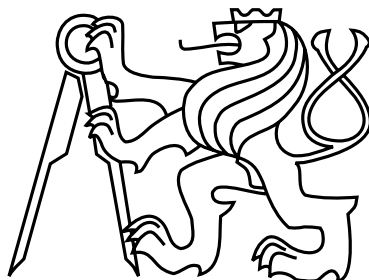


Na tomto místě bude oficiální zadání vaší práce

- Toto zadání je podepsané děkanem a vedoucím katedry,
- musíte si ho vyzvednout na studijním oddělení Katedry počítačů na Karlově náměstí,
- v jedné odevzdané práci bude originál tohoto zadání (originál zůstává po obhajobě na katedře),
- ve druhé bude na stejném místě neověřená kopie tohoto dokumentu (tato se vám vrátí po obhajobě).

České vysoké učení technické v Praze
Fakulta elektrotechnická
Katedra počítačů



Semestrální práce

Systém pro evidenci hospitací v Ruby on Rails

Tomáš Turek

Vedoucí práce: Ing. Martin Komárek

Studijní program: Softwarové technologie a management, Bakalářský

Obor: Softwarové inženýrství

29. ledna 2012

Poděkování

Zde můžete napsat své poděkování, pokud chcete a máte komu děkovat.

Prohlášení

Prohlašuji, že jsem práci vypracoval samostatně a použil jsem pouze podklady uvedené v příloženém seznamu.

Nemám závažný důvod proti užití tohoto školního díla ve smyslu §60 Zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

V České Lípě dne 28. 1. 2012

.....

Abstract

Translation of Czech abstract into English.

Abstrakt

Tato práce se zabývá rozšířením prototypu aplikace pro evidenci hospitací, tak aby jej bylo možné nasadit do reálného provozu na ČVUT FEL. Systém má za účel zjednoduší současnou administrativu kontroly kvality okolo hospitací.

Obsah

1	Úvod	1
2	Popis problému, specifikace cíle	3
2.1	Cíle práce	3
2.1.1	Nultá iterace	3
2.1.2	První iterace	3
2.2	Rešerše	3
3	Analýza a návrh řešení	5
3.1	Požadavky	5
3.1.1	Obecné požadavky	5
3.1.2	Funkční požadavky	5
3.2	Uživatelské role	6
3.2.1	Nepřihlášený uživatel	6
3.2.2	Přihlášený uživatel	6
3.2.3	Hospitovaný	6
3.2.4	Hospitující	6
3.2.5	Administrátor hospitací	6
3.2.6	Administrátor	7
3.3	Doménový model	7
3.3.1	Domény v KOSapi	7
3.4	Životní cyklus hospitace	7
3.5	Technologie	7
4	Realizace	9
5	Testování	11
6	Závěr	13
A	Testování zaplnění stránky a odsazení odstavců	15
B	Pokyny a návody k formátování textu práce	19
B.1	Vkládání obrázků	19
B.2	Kreslení obrázků	20
B.3	Tabulky	20
B.4	Odkazy v textu	21

B.4.1	Odkazy na literaturu	21
B.4.2	Odkazy na obrázky, tabulky a kapitoly	23
B.5	Rovnice, centrovaná, číslovaná matematika	23
B.6	Kódy programu	24
B.7	Další poznámky	24
B.7.1	České uvozovky	24
C	Seznam použitých zkratk	25
D	UML diagramy	27
E	Instalační a uživatelská příručka	29
F	Obsah přiloženého CD	31

Seznam obrázků

B.1	Popiska obrázku	20
F.1	Seznam přiloženého CD — příklad	31

Seznam tabulek

B.1 Ukázka tabulky	20
------------------------------	----

Kapitola 1

Úvod

Na FEL ČVUT existuje systém kontroly kvality výuky, ten slouží k ověření kvality výuky s cílem zkvalitnit výuku pro různé studijní programy. Jedním ze zdrojů informací jsou kontrolní návštěvy ve výuce (hospitace). Účelem těchto návštěv přednášek, nebo cvičení je získání komplexního obrazu o kvalitě výuky jednotlivých předmětů a zároveň slouží jako zpětná vazba pro pedagogy.

Cílem mé práce je navrhnout a vytvořit informační systém pro správu hospitací, který zjednoduší aktuální stav administrativy hospitací, která používá hlavně emailovou komunikaci a systém wiki stránek.

Kapitola 2

Popis problému, specifikace cíle

2.1 Cíle práce

Hlavním cílem semestrální práce je rozšířit prototyp aplikace pro evidenci hospitací na platformě Ruby on Rails, tak aby bylo možné nasadit částečně do reálného provozu v letním semestru 2011/2012 pro program STM. Proto je vývoj veden pomocí iterativního způsobu. Součástí semestrální práce je nultá a první iterace.

Momentální stav plánování hospitací na FEL ČVUT je založena na komunikaci pomocí systému wiki stránek. Tento způsob evidence je příliš náročný na administrativu a v současné době nevyhovuje. Proto je potřeba vytvořit informační systém, který zautomatizuje vnitřní procesy plánování hospitací.

2.1.1 Nultá iterace

Součástí nulté iterace bylo seznámení se s procesy plánování hospitací a vytvoření prototypu aplikace, který dokáže komunikovat s webovou službou KOSapi. Pro komunikaci slouží modul převzatý ze školního projektu VyVy. Analytickou část semestrálního projektu vycházím z rozpracované bakalářské práce Daniela Kreželoka Návrh a implementace systému pro správu hospitací.

2.1.2 První iterace

Součástí první iterace je návrh a implementace procesu naplánování hospitací.

