

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ



**e-learning: specifikace omezení objektů
systému**

Ročníkový projekt

2006

Michal Drahoš

„Originál zadání“

Prohlášení

Prohlašuji, že jsem tento ročníkový projekt vypracoval samostatně pod vedením Kreslíkové Jitky, RNDr., Csc., UIFS FIT VUT

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

podpis

Michal Drahoš
2.5.2006

Poděkování

Děkuji vedoucí ročníkové práce Kreslíkové Jitce, RNDr., CSc., UIFS FIT VUT za poskytnuté rady a doporučenou literaturu.

Abstrakt:

Tento projekt se zabývá problematikou e-learningu (elektronického vzdělávání). Vysvětluje co tento pojem znamená, popisuje jeho výhody i nevýhody, taktéž k čemu slouží a kde se využívá. Úkolem bylo vytvořit aplikaci vhodnou k elektronickému vzdělávání, která by objasňovala specifikaci omezení objektů systému. To znamená vytvořit aplikaci ve které by byla vysvětlena problematika OCL a to v takovém formátu, který by odpovídal struktuře a formě aplikace užívající se k elektronickému vzdělávání. Seznámil jsem se s jazyky OCL i UML 2.0. Prostudoval prostředí CASE Rational Rose a také se obeznámil s problematikou elektronického vzdělávání. Těchto poznatků jsem využil při vytváření e-learningové aplikace.

Klíčová slova:

Elektronické vzdělávání, OCL, UML 2.0, CASE Rational Rose, web-based training, www stránky, skriptovací jazyk PHP, HTML, diagram

Abstract:

This project is dealing with the problem of e-learning. It explains meaning of this term, describes its advantages and disadvantages, what is e-learning used for and where it is used. The objectives were to create an application suitable for electronic learning, which could illustrate specifications of restrictions of the system objects. This means to create an application, where is explained the problem of OCL in that kind of format, which reacts the structure and the form of application used for e-learning. I meet the program languages of OCL and UML 2.0. I read over the CASE Rational Rose environment and meet the e-learning problem, too. These are the knowledges, which I used while I was creating my e-learning application.

Keywords:

E-learning, OCL, UML 2.0, CASE Rational Rose, web-based training, www sites, PHP language, HTML, diagram

Obsah:

1. ÚVOD	8
2. E-LEARNING	9
2.1 Historie	9
2.2 E-learning: co, proč a jak.....	9
2.3 Distribuce.....	10
3. OCL (UML 2.0) a CASE Rational Rose.....	11
3.1 Co je to UML.....	11
3.1.1 Historie a standardy.....	12
3.1.2 K čemu slouží jazyk UML.....	12
3.2 Co je to OCL.....	13
3.2.1 Proč používat OCL.....	13
3.2.2 Případy použití OCL.....	14
3.3 CASE Rational Rose	15
3.3.1 Vlastnosti.....	16
3.3.2 Ukázka prostředí CASE Rational Rose a diagramů vněm modelovaných.....	16
4. OBECNÁ STRUKTURA ELEKTRONICKÝCH DOKUMENTŮ.....	19
5. FORMÁT UČEBNÍHO TEXTU.....	21
5.1 Vzhled a popis aplikace.....	21
6. PŘÍKLADY VYTVOŘENÉ V PROSTŘEDÍ RATIONAL ROSE.....	25
7. ZHODNOCENÍ POUŽITELNOSTI TEXTU VE VÝUCE A ZÁVĚR.....	26
SEZNAM POUŽITÉ LITERATURY	27

1. Úvod

Tento ročníkový projekt je zaměřen na vytváření aplikace elektronického vzdělávání, neboli e-learningové aplikace popisující specifikaci omezení objektů systému. Tímto se myslí vytvoření prostředí ve kterém je srozumitelně vysvětlena a ukázána problematika OCL.

V první kapitole této dokumentace si povíme něco málo o tom co je to vlastně e-learning a jaké je jeho využití. Druhá kapitola podrobněji popisuje OCL (UML 2.0) a také nástroj CASE Rational Rose, který UML podporuje. Ve třetí kapitole se seznámíme s obecnou strukturou elektronických dokumentů vhodných pro samostatné studium. Čtvrtá a pátá kapitola popisuje zpracovaný učební text vysvětlující problematiku specifikací omezení, doplněný příklady vytvářenými v prostředí CASE Rational Rose a kontrolními otázkami. Poslední kapitola se zabývá zhodnocením použitelnosti textu ve výuce.

2. E-learning

V této kapitole se seznámíme s problematikou e-learningu. Nejprve si povíme co se vlastně za pojmem e-learning skrývá. E-learning nebo také elektronické vzdělávání je v podstatě výuka podporovaná počítačem. Je to proces sdílení informací a poskytování vzdělávacích a výcvikových aktivit v elektronické podobě. E-learning můžeme definovat jako výučování prováděné na osobním počítači (PC) prostřednictvím internetu, intranetu nebo pomocí CD/DVD ROM mechanik a médií.

2.1 Historie

Pojem e-learning je „poměrně“ nový. Už v době sálových počítačů se tyto počítače využívali i k výuce. První výukové programy byly textové a pracovali na principu testu, otázky a odpovědi.

Většího rozšíření bylo dosaženo v letech 1984 – 1993, v takzvané multimediální éře. V této době se počítače začali rozšiřovat do domácností a škol a ostatní institucí. Počítače Macintosh, nebo osobní počítače s operačním systémem Windows 3.1 obsahující CD-ROM a také například program Powerpoint sloužící k prezentacím. Tyto technologie přispěly k širšímu používání e-learningu. Důležitou úlohu v této době hrály CD-ROM které umožňovaly přenášení výukového programu kdykoliv a kamkoliv.

Dalším důležitým milníkem je období 1994-1999 kdy se začal rozšiřovat internet. V této době začal e-learning využívat Webu, emailu, mediálních přehrávačů. Začala se objevovat výuka pomocí emailu, intranet CBT(Computer-Based Training = výuka pomocí počítačů) s textem a jednoduchou grafikou a WBT(Web-Based Training = výuka pomocí počítače připojeného do sítě). Toto období můžeme nazvat první vlnou e-learningu.

Rozvíjením technologií jako jsou JAVA/IP síťové aplikace, dále také rozšíření Webových aplikací, streaming videa a zvuku nám umožnilo posunout hranici e-learningu ještě výše. V dnešní době můžeme spojit elektronickou výuku s výukou s lektorem v reálném čase a to vše prostřednictvím internetu. Toto období od roku 2000 bývá označováno jako druhá vlna e-learningu.

