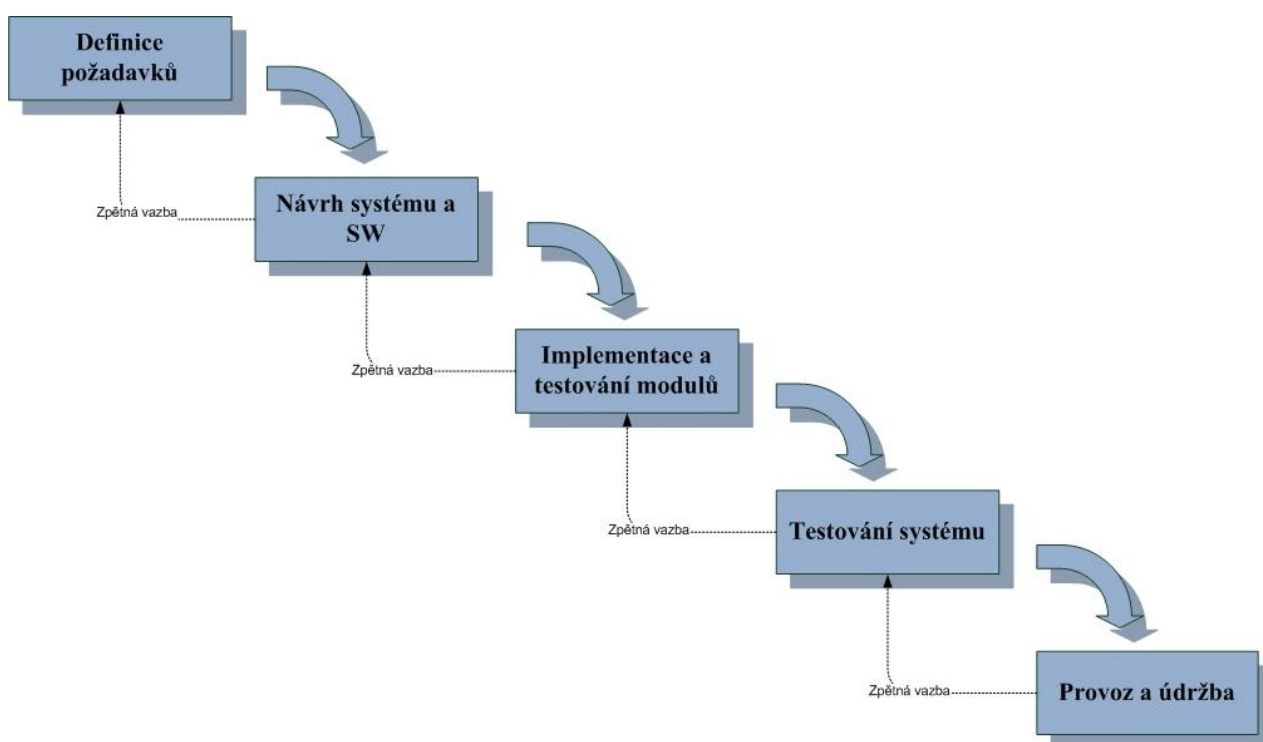
- [12] TULLOCH, Mitch; NORTHUP, Tony; HONEYCUTT, Jerry; Microsoft Windows Vista team. *Microsoft Windows Vista Resouce Kit. Přehled, zavádění systému, správa systému*. Brno : Computer Press, a. s., 2008. 782 s. ISBN 978-80-251-1990-7.
- [13] VRANA, Ivan; RICHTA, Karel. *Zásady a postupy zavádění podnikových informačních systémů: praktická příručka pro podnikové manažery*. Praha : Grada Publishing, a. s., 2005. 188 s. ISBN 80-247-1103-6.
- [14] *Informační systémy* [online], [cit. 2010-07-28]. Životní cyklus projektu informačního systému a jeho zavádění do provozu, Způsoby zavádění informačních systémů do provozu. Dostupné na WWW: <http://homen.vsb.cz/~s1i95/ISVDAS/is/IS_uvod.htm>.
- [15] *Vojenský předpis Vet-1-2 :Veterinární zabezpečení*. Praha : Ministerstvo obrany, 2007. 123 s.
- [16] MACEK, Jakub. *Vývojová fáze informačního systému* [online], 2010 [cit. 2010-09-19]. Vývojová fáze informačního systému. Dostupné z WWW: <<http://newsletter.poski.com/2010-05-28.html>>.
- [17] LACKO, Branislav. Výkonnost softwaru pro somatické řízení. *Automa*, 2008, č. 6 [online]. 2008 [cit. 2010-09-26]. Dostupné z WWW: <<http://www.odbornecasopisy.cz/rec/pdf/37363.pdf>>. ISSN 1210-9592.
- [18] *InterSystém Česká republika* [online]. 2011 [cit. 2011-02-28]. InterSystems Caché – World' fastest database. Dostupné z WWW: <<http://www.intersystems.cz/cache/>>.
- [19] HERBER, Michal; VRABLC, Jiří. *Manuál aplikace VETERIS*. Praha: CompuGROUP Medical Česká republika s.r.o, 2010. 82 s.

Příloha A - Základní modely životního cyklu

Model vodopád

Jedná se o klasický model používaný již v 70. letech. Tento model určuje pořadí jednotlivých etap podle jejich logické návaznosti – definice požadavků, analýza, návrh, implementace a testování. Základem je postupné provádění jednotlivých oddělených a navazujících etap životního cyklu. Přejít do další etapy probíhá až po zdárném dokončení předchozí etapy. Tento model je vhodné použít v případech návrhu IS, kdy zadavatel zná jasně a přesně své potřeby a požadavky. Tedy v případě, že je jednoznačně znám problém a způsob jeho řešení. Za těchto podmínek se jedná o poměrně levný a rychlý postup. Jde o systematický model, který zajišťuje jakost průběžnou kontrolou jednotlivých etap. [11]

Následující obr. A.1 uvádí postup a návaznost jednotlivých etap modelu vodopád.