2.2 E-learning: co, proč, jak

E-learning je definovaný jako výuka zprostředkovaná počítačem pomocí CD-ROM, internetu nebo intranetu s následujícími vlastnostmi:

- Zahrnuje obsah týkající se učebních cílů

- Používá instrukční metody jako příklady a cvičení, sloužící k pochopení látky
- Používá mediální prvky jako slova a obrázky k dodání obsahu
- Ke zvýšení organizačních vlastností staví nové znalosti a dovednosti provázaně s individuálními učebními cíly

Nyní si popíšeme co rozumíme a co se vztahuje k pojmům *co*, proč a *jak*:

Pod slovem *co* se skrývá, co obsahují kurzy e-learningu. Ty obsahují jak obsahovou část (to jsou informace) tak i instrukční metody (to jsou techniky) jak pomoci lidem se učit

Pod slovíčkem *jak*, je pospáno jak je e-learning poskytován. E-learning je prezentován pomocí počítače, a to pomocí slov *jak* v tištěné tak i v mluvené formě, také pomocí obrázků ať už jako fotografií, ilustrací, animací či videa

Proč nám říká proč vlastně využíváme e-learningu. E-learningové jsou tvořeny tak aby pomohli studentovi dosáhnout osobního učebního cíle nebo splnili svoji práci způsobem který zvyšuje spodní hranici cílů organizace.

Když si rozebereme slovo e-learning zjistíme že „e“ značí to „jak“, způsob jakým jsou kurzy prováděny což jest elektronickým způsobem. „learning“ zase značí „co“ a „proč“ to ještě že se jedná o způsob učení.

2.3 Distribuce

E-learning je šířen pomocí CD-ROM, internetu, intranetu, nebo i pomocí dalších prostředků (telefon, atd.) nebo kombinováním těchto prostředků. V dnešní době je nejrozšířenější způsob poskytování e-learningu internet, intranet a také se používají přenosová média jako DVD-ROM.

Každý z těchto způsobů má nějaké výhody a nevýhody. Například pokud využíváme e-learning šířený pomocí internetu je nutné mít možnost nejlépe trvalého připojení. Tato nutnost odpadá v případě využití CD-ROM či DVD-ROM. V těchto případech máme kompletní aplikaci na médiu a v počítači. Je zde ale problém týkající se aktualizací, kdy v případě změn je nutno si pořídit médium s aktualizací a aplikaci přeinstalovat. Kombinací těchto možností je že je aplikace dodána na CD-ROM a z internetu jsou stahovány aktualizace. Tento přístup je výhodný v tom, že i přesto že musíme mít k připojení k internetu, nemusí být tak rychlé a trvalé jako v případě že kurz probíhá čistě na internetu. Výhody WBT (Web-Based Training = výuka pomocí počítače připojeného do sítě) jsou snadná aktualizace, komunikace a distribuce. Jako hlavní výhoda CBT (Computer-Based Training = výuka pomocí počítačů) je, že nemusíme mít připojení k internetu, protože i v dnešní době ne každý připojení k internetu má. V dnešní době kdy přenosová média mají kapacitu i 8,5 GB je možné takto přenášet velké množství informací (video ve vysokém rozlišení), což přes internet není až tak možné.

3. OCL (UML 2.0) a CASE Rational Rose

3.1 Co je to UML

Jazyk UML (Unified Modeling Language, unifikovaný modelovací jazyk) je univerzální jazyk pro vizuální modelování systému. Je to standardní průmyslový jazyk pro specifikaci, vizualizaci, vytváření, a dokumentaci artefaktů softwarových systémů. Přestože je nejčastěji spojován s modelováním objektově orientovaných softwarových systémů, má mnohem širší využití, což vyplývá z jeho zabudovaných rozšiřovacích mechanismů.

Byl navržen, aby spojil nejlepší existující postupy softwarového inženýrství a modelovacích technik. Explicitně je navržen tak, aby jej mohli implementovat všechny nástroje CASE (computer-aided software engineering). Diagramy vytvořené v jazyku UML jsou srozumitelné pro lidi, ale navíc je mohou snadno interpretovat i CASE programy.

3.1.1 Historie a standardy

Jazyk UML vytvořili metodici Grady Booch, Ivar Jacobson a Jim Rumbaugh ve společnosti Rational Software. Jejich cílem bylo sjednotit různé stávající systémy do jediného nejlepšího modelovacího jazyka. V roce 1996 navrhlo sdružení OMG (Object Management Group) specifikaci RFP (Request for Proposal) pro objektově orientovaný jazyk pro vizuální modelování, v němž jako standard navrhlo jazyk UML. Jednalo se o verzi UML 1.0. V roce 1997 byl jazyk UML přijat jako standard. Přijatá byla rozšířená verze s označením 1.1. Jazyk UML se stal standardem v softwarovém průmyslu a dále se vyvíjí. V roce 1999 byl zaveden standard OML 1.3 a v roce 2001 byla uvedena verze UML 1.4. Tato ročníková práce se věnuje specifikaci UML 2.0.

3.1.2 K čemu slouží UML jazyk?

Jazyk UML byl navržen hlavně pro následující účely:

- Modelování aplikačních procesů s případy použití
- Modelování tříd a objektů
- Modelování komponent
- Modelování rozdělení a nasazení

3.1.3 Ukázka UML modelů

Diagram tříd

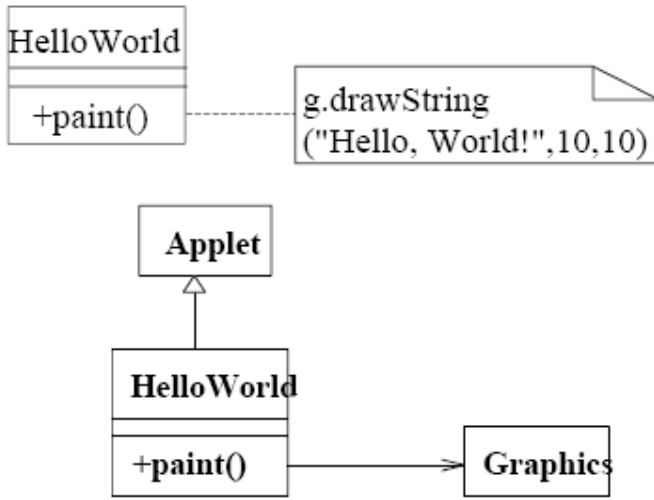


Diagram sekvence

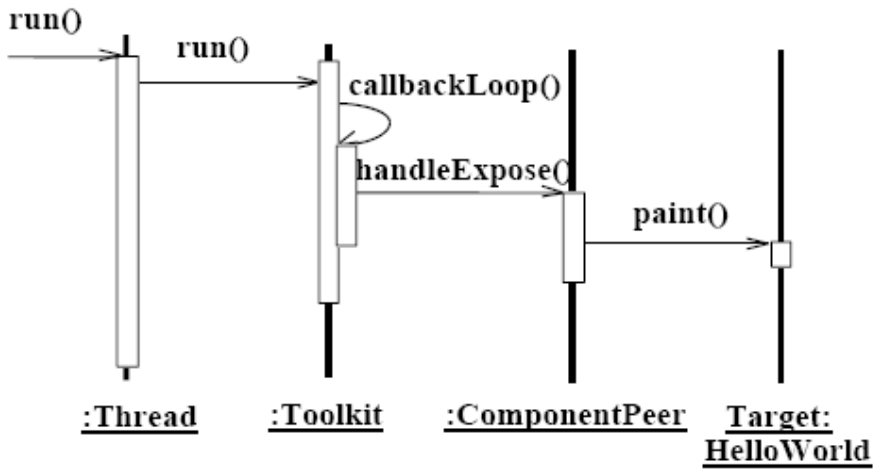
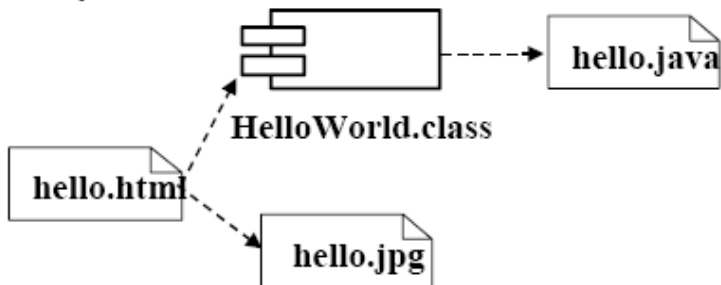


Diagram komponent



Stavový diagram



3.2 Co je to OCL

Object Constraint Language (OCL) je formální jazyk užívaný pro popis výrazů na UML modelech. Tyto výrazy typicky specifikují neměnné podmínky které musí platit pro modelované systémy nebo dotazy nad objekty popsané v modelu. Všimněte si, že když OCL výrazy jsou ohodnocené, nemají vedlejší efekty; to jest jejich vyhodnocení nemůže změnit stav odpovídajícího provádění systému.

OCL výrazy mohou být použity pro specifikování operace / akce které, když vykonáme, změníme stav systému.

Tvůrci UML modelů mohou užívat OCL pro upřesnění aplikačně - specifického omezení v jejich modelech. Také mohou užívat OCL pro zpřesnění dotazů v UML modelu, které jsou úplně nezávislé na programovacím jazyku.

3.2.1 Proč používat OCL

A UML diagram, jako například diagram tříd (class diagram), není typicky dost popisný aby poskytl všechny důležité aspekty specifikace. Je zde, kromě jiného, potřeba popsat další omezení o objektech v modelu. Taková omezení jsou často popsána v přirozeném jazyce. Praxe ukázala že toto bude mít vždy za následek mnohoznačnosti. Za účelem psát jednoznačná omezení, byli vyvinuté tzv. formální jazyky. Nevýhoda tradičních formálních jazyků je, že jsou použitelné pro osoby matematickým vzděláním, ale obtížně použitelné pro průměrného obchodníka nebo systémového modeláře.

OCL byl vyvinutý aby vyplnil tuto mezeru. Je to formální jazyk který je snadný pro čtení a zápis. Byl vyvinut jako obchodní modelový jazyk uvnitř pojišťovacího oddělení IBM a má své kořeny v Syntropy metodě.

Dále si zde povíme o pár konkrétních rysech OCL. Kompletní popis OCL je zahrnut ve vytvořené e-learningové aplikaci.

OCL je ryzí specifikační jazyk, proto OCL výraz je zaručeně bez vedlejšího efektu. Při vyhodnocování OCL výrazu je vracena hodnota. Ta v modelu nemůže cokoliv změnit. Toto znamená, že se stav systému nikdy nezmění kvůli vyhodnocení OCL výrazu, třebaže OCL výraz může být užívaný pro upřesnění změny stavu (např. v následné podmínce).

OCL není programovací jazyk, proto není možné psát programovou logiku nebo řízení toku v OCL. Nemůžeme vyvolávat procesy nebo aktivovat nedotazující se operace uvnitř OCL. Protože OCL je v první řadě modelový jazyk OCL výrazy nejsou samozřejmě přímo proveditelné.

OCL je typový jazyk, takže každý OCL výraz má datový typ. Aby byl správně vytvořen, musí se OCL výraz přizpůsobit typu shodného s pravidly jazyka. Například, nemůžeme srovnávat celé číslo (Integer) s řetězcem (String). Každý klasifikátor definovaný uvnitř UML modelu reprezentuje různý OCL datový typ. Navíc, OCL zahrnuje soubor doplňkových předdeklarovaných datových typů.

Jako specifikační jazyk, všechny implementační problémy jsou mimo rozsah a nemohou být vyjádřeny v OCL. Vyhodnocení OCL výrazu je okamžité. Toto znamená, že stavy objektů v modelu se nemohou změnit během vyhodnocení.

Dále se nemá cenu rozepisovat, protože to není cílem této dokumentace.

3.2.2 Případy použití OCL

OCL může být použit pro několik různých účelů:

- Jako dotazovací jazyk
- Pro specifikování invariant na třídách a typů v modelů tříd
- Pro upřesnění datového typu invarianta pro stereotypy
- Pro popisování předchozích a následných podmínek(pre- and post conditions) na operacích a metodách
- Pro popisování Guards
- Pro upřesnění cíle (nastavení) zpráv a akcí
- Pro upřesnění omezeních operací
- Pro zpřesnění odvození pravidel pro atributy každého výrazu v UML modelu

Ukázka OCL příkazu

Pro názornost uvádím jak vypadá OCL příkaz společně s diagramem tříd ke kterému se vztahuje.

context c : Company **inv** enoughEmployees:
c.numberOfEmployees > 50

Tento OCL příkaz udává, že počet zaměstnanců, musí být větší než 50

Následuje diagram tříd, ke kterému se předchozí OCL výraz vztahuje

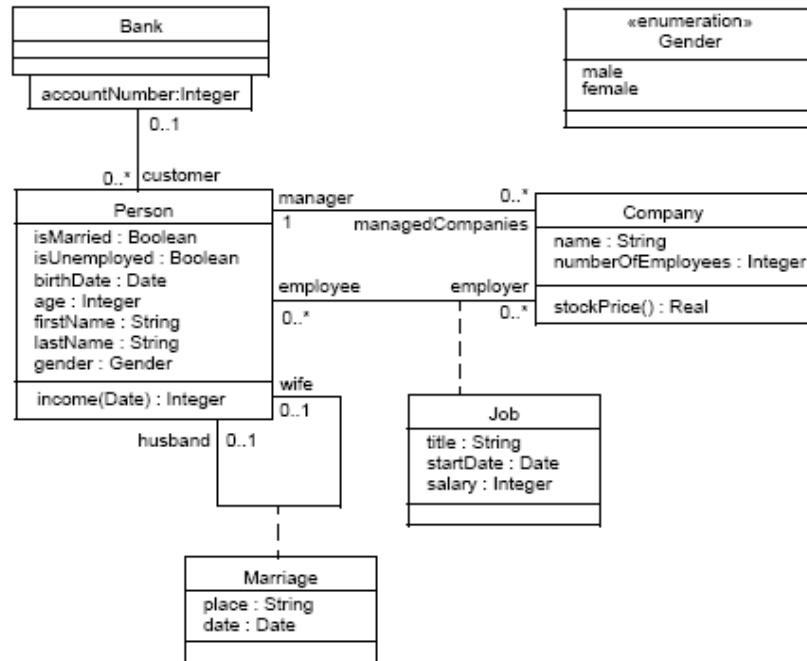


Diagram tříd

3.3 CASE Rational Rose

Rational Rose je objektově orientovaný Unified Modeling Language (UML) softwarový návrhový nástroj určený k vizuálnímu modelování a budování komponent enterprise-level software aplikace. Stejným způsobem jako divadelní ředitel nastíní hru, softwarový návrhář užívá Rational Rose aby vizuálně vytvořil (model) konstrukce pro aplikaci zhruba koncipující třídy s actory (hůlková čísla), užívající case elementy (ovály), objekty (obdélníky) a zprávy/vztahy (šipky) v časovém diagramu používáním táhni a pusť symbolů. Rational Rose dokumentuje diagram tak jak je zkonstruovaný a potom vyrábí kód v podle volby návrháře ať už C++, Visual Basic, Javy, Oracle8, CORBA nebo Data Definition Language.