Obr. A.1 - Model vodopád (zdroj: upraveno na základě [9])

V praxi však lze málokdy řešit postup jednotlivých etap definovaných tímto modelem. První podoby provozuschopného znění IS vznikají až po posledních etapách řešení, je zde tedy nutná trpělivost zadavatele. Pokud si v této etapě zadavatel uvědomí ještě nějaké potřebné úpravy, což se stává v praxi velmi často, nebo pokud se vyskytnou případné chyby, jejich zapracování a odstranění je tím nákladnější na finanční prostředky a čas, podle toho, kolik ukončených předchozích etap je mezi vznikem chyby (úpravy) a zjištěním chyby.[9],[11]

Součástí každého modelu (projektu) je poměrně objemná dokumentace, která obsahuje například definici požadavků, strategické cíle, návrhy, analýzy systému včetně zapracování veškerých změn v průběhu cyklu. Může zde vzniknout i nechuť uživatelů k jejímu studiu z důvodu velkého rozsahu a nesnadnému definování jejich požadavků a potřeb. [2]

Model vodopád má své nevýhody, ale je významně lepší než náhodný, metodicky neřízený přístup k řešení informačního systému. [2]

Model iterativní

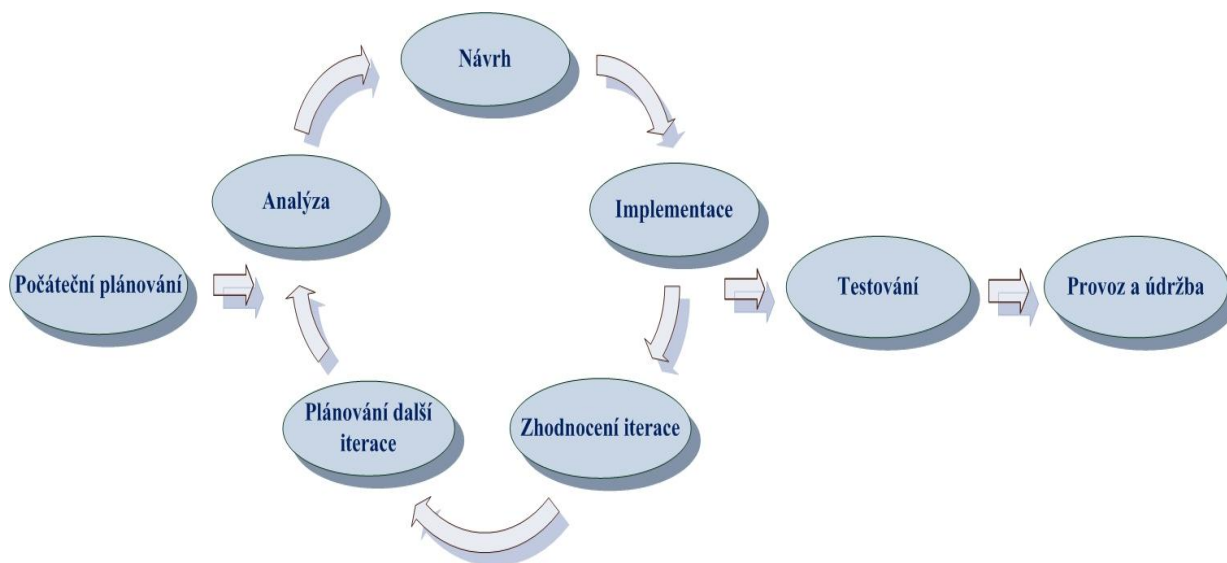
Vychází ve své podstatě z vodopádového modelu, postup však probíhá v několika kratších cyklech (iteracích), které obsahují podobné etapy jako vodopádový model – analýza, návrh, implementace. [9], [16]

Iterativní model vyžaduje intenzivní účast zadavatele projektu. Na začátku jsou sestaveny jeho požadavky a potřeby, které jsou zpracovány v iteraci. Podstatné u tohoto modelu je, že po každé prošlé iteraci je jejím výsledkem použitelná verze informačního IS. Na konci každé iterace by měl tedy zadavatel otestovat všechny nově vytvořené funkce. Případné změny, které většinou vznikají až v průběhu, se zaznamenají a zapracují do následující iterace. Postupným upřesňováním a cizelováním se zajistí, že IS ideálně zcela odpovídá skutečným potřebám a požadavkům zadavatele. Po zapracování všech připomínek zadavatele a odstranění zjištěných chyb je vytvořena, otestována a předána konečná verze IS do rutinního (běžného) provozu. [9], [16]

Postup podle tohoto modelu životního cyklu je zdlouhavější, ale rozhodně kratší než v případě, kdy u vodopádového modelu dojde k dodatečným úpravám, případně k odstraňování chyb. [16]

Výhodou iterativního modelu je společná odpovědnost zadavatele a řešitele (dodavatele) na výsledné podobě tvořeného IS. Díky vysoké míře spoluúčasti zadavatele je výsledkem i velmi dobrá informovanost o průběhu prací a stavu IS. [9]

Model iterativního modelu je zachycen na obr. A.2.

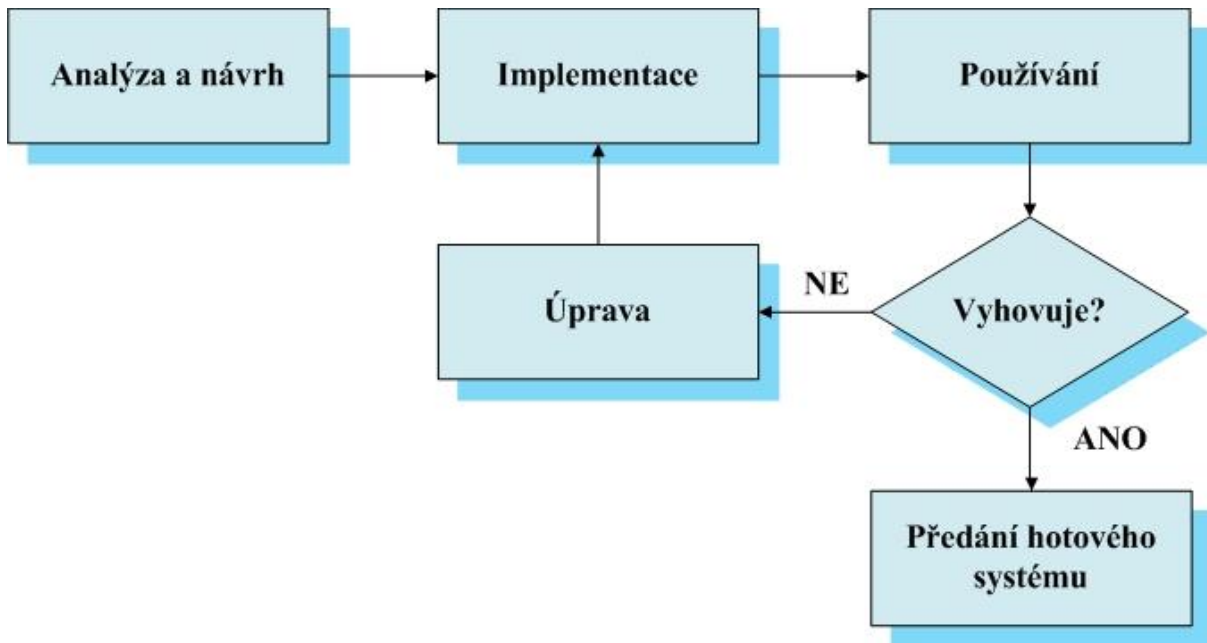


Obr. A.2 - Iterativní model (zdroj: upraveno na základě [9])

Model výzkumník

Základem tohoto modelu je, že během postupu je možné se vracet k předchozím etapám a provádět v nich nově vzniklé potřebné úpravy, jak je názorně zobrazeno na obr. A.3. Oproti iterativnímu modelu se však neprovádějí všechny jeho etapy. IS je postoupen do rutinního provozu po akceptaci zadavatelem. Tento model je velmi přizpůsobivý dodatečným připomínkám a požadavkům uživatele. [9]