3.3.1 Vlastnosti

Rational Rose Enterprise – nástroj pro vizuální modelování

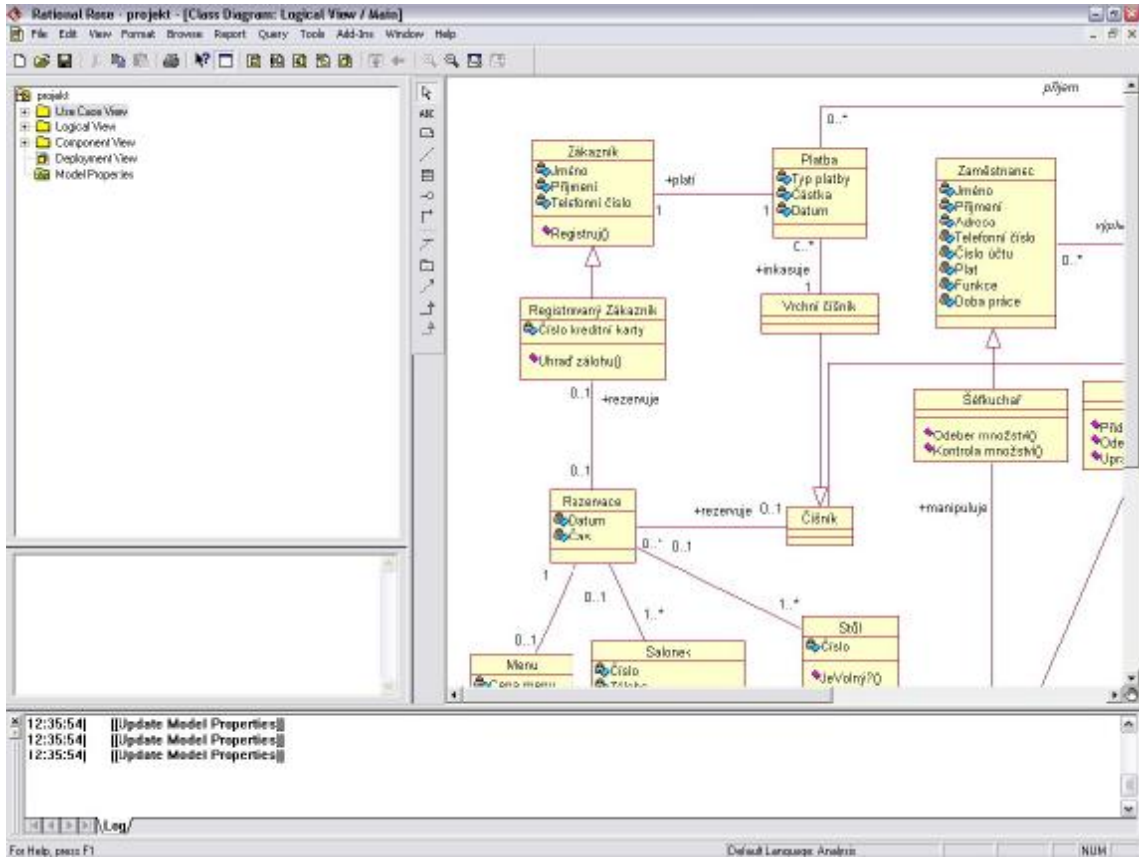
Mezi jeho výhody patří:

- Bezkonkurenční podpora UML přímo od jeho tvůrců, společnosti Rational Software a jejích odporníků, Jamese Jacobsona, Jima Rumbaugh a Grady Booche
- Podpora business proces modelování, modelování architektury systému a datového modelování
- Podpora relačních databází (Oracle, MS SQL Server, DB2, Sybase)
- Objektově relační mapování pro sledování přechodu mezi objektovým modelem a návrhem databáze a naopak
- Společné prostředí a společná notace pro všechny modelovací aktivity a všechny členy týmu, včetně business analytiků, architektů systému a návrhářů databází
- Round-trip engineering umožňuje snadno dokumentovat stávající systémy a při vývoji rychle předcházet mezi návrhem systému a jeho fyzickou implementací v kódu a naopak
- Podpora běžně používaných vývojových prostředí (C++, Visual C++, Visual Java, Centura, PowerBuilder, Delphi, Smalltalk atd.)
- Round-trip engineering umožňuje databází snadno dokumentovat existující databáze, migrovat schémata mezi různými databázemi a průběžně synchronizovat model s implementací databáze a naopak
- Podpora modelování internetových aplikací

3.3.2 Ukázka prostředí CASE Rational Rose a diagramů vněm modelovaných

Nyní si ukážeme jak vypadá prostředí programu CASE Rational Rose a ukážeme si pár modelů vytvořených v tomto prostředí.

Následující obrázek vyobrazuje prostředí programu CASE Rational Rose.



Další obrázky nám ukazují jak budou vypadat jednotlivé typy diagramů vymodelované v prostředí CASE Rational Rose:

Diagram použití (Use case diagram)