Velkou nevýhodou modelu výzkumník je však z tohoto důvodu obtížnost plánování časového průběhu jednotlivých etap, finanční náklady i personální zabezpečení. Problémem je i náročnost zpracování dokumentace. Pokud se dokumentace vytvoří na počátku nebo na konci postupu podle tohoto modelu, nejsou zde zapracovány následné změny jednotlivých etap a nebylo by tedy zřejmé, jak se dospělo ke konečné verzi IS, ani jaký byl stav původní dokumentace. Je tedy velmi důležité, aby se dokumentace vytvářela průběžně. [9]

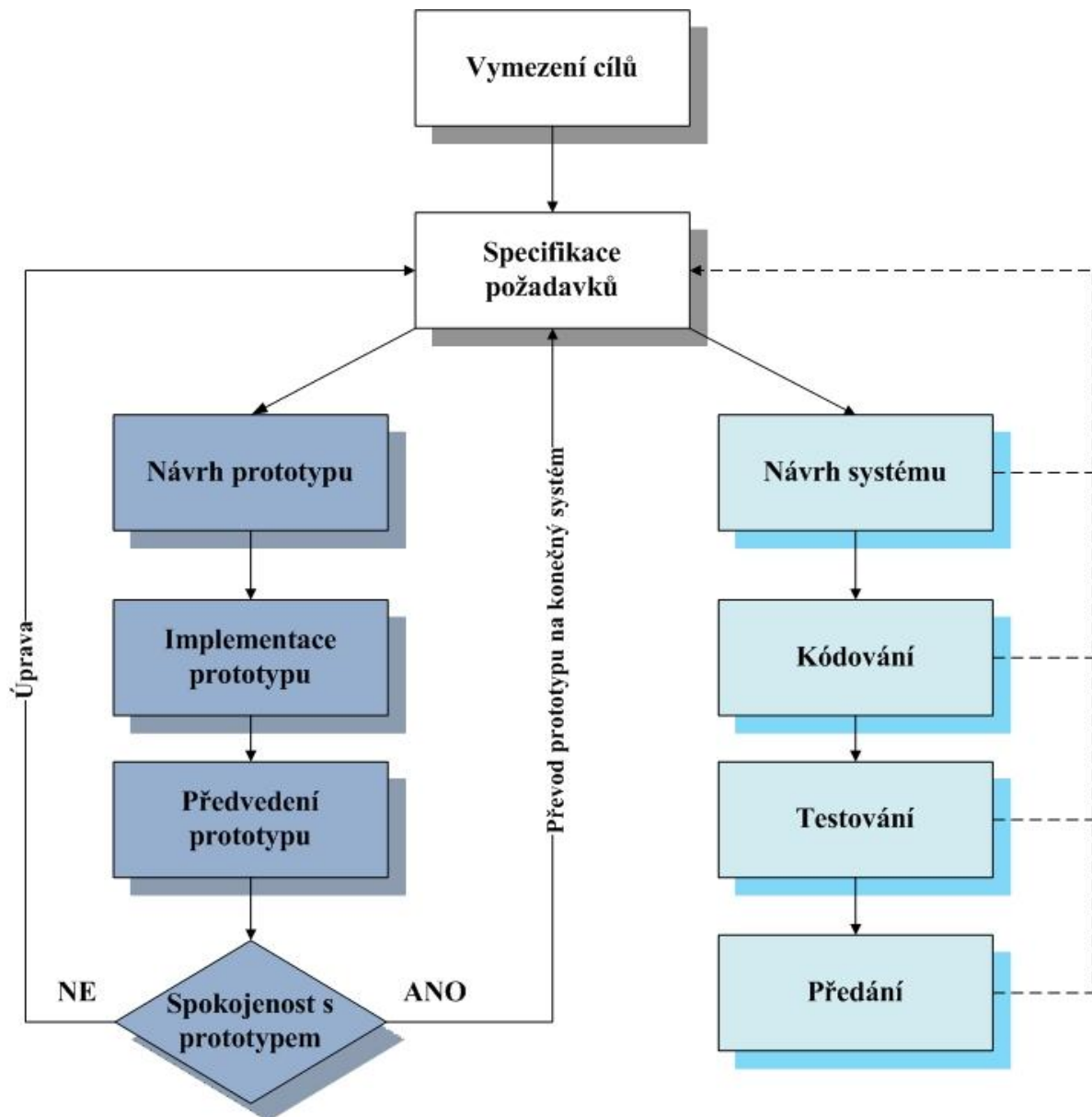


Obr. A.3 - Model výzkumník (zdroj: upraveno na základě [9])

Model prototypový

Jeho největší předností je, že se zde už počítá s možností vzniku dalších požadavků a potřeb zadavatele, případně i změn oproti původnímu zadání včetně jejich zpracování. Lze říci, že prototyp je zavedení zkušební, zjednodušené verze celého nebo pouze části IS. Nejedná se o plně funkční verze, je zde navrženo vnější rozhraní v takové funkčnosti, aby bylo možné ve velmi krátkém čase seznámit postupně zadavatele (uživatelé) pomocí těchto prototypů s prvotními návrhy IS, na kterých si může ověřit, zda splňuje požadovanou funkcionalitu. Na základě jeho připomínek a dodatečných požadavků je prototyp dále upravován, dokud není zadavatel spokojen. Poté dochází k návrhu a tvorbě samotného IS. Prototypy tedy především zabezpečují úlohu komunikace (prostředníka) mezi dodavatelem (analytikem, programátorem...) a zadavatelem (uživatelé). [9], [11] Schéma prototypového modelu je zachyceno na obr. A.4.

Prototypový model tedy umožňuje rychlejší vývoj a tvorbu IS dle představ zadavatele, ale jistou nevýhodou je možná neochota zadavatele (uživatelé) vzdát se prototypu. Za další nevýhodu lze považovat, že při tvorbě rozsáhlých a složitých IS je tento postup relativně náročný, a to nejen z finančních důvodů. Proto se u takových řešení předem stanovují počty prototypů a jejich časový plán. [11]