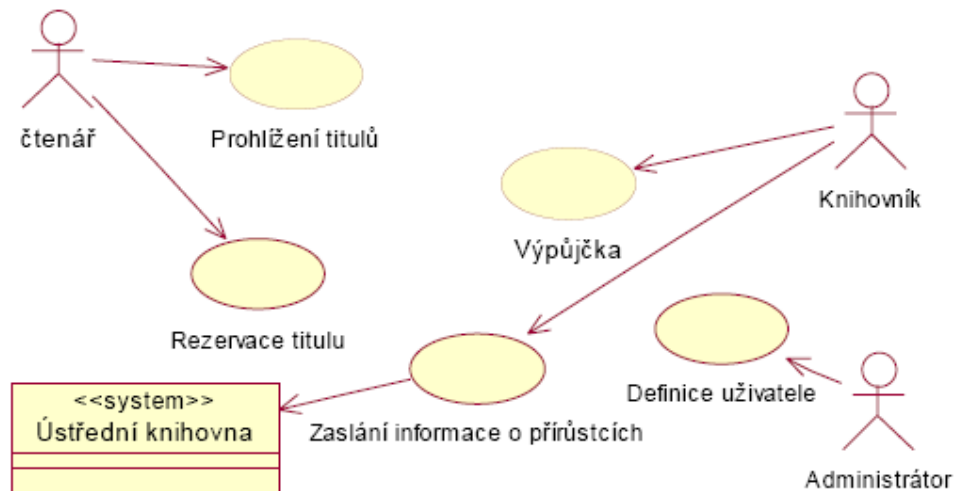
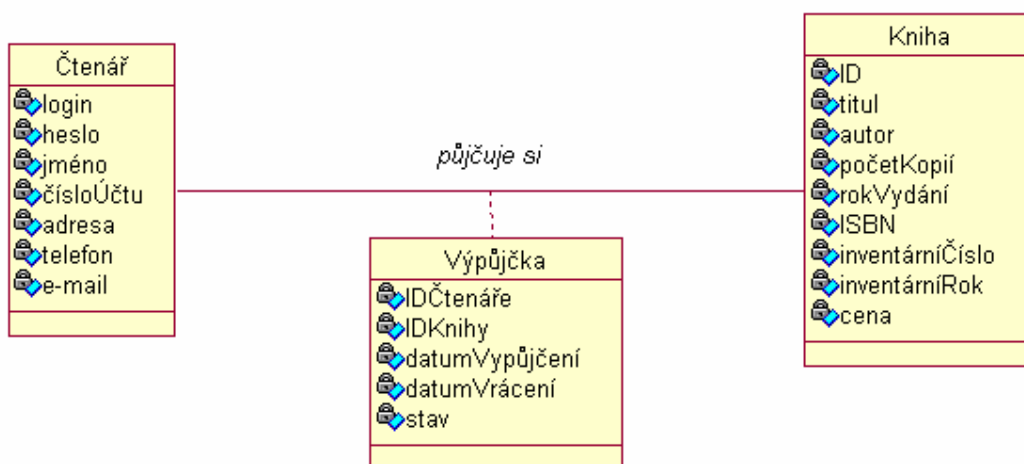
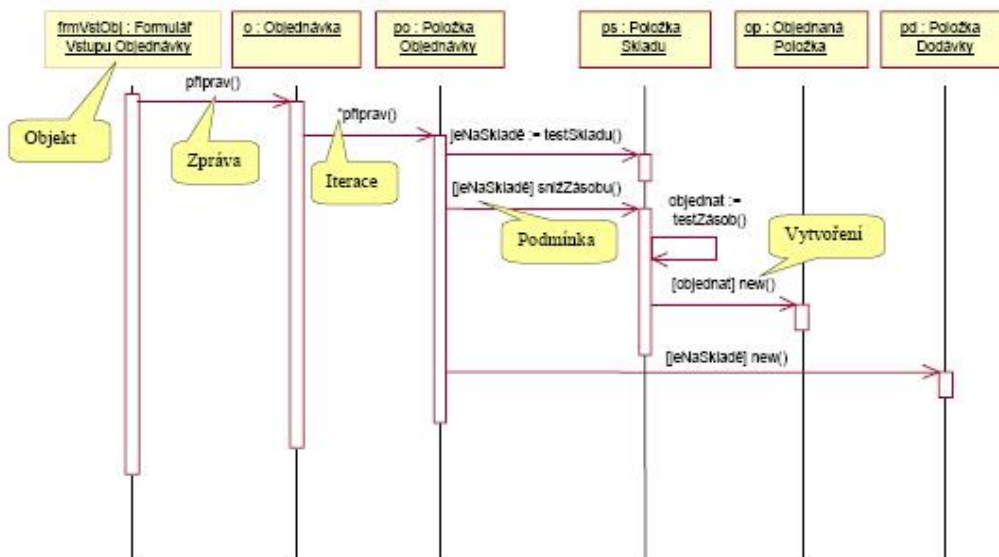


Diagram tříd (Class diagram)



Příklad asociace na diagramu tříd

Diagram sekvence (Sequence diagram)



Pro www stránky popisující specifikaci omezení objektů systému si vystačíme s diagramem tříd. Ostatní diagramy jsou zobrazeny pro úplnost a kompletnost popisu nástroje CASE Rational Rose.

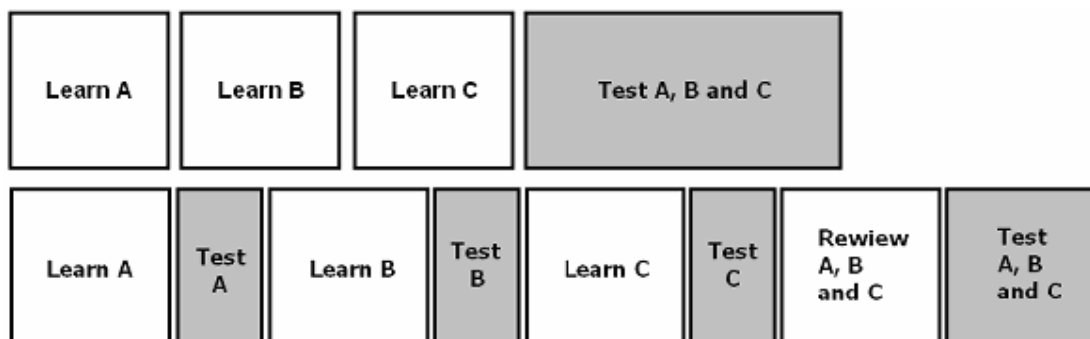
4. Obecná struktura elektronických dokumentů

Poté co jsme si vysvětlili základní pojmy, dostáváme se do části ve které si popíšeme jak by měla vypadat struktura elektronických dokumentů.

V této kapitole budu vycházet z knihy Williama Hortona: Designing Web-Based Training : How to Teach Anyone Anything Anywhere Anytime. V této knize je názorně popsáno jak vytvářet elektronické kurzy, jaká má být jejich struktura a co vše mají popisovat a obsahovat.. Tudiž mnou navržená struktura elektronické aplikace se bude opírat o poznatky získané z této knihy.

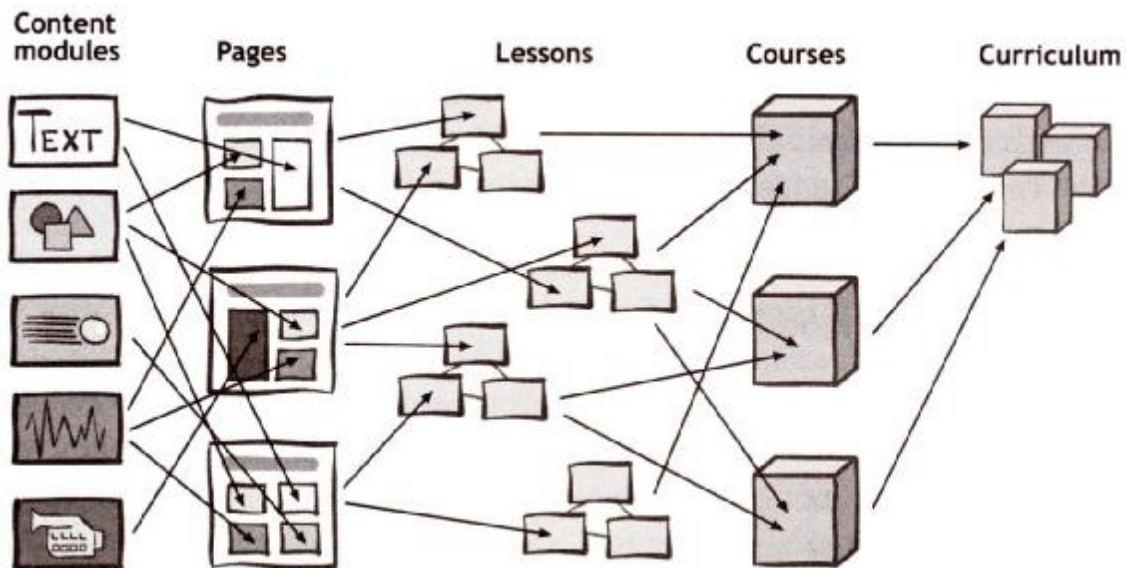
Struktura dokumentu bude odpovídat v tomto projektu struktuře dokumentu WBT (Web-Based Training) kurzu. Pojem WBT jsme si již vysvětlili v kapitole 2.3.

Elektronický kurz se skládá z několika lekcí. V každé lekci se nachází části, které se zabývají seznámením s problematikou a ukázkou příkladů, otestováním právě naučené látky a zobrazení výsledků testů např. v podobě grafu. Obtížnost látky by měla postupně narůstat s další lekcí. Informační a testové části lekce, by se měli střídat v kratších intervalech, aby si uživatel novou problematiku hned odzkoušel viz obr 4.1.



obr. 4.1

Textová informace by měla být doplněna obrazem, videem a zvukem. Toto je výhoda elektronického kurzu oproti jiným výukovým prostředkům. Toto je ukázáno v následném obr. 4.2, který zobrazuje propojení mezi kurzy, lekcemi, stránkami lekcí a jejich obsahem. Toto vše je zahrnuto celkovém studijním plánu. Vzhled a ovládací prvky elektronického kurzu či dokumentu by měli být navrženy co nejpřehledněji.

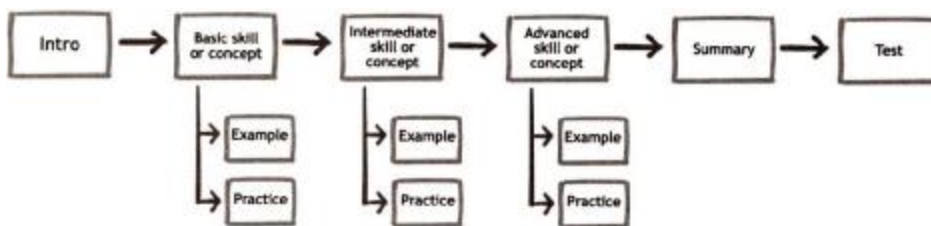


Obr. 4.2.

V tomto projektu se nebudeme zabývat celým studijním plánem ale pouze jedním kurzem, či spíše jednou lekcí.

Poznanky o tvorbě elektronických kurzů či dokumentů můžeme shrnout do několika bodů.

- Lekce se skládají z posloupností logicky spojených webových stránek, které společně vytvářejí bohaté učební zážitky.
- Klasická vyučovací struktura je nejběžnější, nejbezpečnější, a nejméně účinná z běžných vyučovacích struktur.



Ostatní struktury učebních metod vyrobené na objednávku splňují potřeby žáků.

- Určité komponenty jsou běžné v mnoha typech lekcí. Tyto zahrnují následující stránky: přivítání, související zdroje, úvod, shrnutí, událost přehrání nahraného záznamu, zpětná vazba lekcí, rozvětvení, procedury, a uvádění příkladů.
- Při organizování lekcí a kurzů, analyzujte závislosti mezi pojmy a žákovou aktuální úrovní znalostí.
- Usilujte o kompletní samostatné moduly, které mohou se mohou kombinovat pro učení komplexních předmětů.
- Namísto sekvenčních struktur uvažujte o vrstvených strukturách

5. Formát učebního textu

V této kapitole se budu zabývat formátem učebního textu který jsem zvolil pro svůj projekt. Pro svůj projekt jsem si zvolil webové prostředí a to hned z několika důvodů. Hlavním důvodem bylo to aby byla lehce dostupná kdekoliv na škole. K tomu posloužil školní server, kam ji stačilo nahrát. Pro spuštění a její používání stačí, když si sedneme k jakémukoliv počítači s připojením k internetu a zadat správnou webovou adresu do okna prohlížeče. Dalším důvodem byla snaha vytvořit co nejkompatibilnější aplikaci, která byla spustitelná na jakémkoliv počítači kdekoliv na světě. Jedinou nevýhodou tohoto přístupu je nutnost připojení k internetu. Tento problém jsem vyřešil možností dodání aplikace na CD-ROMu a jejím následném spuštění lokálně na jednotlivém počítači bez nutnosti připojovat se na internet. Důležitou roli při výběru bylo také to že takováto aplikace je velmi přehledná a také v pozdějších fázích snadno upravitelná.

Nyní si povíme něco o použitých technologiích a postupech které jsem použil při vytváření ročníkového projektu. Projekt jsem vytvořil jako www stránky s pomocí php skriptů. Tento způsob se mi zdál pro tento projekt jako nejvýhodnější.

K vlastnostem WWW stránek patří:

- používané technologie je pouze HTML a v omezené míře PHP bez využití MySQL
- veškerý obsah je statický
- nehodí se pro prezentaci většího množství strukturovaných dat které je potřeba dynamicky třídit, např. rozsáhlejší ceníky.

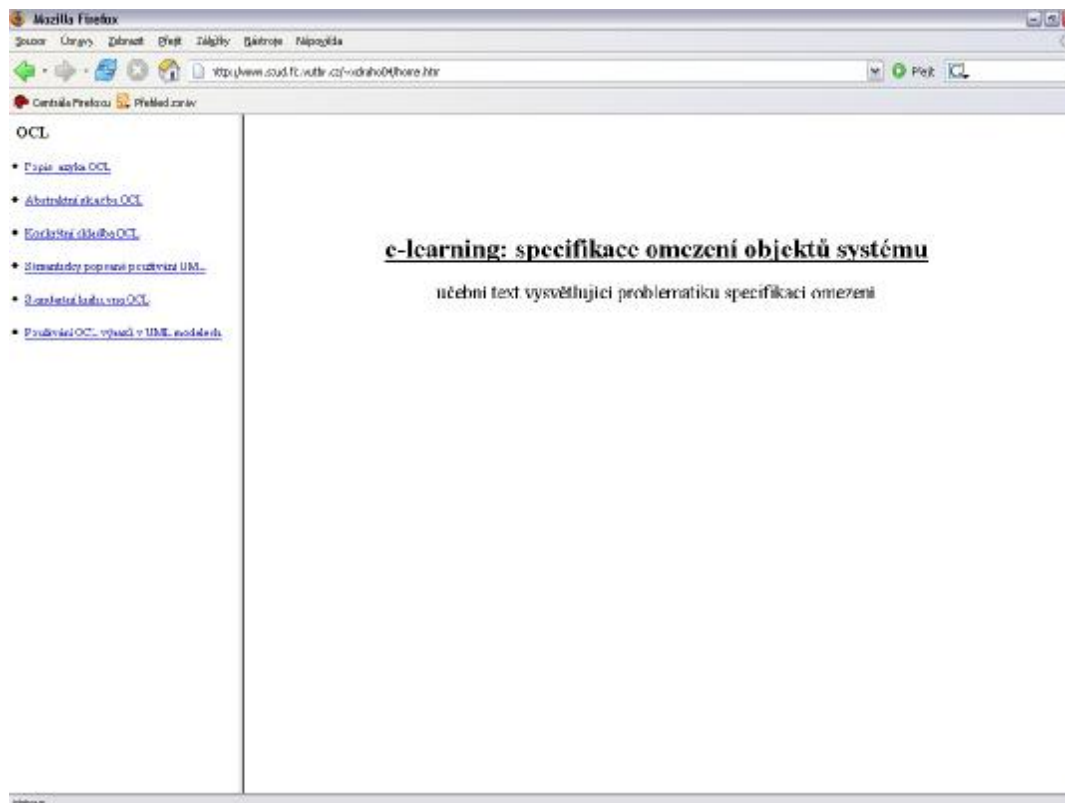
Což za účelem vytvoření e-learningové aplikace je plně dostačující.