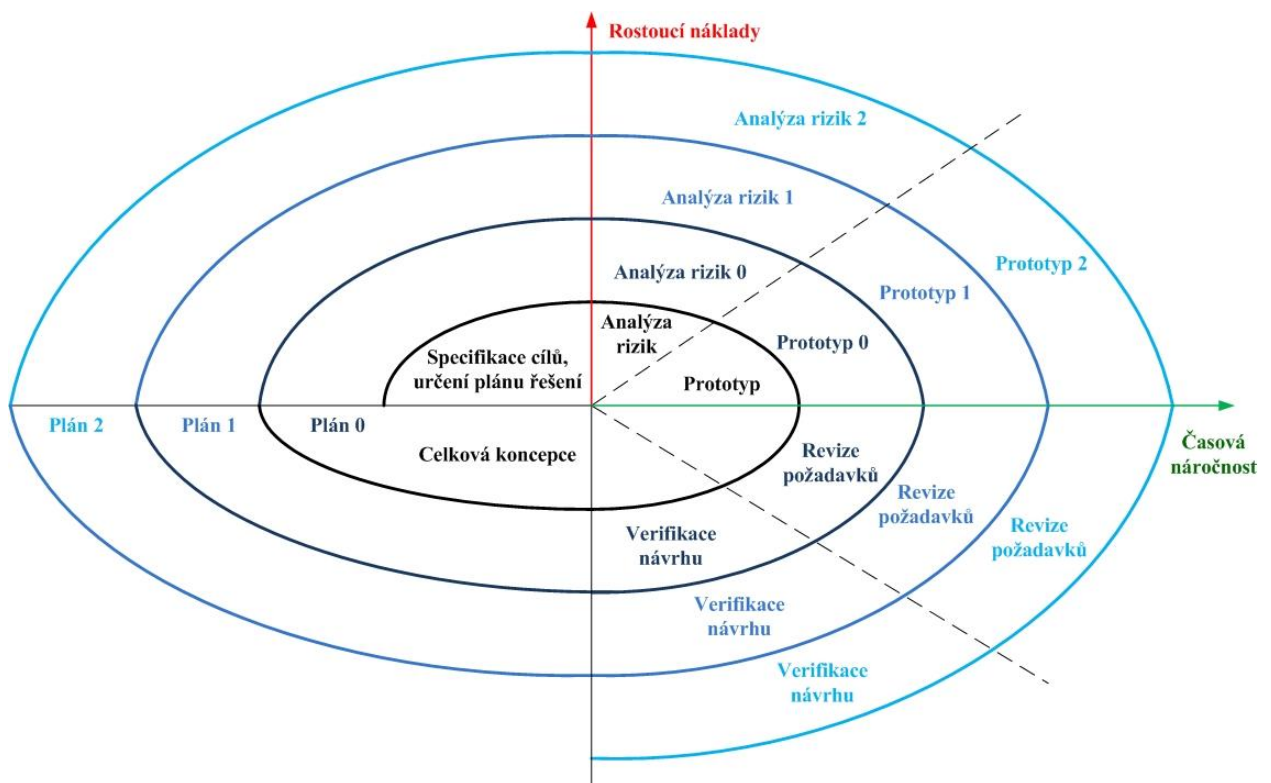
Obr. A.4 - Prototypový model (zdroj: upraveno na základě [11])

Model spirála

Tento model v sobě zahrnuje spojení prototypového modelu a modelu vodopád obohacených o analýzu rizik. Principem je postupné opakování jednotlivých etap, které jsou po testování a ověření obohaceny o další požadavky zadavatele. Každá z etap se skládá ze základních částí, a to ze specifikace cílů a určení plánu řešení, vyhodnocení variant řešení a analýzy rizik, tvorby prototypu dané etapy včetně předvedení a vyhodnocení. Dále pak revize požadavků, testování prototypu a ověření výstupu dané etapy, zda splňuje požadované nároky. [11]

U spirálového modelu je vhodná účast zkušených programátorů a analytiků, kteří budou věnovat značnou pozornost nejen celému průběhu, ale především analýze rizik, neboť se tím předchází možným závažným chybám. Rovněž vyžaduje intenzivní spolupráci zadavatele ke specifikaci jeho podnikových cílů, požadavků a potřeb jak na začátku, tak i v průběhu realizace jednotlivých etap. Dále pak k testování a ověřování jednotlivých prototypů atd. Nevýhodou modelu je, že umožňuje pouze nepřesné plánování termínů i postupných výstupů včetně jejich plnění. [11]

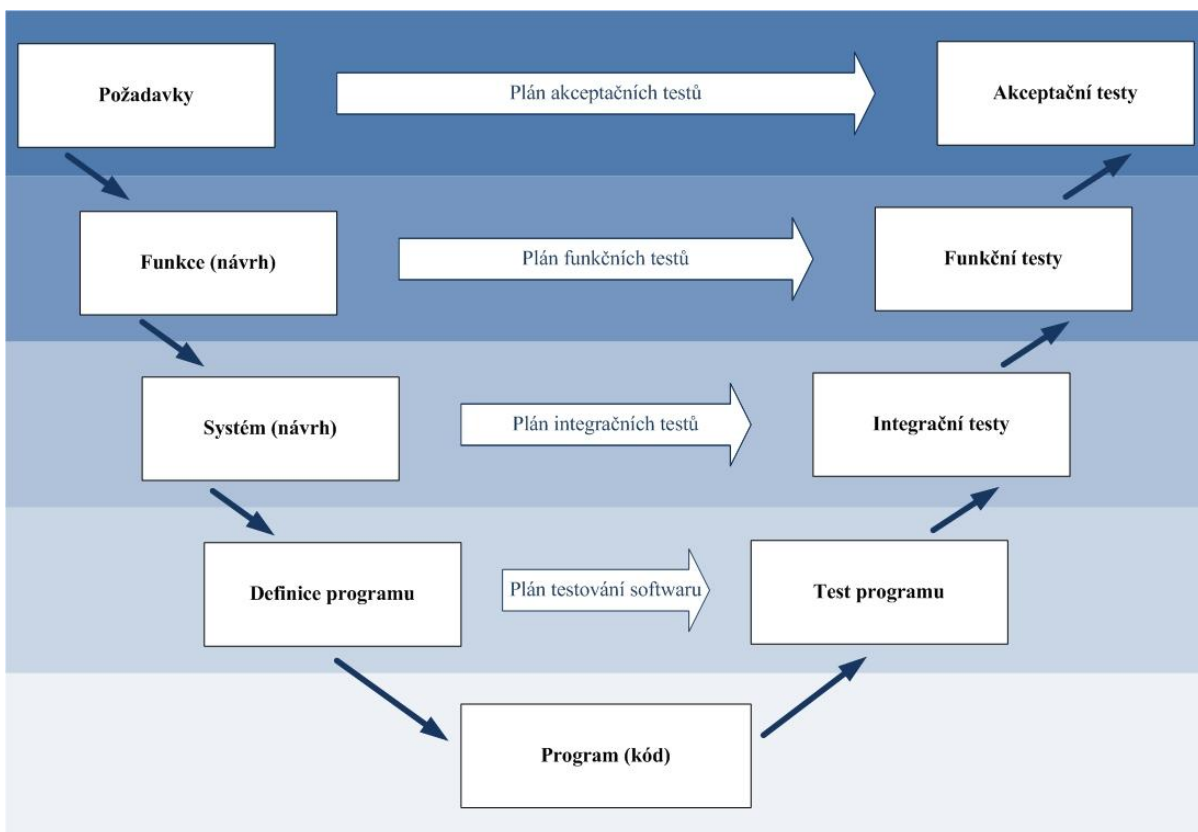
Návaznost jednotlivých etap, časová a finanční náročnost jsou patrné z obr. A.5.