5.1 Vzhled a popis aplikace

Jak již jsem zmínil v předchozí kapitole, k vytvoření e-learningové aplikace jsem si zvolil webové prostředí a www stránky. Nyní si ukážeme jak tyto stránky vypadají, jaká je jejich struktura a trochu si popíšeme jak s nimi pracovat.

Stránky jsou vytvořeny v jazyce html a pomocí PHP skriptů. Po grafické stránce jsou poměrně jednoduché zato ale velmi přehledné a srozumitelné.

Na obrázku číslo 5.1 vidíme úvodní stránku. Ta obsahuje 2 okna z nichž jedno to vlevo je tzv. menu okno, ve kterém můžeme vidět obsah celé aplikace.

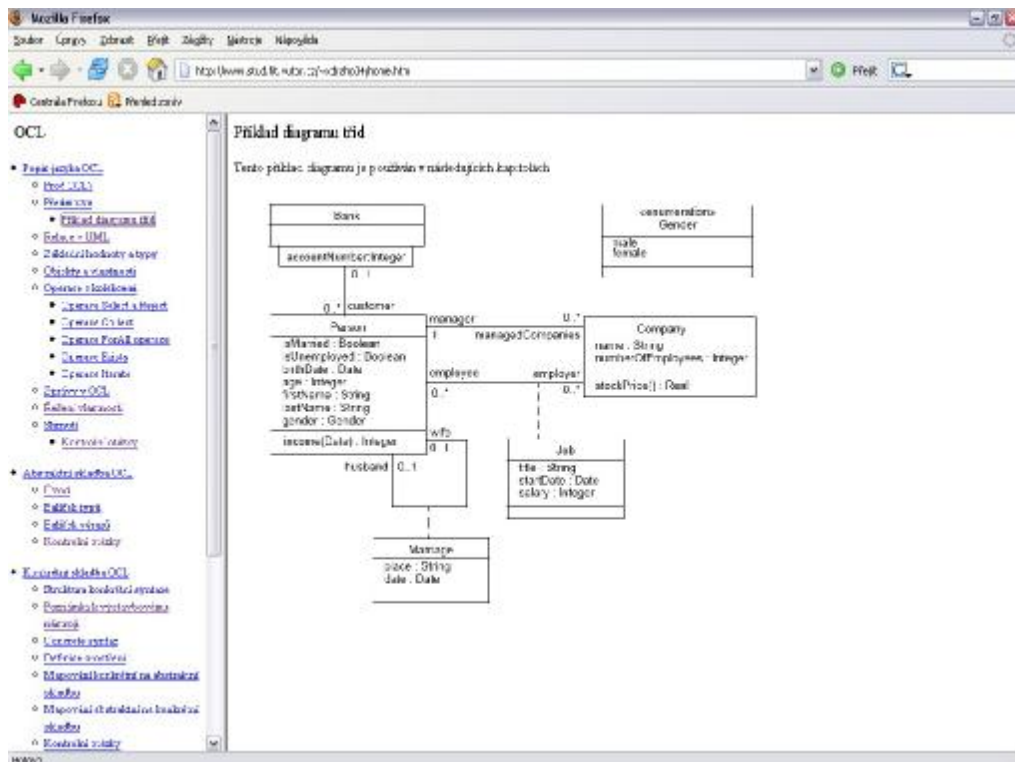


Obrázek č. 5.1

Obsah je rozdělen do 6 hlavních bodů, kde se zabývá určitou částí problematiky OCL. Každý bod je vlastně jedna kapitola, která obsahuje několik podkapitol. Podkapitoly se nám zobrazí pokud klikneme myší na kapitolu, která nás zajímá. Opětovným kliknutím myší na hlavní kapitolu se nám zobrazené podkapitoly opět schovají jak je zobrazeno na obrázcích 5.2 s 5.3. Tímto způsobem můžeme rozbalovat i podkapitoly.