Obr. A.5 - Model spirála (zdroj: upraveno na základě [11])

V-model

Stejně jako všechny předchozí modely tak i tento je grafickým schématem vývoje fází životního cyklu. V-model můžeme brát jako rozšíření vodopádového modelu. Ukazuje vztahy mezi fázemi životního cyklu vývoje IS a jeho přiřazených fází testování. Levá strana modelu znázorňuje postup od požadavků k vytvoření IS, pravá strana pak představuje postup potřebných testů a validací. Každá tato horizontální vrstva modelu vyžaduje jiný typ testů i jiný postup měření. Vertikální osa pak představuje časovou posloupnost a úroveň abstrakce, viz obr. A.6. [17]

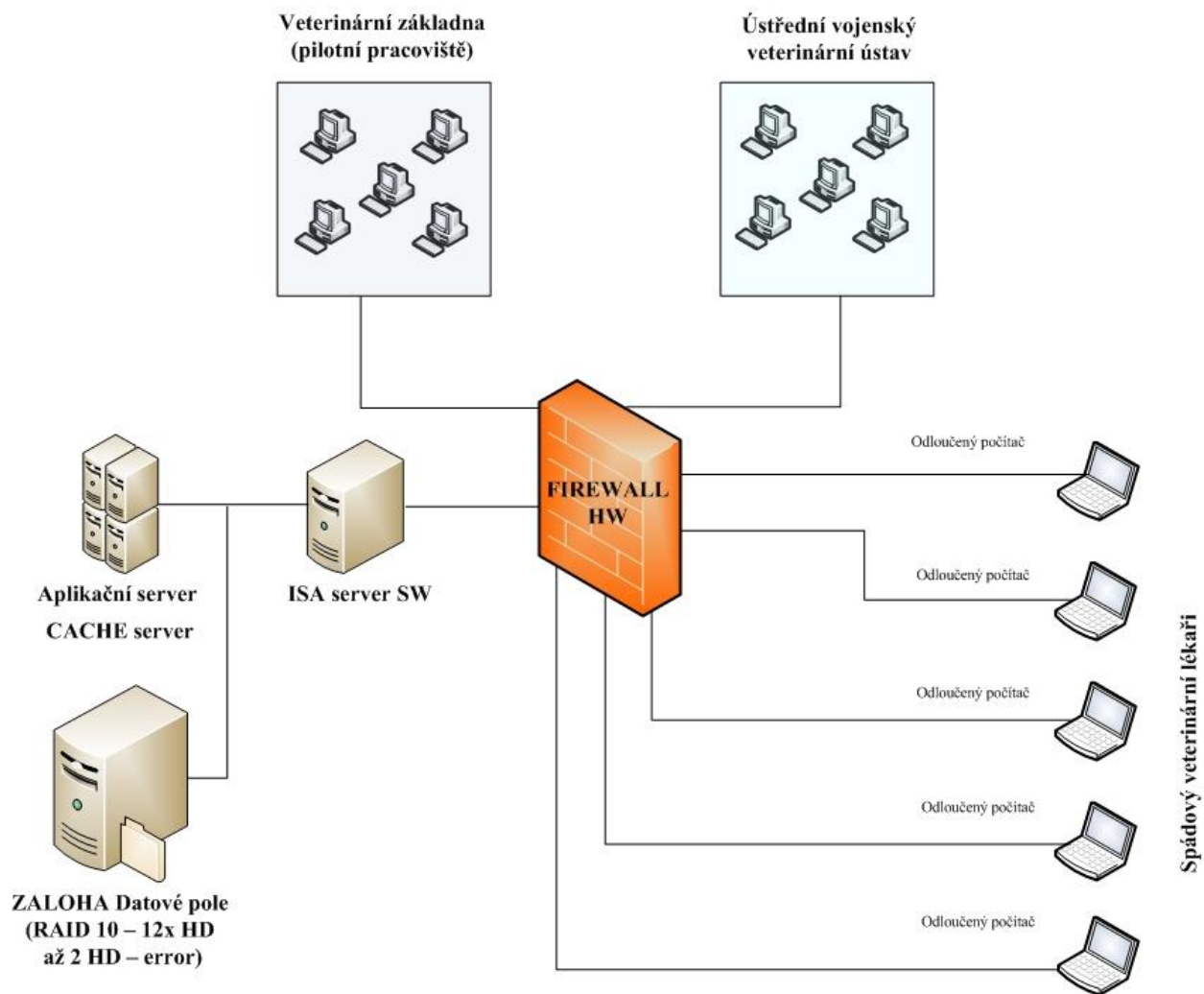


Obr. A.6 - V-model (zdroj: upraveno na základě [17])

Příloha B – Požadavky na informační systém, popis databázového serveru

Pro potřeby záloh a přístupu k datům byl pořízen nový databázový server. Veškeré servery pro potřeby celého ZDRAVIS jsou pořizovány s minimální zárukou 3 let a se smlouvou na pozáruční servis, který zabezpečuje opravu serverů do 24 hodin od nahlášení vyjma sobot a nedělí. Jednotlivé disky jsou majetkem AČR. Protože ve většině případů obsahují citlivá data, nesmí opustit toto prostředí ani v případě jakékoliv opravy, která musí být prováděna na místě. V případě neopravitelné závady se likvidují zvláštním způsobem stanoveným zvláštním interním předpisem. Nejprve se provádí mazání dat bez možnosti obnovení pomocí vhodného programu, například Universal Shredder, který je snadno ovladatelný a zabezpečuje dodržení bezpečnostní normy C2. Jedná o několikanásobné přepsání celého obsahu disku přesně definovanými sekvencemi binárních dat, které znemožňuje jejich obnovu. Po výmazu přichází vlastní fyzická likvidace, kdy se jednotlivé plotny disků vymontují a provede se jejich fyzické sešrotování. Databázovým prostředím je platforma databáze CACHE firmy Intersystem. Dle specifikace dodavatele [18] je CACHE „inovativní objektová databáze, zpracovávající SQL mnohem rychleji než relační databáze. Její předností je především garantovaný výkon. Jedná se o postrelační databázový systém spojující jedinečným způsobem možnosti přístupu k údajům a to SQL, objektový přístup a vícerozměrný přístup k datovým polím. Všechny tyto metody lze použít zároveň, i při současném přístupu k jedné položce. Výhodou je i snadná škálovatelnost databázového systému.“

Server aplikace VETERIS je umístěn v centrální serverovně ZDRAVIS, která je od intranetu oddělena vlastním routerem s firewallem a potom softwarovým firewallem na bázi software Microsoft ISA server 2006. Antivirová ochrana je řešena softwarem AVG s centrální správou. Místnost serverů je za protipožárními bezpečnostními dveřmi a hlídána elektronickým zabezpečovacím a požárním systémem. Hardwarová konfigurace tohoto serveru je HP s dvěma procesory Intel Xeon 3,2 GHz s 3,5 GB RAM. Má osazeny dva disky 450 GB v RAID 1 (zrcadlení) pro dosažení vyšší úrovně zabezpečení. Velikost disků je vzhledem k množství ukládaných dat dostatečná. Zálohy se provádějí s 24hodinovým intervalem. Ukládají se na určené diskové pole serveru pro zálohy dat. Pole má ochranu úrovně RAID10 a mohou v něm vypadnout i dva disky současně bez ztráty dat v poli. Pásková zařízení se přestala používat po špatných provozních zkušenostech, kdy při rutinních kontrolách opakovaně nešly jednotlivé zálohy přečíst, a navíc je obnova dat relativně pomalá. Schematické znázornění zapojení a zabezpečení je uvedeno na výše uvedeném obr. B.1.