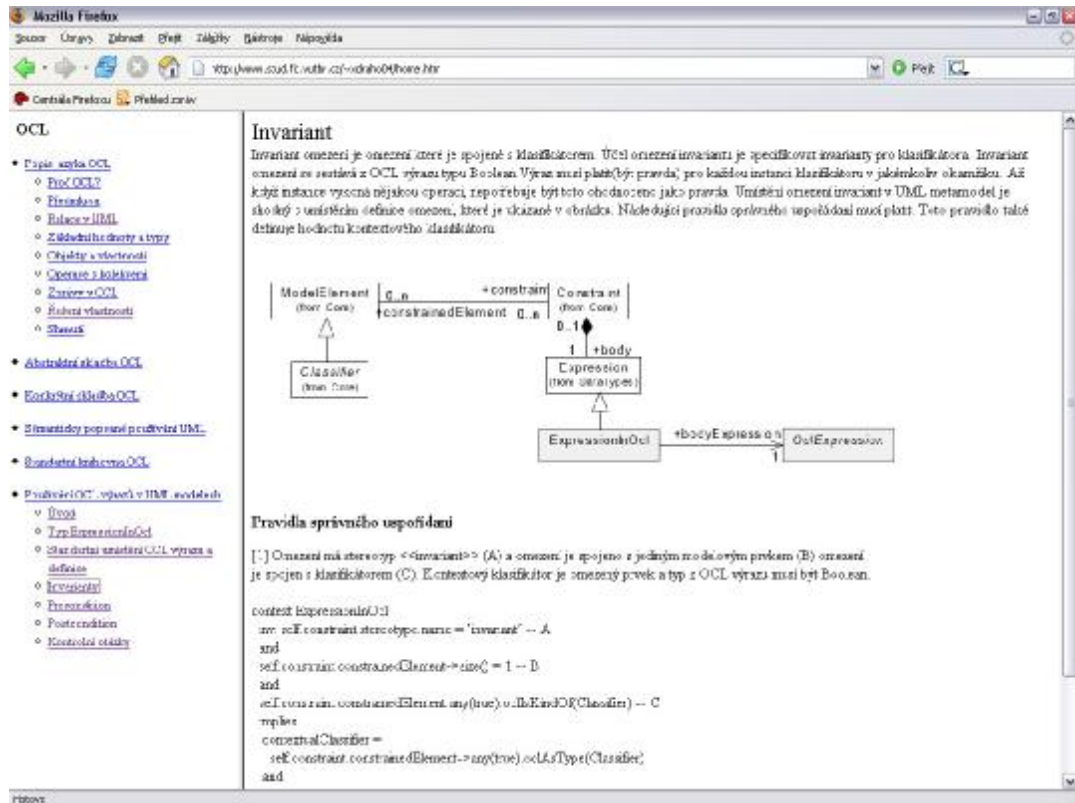


Obrázek č. 5.2



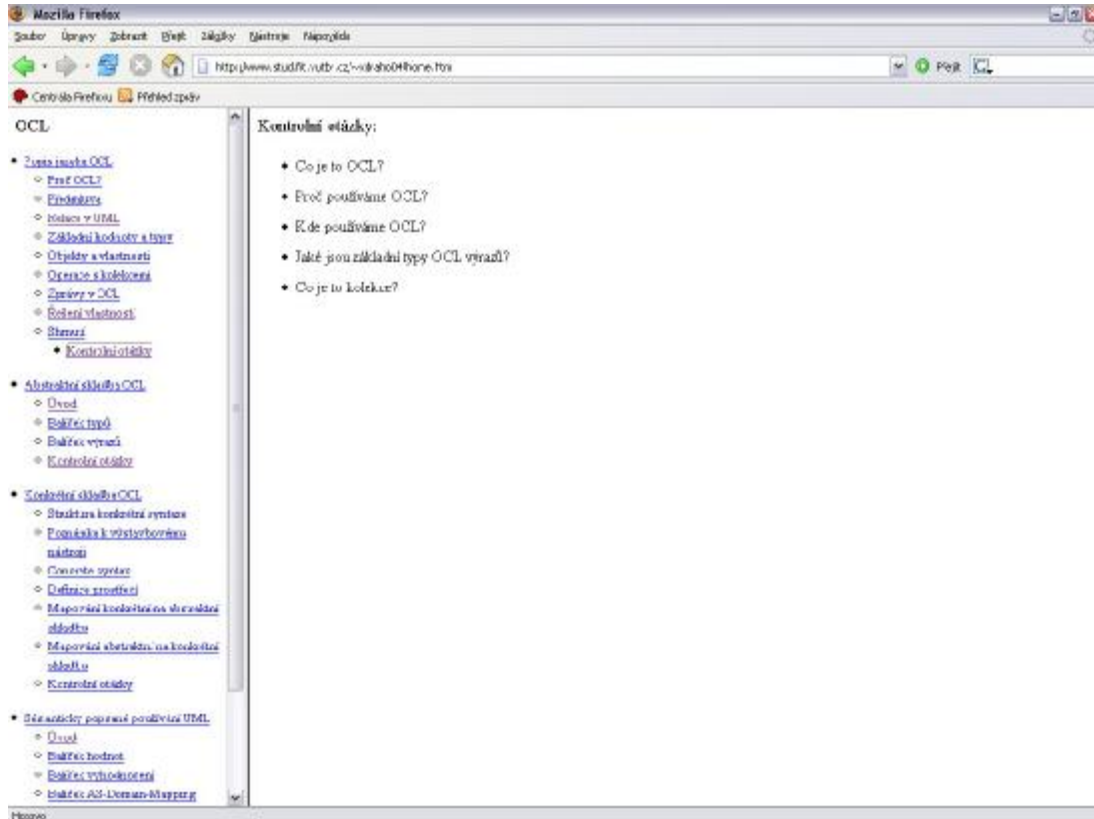
Obrázek č. 5.3

V pravém, větším okně, se nám při kliknutí na jednotlivé kapitoly zobrazí jejich obsah. Ten je uveden velkým nadpisem a dále textem vysvětlujícím danou problematiku. Text je doplněn obrázky a příklady OCL výrazů, což je patrné z obrázku číslo 5.4.



Obrázek č. 5.4

Na konci každé kapitoly v menu je podkapitola otázky, ve které se nachází několik kontrolních otázek sledujících pokrok v pochopení prezentované látky. Viz obrázek číslo 5.5.



Obrázek č. 5.5

Kompatibilita: Tato aplikace byla testována v prostředí Microsoft Internet Explorer a Mozilla Firefox.

6. Příklady vytvořené v prostředí CASE Rational Rose

Co se týká příkladu tak jsou uvedeny v učebním textu a i v této dokumentaci. Prostor programu CASE Rational Rose je rozepsáno v kapitole 3.3 CASE Rational Rose. Jsou zde uvedeny informace nejen o programu samotném, ale také jsou zde uvedeny příklady modelů vymodelovaných v tomto prostředí.

7. Zhodnocení použitelnosti textů ve výuce a závěr

První dva body zadání mého ročníkového projektu, tedy seznámení se s problematikami e-learningu, OCL (UML 2.0) a CASE Rational Rose pokrývají kapitoly 1 a 2. Po prostudování této problematiky jsem začal vytvářet aplikaci sloužící k výuce. Aplikace se zabývá výše uvedeným problémem a je určena pro výuku na vysokých školách. Je vytvořena v html jazyce a pomocí skriptovacího jazyku PHP. Při vytváření projektu jsem si oživil a vylepšil znalosti pro vytváření webových stránek a také se naučil pracovat s PHP skripty.

Problematika e-learningu je velmi zajímavé téma, protože si myslím, že je to velmi účinná alternativa k běžným vzdělávacím prostředkům. Je vhodná jak pro skupiny lidí, tak i pro samouky. K hlavním výhodám elektronického vzdělávání patří vlastnost prezentovat multimediální obsah. Jako nevýhodu bych uvedl, že v mnoha případech při tomto typu vzdělávání dochází k izolaci žáka a učitele, což z pohledu pedagogiky není nejlepší řešení.

Toto téma mě zaujalo natolik, že bych v něm rád pokračoval a to v rámci tvorby diplomové práce. Hlavním úkolem by bylo zdokonalení a rozšíření projektu, například implementování testové části a také hlubší prostudování problematiky elektronického vzdělávání.

Přehled použité literatury:

- [1] Horton, W.: Designing Web-Based Training : How to Teach Anyone Anything Anywhere Anytime, Wiley; 1 edition, ISBN: 047135614X
- [2] Stiller, E., LeBlanc, C.: Project-Based Software Engineering. An Object-Oriented Approach, Addison Wesley, 2002
- [3] Horton, W.: E-learning Tools and Technologies. John Wiley & Sons, Inc.; 1st edition (January 10, 2003), ISBN: 0471444588.
- [4] Shmuller, J.: Myslíme v jazyku UML, knihovna programátora, Grada Publishing, spol. s r.o.,2001, ISBN: 8024700298.
- [5] Arlow J., Neustat I.:UML a unifikovaný proces vývoje aplikací, Computer Press, 2003, ISBN: 807226947X.
- [6] Clark, R. C., Mayer R. E., E-Learning and science of instruction: proven guidelines for customers and designers of multimedia learning, John Wiley & Sons, Inc., 2003, ISBN: 0787960519
- [7] Zendulka, J.: Elektronické podklady k přednáškám předmětu PPS, FIT VUT Brno
- [8] History of e-Learning [Online]
URL.:<<http://knowledgegenet.com/corporateinformation/ourhistory/history.jsp>>
- [9] UML 2.0 OCL Specification [Online]
URL.:<<http://www.omg.org/docs/ptc/03-10-14.pdf>>