Obr. B.1 - Plánované zapojení a zabezpečení aplikace VETERIS (zdroj: vlastní)

Příloha C – Vzor harmonogramu implementace aplikace VETERIS

Tabulka 1 : Vzor harmonogram implementace aplikace VETERIS (zdroj: vlastní)

ID	Kód WBS	Název úkolu	Doba trvání	Zahájení	Dokončení	Předchůdci	Názvy zdrojů
1	1	PLÁN IMPLEMENTACE VETERINÁRNÍHO IS	212 dny	1.3.10	31.12.10		
2	1.1	FÁZE PŘÍPRAVNÁ A ZAHAJOVACÍ	15 dny	1.3.10	22.3.10		
3	1.1.1	Provéření zabezpečení stávající HW a SW infrastruktury	10 dny	1.3.10	15.3.10		
4	1.1.1.1	Kontrola, doplnění	10 dny	1.3.10	15.3.10		Pracovníci IT
5	1.1.2	Příprava dat	10 dny	1.3.10	15.3.10		
6	1.1.2.1	Pilotní pracoviště - kontrola duplicit, doplnění, vyčištění	5 dny	1.3.10	8.3.10		Pracovníci IT
7	1.1.2.2	Druhé pracoviště - kontrola duplicit, doplnění, vyčištění	5 dny	8.3.10	15.3.10	6	Pracovníci IT
8	1.1.3	Dodávka DB serveru	2 dny	15.3.10	17.3.10		
9	1.1.3.1	Instalace	1 den	15.3.10	16.3.10	7	Pracovníci IT ;Server[1 ks]
10	1.1.3.2	Zapojení do stávající LAN	1 den	16.3.10	17.3.10	9	Pracovníci IT
11	1.1.4	Dodávka přenosných počítačů	3 dny	15.3.10	18.3.10		
12	1.1.4.1	Instalace	2 dny	15.3.10	17.3.10	7	Pracovníci IT ;Notebook[5 ks]
13	1.1.4.2	Předání na Spádová pracoviště	1 den	17.3.10	18.3.10	12	Pracovníci IT
14	1.1.5	Zhodnocení připravenosti zadavatele	1 den	18.3.10	19.3.10	13;10,4	Odb.konzultant;Manažer projektu;Pracovníci IT
15	1.1.6	Zhodnocení připravenosti dodavatele aplikace	1 den	19.3.10	22.3.10	14	Dodavatel IT;Manažer projektu;Pracovníci IT
16	1.1.7	Dodávka prototypu veterinární aplikace	0 dny	22.3.10	22.3.10	15	Dodavatel IT
17	1.2	FÁZE PROVÁDĚCÍ	189 dny	22.3.10	20.12.10		
18	1.2.1	Instalace na jednu pracovní stanici 1.ambulance pilotního pracoviště	64 dny	22.3.10	21.6.10		
19	1.2.1.1	Instalace	1 den	22.3.10	23.3.10	16	Pracovníci IT
20	1.2.1.2	Migrace dat (ostrá)	1 den	23.3.10	24.3.10	19	Pracovníci IT
21	1.2.1.3	Školení klíčových uživatelů	2 dny	24.3.10	26.3.10	20	Dodavatel IT
22	1.2.1.4	Testování, analýza funkčnosti a zálohování	40 dny	26.3.10	24.5.10	21	Pracovníci IT
23	1.2.1.5	Optimalizace nastavení systému + update	20 dny	24.5.10	21.6.10	22	Dodavatel IT
24	1.2.2	Rozšíření instalace - pilotní pracoviště (peer-to-peer)	36 dny	21.6.10	12.8.10		
25	1.2.2.1	Instalace 2 ambulance	1 den	21.6.10	22.6.10	23	Pracovníci IT [50%]
26	1.2.2.2	Instalace rentgen	1 den	21.6.10	22.6.10	25SS	Pracovníci IT [50%]
27	1.2.2.3	Instalace kancelář	1 den	21.6.10	22.6.10	26SS	Pracovníci IT [50%]
28	1.2.2.4	Testování, analýza funkčnosti a zálohování	15 dny	22.6.10	15.7.10	27;26,25	Pracovníci IT
29	1.2.2.5	Optimalizace nastavení systému + update	20 dny	15.7.10	12.8.10	28	Dodavatel IT
30	1.2.3	Migrace dat na databázový server	33 dny	12.8.10	29.9.10		
31	1.2.3.1	Migrace dat - pilotní pracoviště a připojení	1 den	12.8.10	13.8.10	29	Pracovníci IT
32	1.2.3.2	Migrace dat - druhé pracoviště a připojení	1 den	13.8.10	16.8.10	31	Pracovníci IT
33	1.2.3.3	Školení všech koncových uživatelů	1 den	16.8.10	17.8.10	32	Pracovníci IT
34	1.2.3.4	Testování, analýza funkčnosti a zálohování	20 dny	17.8.10	14.9.10	33	Pracovníci IT
35	1.2.3.5	Optimalizace nastavení systému + update	8 dny	14.9.10	24.9.10	34	Dodavatel IT
36	1.2.3.6	Zátěžové a ověřovací testy připojení a síťové funkčnosti	2 dny	24.9.10	29.9.10	35	Pracovníci IT
37	1.2.4	Centralizace dat	19 dny	29.9.10	26.10.10		

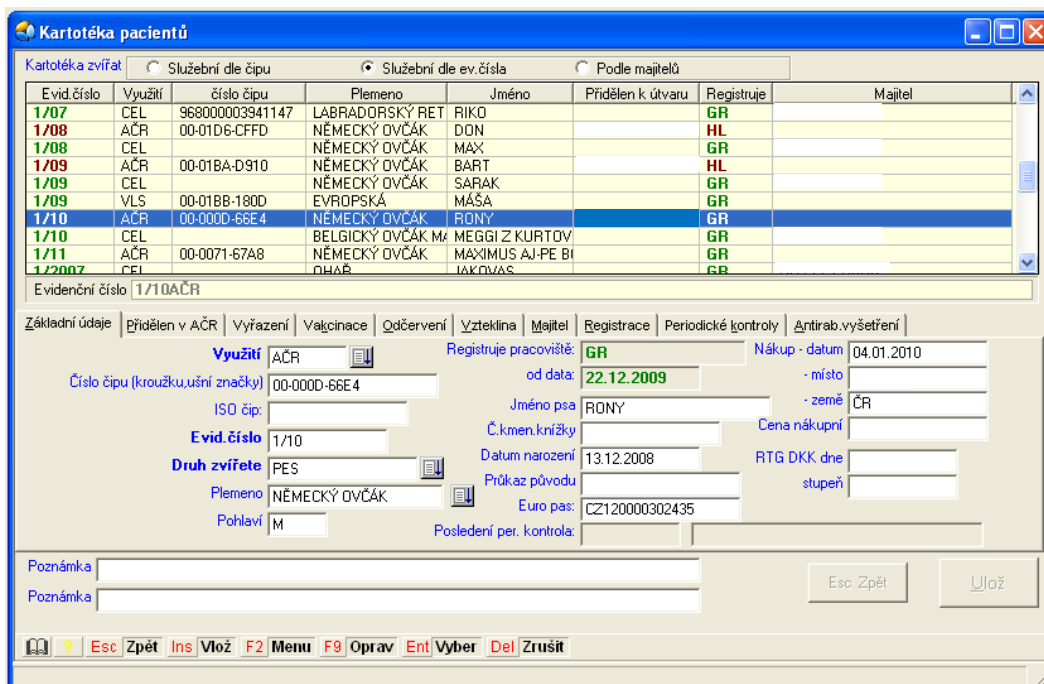
ID	Kód WBS	Název úkolu	Doba trvání	Zahájení	Dokončení	Předchůdci	Názvy zdrojů
38	1.2.4.1	Příprava sloučení dat	10 dny	29.9.10	13.10.10	36	Dodavatel IT
39	1.2.4.2	Sloučení bází dat jednotlivých pracovišť	0 dny	13.10.10	13.10.10	38	Dodavatel IT
40	1.2.4.3	Připojení pilotní pracoviště	1 den	13.10.10	14.10.10	39	Pracovníci IT
41	1.2.4.4	Připojení druhé pracoviště	1 den	13.10.10	14.10.10	40SS	Pracovníci IT
42	1.2.4.5	Zátěžové a ověřovací testy připojení a síťové funkčnosti	8 dny	14.10.10	26.10.10	41,40	Pracovníci IT
43	1.2.5	Rozšíření na ostatní spádová pracoviště	27 dny	26.10.10	6.12.10		
44	1.2.5.1	1. Spádové pracoviště	1 den	26.10.10	27.10.10	42	Pracovníci IT [50%]
45	1.2.5.2	2. Spádové pracoviště	1 den	26.10.10	27.10.10	44SS	Pracovníci IT [50%]
46	1.2.5.3	3. Spádové pracoviště	1 den	26.10.10	27.10.10	45SS	Pracovníci IT [50%]
47	1.2.5.4	4. Spádové pracoviště	1 den	26.10.10	27.10.10	46SS	Pracovníci IT [50%]
48	1.2.5.5	5. Spádové pracoviště	1 den	26.10.10	27.10.10	47SS	Pracovníci IT [50%]
49	1.2.5.6	Školení spádových uživatelů	1 den	27.10.10	29.10.10	48;47;46;45;	Pracovníci IT
50	1.2.5.7	Zátěžové, ověřovací testy připojení k centrální bázi dat	20 dny	29.10.10	29.11.10	49	Pracovníci IT
51	1.2.5.8	Testy přenosu dat - odloučený počítač	5 dny	29.11.10	6.12.10	50	Pracovníci IT
52	1.2.6	Finální optimalizace nastavení systému + update	10 dny	6.12.10	20.12.10	51	Dodavatel IT
53	1.3	FÁZE UKONČENÍ A PŘEDÁNÍ DO RUTINÍHO PROVOZU	8 dny	20.12.10	31.12.10		
54	1.3.1	Příprava smlouvy o poskytování SW služeb	3 dny	20.12.10	23.12.10	52	Manažer projektu[50%]
55	1.3.2	Zhodnocení způsobilosti aplikace pro rutinní provoz	2 dny	20.12.10	22.12.10	52	Odb.konzultant;Dodavatel IT;Manažer projektu[50%];Pracovníci IT
56	1.3.3	Závěrečné potvrzení akceptace řešení	1 den	22.12.10	23.12.10	55	Odb.konzultant;Manažer projektu[50%];Pracovníci IT
57	1.3.4	Finální potvrzení akceptačního protokolu	1 den	23.12.10	27.12.10		
58	1.3.4.1	Potvrzení	1 den	23.12.10	27.12.10	56	Vedení[35%]
59	1.3.4.2	Fakturace	1 den	23.12.10	27.12.10	58SS	Finanční
60	1.3.5	Podpis smlouvy o poskytování SW služeb	1 den	23.12.10	27.12.10	54	Vedení[35%]
61	1.3.6	Ukončení zkušebního provozu, rutinní provoz všech pracovišť	0 dny	27.12.10	27.12.10	58;59;60	Vedení[30%]
62	1.3.7	Zhodnocení procesu implementace	2 dny	27.12.10	29.12.10	61	Odb.konzultant;Pracovníci IT
63	1.3.8	Zpracování zprávy o zavádění aplikace	2 dny	29.12.10	31.12.10	62	Odb.konzultant
64	1.3.9	Ukončení projektu	0 dny	31.12.10	31.12.10	63	Vedení

Příloha D - Popis aplikace VETERIS

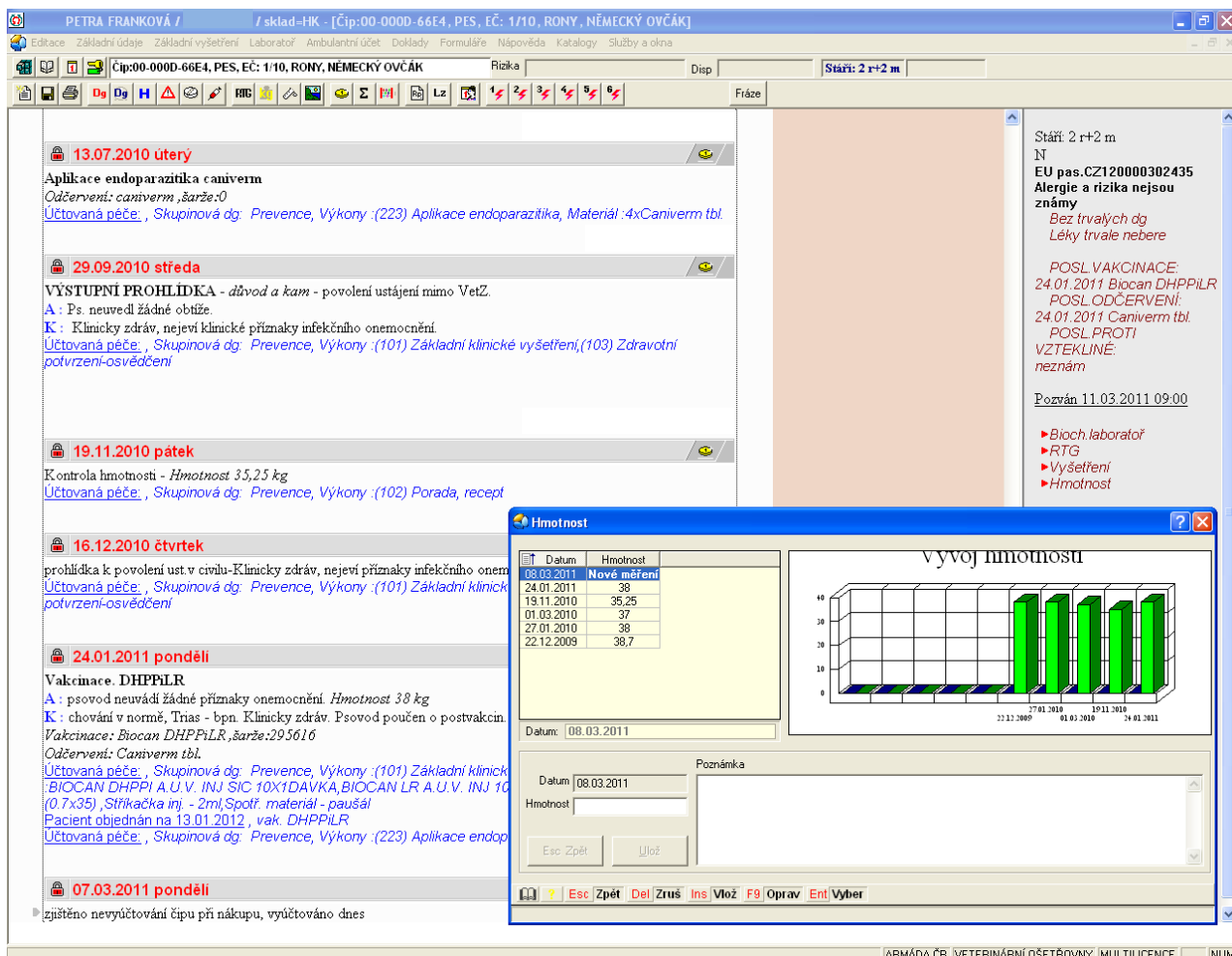
Mezi hlavní funkce aplikace VETERIS patří zabezpečení celoarmádního systému veterinární péče, postupné vytváření podmínek pro komunikaci s informačními systémy státní veterinární správy, zabezpečení integrity veterinárních dat služebních zvířat vojenské správy a vytvoření centrálního registru veterinárních dat služebních zvířat vojenské správy. Hlavní přínosy aplikace VETERIS se předpokládají ve zvýšení kvality a produktivity procesu velení a řízení vojenské veterinární služby a jejího informačního zabezpečení. Spočívají v kvalitativní a kvantitativní změně zpracování informací a přístupu k nim na všech úrovních, vytvoření výchozích předpokladů pro zkvalitnění rozhodovacích procesů, postupné zavádění bez dokladového styku, větší spolehlivost a průkaznost informací, zvýšení objektivity velení a řízení ve všech úrovních i vytvoření zpětných vazeb pro efektivní ovlivnění činnosti vojenských veterinárních lékařů. Předpokládaná efektivnost IS v oblasti veterinární služby spočívá zejména ve zlepšení odborné úrovně poskytované léčebné péče, zvýšení hodnověrnosti analýz, statistik a hlášení, snížení nákladů na efektivnější využívání sil a prostředků a v další řadě dílčích přínosů ve veterinární službě narůstajících postupným osvojováním aplikace VETERIS uživateli.

Po zavedení centrální databáze ošetřovaných zvířat mají všechna připojená pracoviště okamžitě aktuální záznamy o každém ošetřovaném zvířeti. Definicí uživatelských práv je zajištěno, že každý uživatel může nahlížet na medicínské záznamy zvířat bez ohledu na to, na kterém pracovišti byly zápisy provedeny. Opravy či výmaz některých údajů je však povolen jen autorovi záznamu a jsou i situace, kdy je toto zapovězeno i jemu. Každý uživatel také smí přistupovat jen ke stanoveným skladům a finančním dokladům. Zásadně není dovolena manipulace se skladem a finančními doklady jiného veterinárního pracoviště. Nadřízený stupeň má přístup k těmto údajům dovolen, avšak pouze za účelem čtení a vyhodnocení. [19]

Program umožňuje získat mnoho různých přehledů s různými výběrovými podmínkami za libovolné období, a tím umožňuje sledovat úroveň poskytované péče z hlediska odborného i finančního. Opět se zde rozlišuje přihlášený uživatel a podle jeho přístupových práv může provádět přehledy a rozbory jen za sebe, za celé své pracoviště. Pouze nadřízený stupeň smí provádět tyto přehledy a rozbory za jiné pracoviště, za skupinu pracovišť nebo za všechny dohromady. [19] Ukázka uživatelského rozhraní je na níže uvedených obrázcích.



Obr. D.1 - Ukázka aplikace VETERIS (zdroj: vlastní)



Součástí programu jsou funkce pro vedení skladu léků a materiálu, tedy příjem do skladů, hromadný výdej, jednotlivý výdej pro ošetřované zvíře, sledování stavu zásob jednotlivých položek v naturálních jednotkách i v penězích, upozorňování na nízké stavy zásob, sledování blízkých se expirací, záznamu inventur a především funkce odloučený počítač. [19]

Tato funkce tedy umožňuje veterinárním lékařům pracovat s aplikací VETERIS i na vzdáleném pracovišti, útvaru, zařízení se služebními zvířaty bez připojení k centrálnímu registru služebních zvířat. Funkce se spouští z nabídky *Služby - Odloučený počítač*. Po spuštění se vybere záložka podle toho, zda je třeba provést export dat (na přenosné médium), nebo import dat (nahrání z přenosného média do hlavního počítače). Zadá se datum počátku a konce období, za které je třeba data přenést, a stiskne se tlačítko *Vytvoř soubor*. Při exportu dat program předvyplní název souboru a uživatel stanoví, kam se má výsledný soubor uložit. Při nahrávání dat (importu) je postup obrácený – nejprve se stiskne tlačítko *Otevři soubor* a označí se soubor k přenosu (program zobrazí podrobnou identifikaci datového souboru) a pak se spustí nahrávání. V praxi se tímto způsobem vyexportuje celá databáze služebních zvířat, popřípadě pouze skupina, které se bude ošetřeny v terénu týkat. Tuto databázi či skupinu dat si uživatel naimportuje na mobilní počítač. Po práci v terénu opět přes tuto funkci vyexportuje data vzniklá mimo systém. Funkce si hlídá pomocí časových razítek jednotlivé záznamy, takže se nemůže stát, když se náhodou vyexportují data, která jsou již v centrální databázi, že by došlo k jejich duplikaci či nechtěnému přepsání či zrušení. Systém při importu dat kontroluje časová razítka a u záznamů se stejným časovým údajem přenos vynechá.