2.2 Rešerše

Pro zjištění fungování hospitací

Kapitola 3

Analýza a návrh řešení

Cílem této kapitoly je vyhodnotit funkční a nefunkční požadavky na aplikaci. Definuji zde doménový model a případy užití s business procesy pro vytvoření hospitace.

3.1 Požadavky

Požadavky na systém se dělí na dvě sekce funkčních a nefunkčních požadavky. Pro definování těchto požadavků vycházejí ze zadání práce. Tyto požadavky definují návrh a pozdější implementaci systému a definují meze systému.

3.1.1 Obecné požadavky

Obecné požadavky jsou požadavky, které se netýkají funkčnosti ale celkového návrhu a použitých technologií.

1. Systém bude postaven na webovém frameworku Ruby on Rails.
2. Systém bude používat databázi MySQL.
3. Systém bude webovou aplikací.
4. Serverová část systému poběží na aplikačním serveru Apache2.
5. Systém bude používat webovou službou KOSapi.
6. Systém bude pro autentizaci používat FELid.

3.1.2 Funkční požadavky

Tato sekce se zabývá požadavky na funkčnost systému. Systém umožní:

1. Systém umožní spravovat uživatele.
2. Systém umožní průběžné plánování hospitací.

3. Systém umožní přiřazení hospitujícího k hospitaci.
4. Systém umožní vyhledávat předměty z KOSapi.
5. Systém umožní vyhledávat osoby z KOSapi.

3.2 Uživatelské role

Uživatelské role v aplikaci jsou odvozeny z modelu aktérů systému u každého uživatele je diagram případu užití. Rozložení rolí v systému je takové, že uživatel systému se může nacházet v několika rolích. V systému jsou následující role:

3.2.1 Nepřihlášený uživatel

Nepřihlášený uživatel Nepřihlášený uživatel je host naší aplikace, má nejmenší pravomoc v systému. V tomto stavu je každý uživatel, který se doposud nepřihlásil do. Tento uživatel má právo pouze si prohlížet veřejné naplánované hospitace.

3.2.2 Přihlášený uživatel

Přihlášený uživatel je uživatel systému, který se již přihlásil do systému. Tento uživatel se umí pouze na rozdíl od nepřihlášeného uživatele pouze odhlásit do systému.

3.2.3 Hospitovaný

Hospitovaný je role pro přihlášeného uživatele v systému. Tento uživatel je vyučující, který vyučuje předmět na němž je naplánovaná hospitace. Tato role přidává možnost přístupu k naplánované hospitaci. U těchto hospitací má právo na zobrazení informací, zobrazení hodnocení a sepsat stanovisko k názorům hospitujícího.

3.2.4 Hospitující

Hospitující je role pro přihlášeného uživatele v systému. Je uživatel přidělený k vykonání hospitace. U přidělených hospitací má uživatel právo na zobrazení informací a možnost vytvořit hodnocení z vykonané hospitace.

3.2.5 Administrátor hospitací

Administrátor hospitací je přihlášený uživatel, který má právo spravovat hospitace a k nim přidělovat hospitující. Dále má právo přidělovat uživatelům role administrátor hospitací a hospitující.

3.2.6 Administrátor

Administrátor je uživatel který má nejvyšší pravomoc v systému. Má pravomoc spravovat všechny části aplikace.

3.3 Doménový model

Doménový model reprezentuje klíčové domény systému a jejich vztahy mezi sebou. V diagramu je vidět propojení domén s doménami v KOSapi.

3.3.1 Domény v KOSapi

- Osoba uživatelé v kosu
- Semestr semestry
- Předmět pře
- Instance předmětu in
- Paralelka par
- Místnost místnost

3.3.2 Domény aplikace

- Uživatel
- Hospitace
- Poznámka
- Příloha
- Formulář
- Hodnota
- Typ formuláře
- Položka

3.4 Životní cyklus hospitace

3.5 Technologie

Kapitola 4

Realizace

Popis implementace/realizace se zaměřením na nestandardní části řešení.

Kapitola 5

Testování

- Způsob, průběh a výsledky testování.
- Srovnání s existujícími řešeními, pokud jsou známy.

Kapitola 6

Závěr

- Zhodnocení splnění cílů DP/BP a vlastního přínosu práce (při formulaci je třeba vzít v potaz zadání práce).
- Diskuse dalšího možného pokračování práce.

Příloha A

Testování zaplnění stránky a odsazení odstavců

Tato příloha nebude součástí vaší práce. Slouží pouze jako příklad formátování textu.

Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili?

Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili?

Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili?

Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili? Určitě existuje nějaká pěkná latinská věta, která se k tomuhle testování používá, ale co mají dělat ti, kteří se nikdy latinsky neučili?

Příloha B

Pokyny a návody k formátování textu práce

Tato příloha samozřejmě nebude součástí vaší práce. Slouží pouze jako příklad formátování textu.

Používat se dají všechny příkazy systému L^AT_EX. Existuje velké množství volně přístupné dokumentace, tutoriálů, příruček a dalších materiálů v elektronické podobě. Výchozím bodem, kromě Googlu, může být stránka CSTUG (Czech Tech Users Group) [?]. Tam najdete odkazy na další materiály. Většinou dostačující a přehledně organizovanou elektronikou dokumentaci najdete například na [?] nebo [?].

Existují i různé nadstavby nad systémy T_EX a L^AT_EX, které výrazně usnadní psaní textu zejména začátečníkům. Velmi rozšířený v Linuxovém prostředí je systém Kile.

B.1 Vkládání obrázků

Obrázky se umísťují do plovoucího prostředí `figure`. Každý obrázek by měl obsahovat **název** (`\caption`) a **návěští** (`\label`). Použití příkazu pro vložení obrázku `\includegraphics` je podmíněno aktivací (načtením) balíku `graphicx` příkazem `\usepackage{graphicx}`.

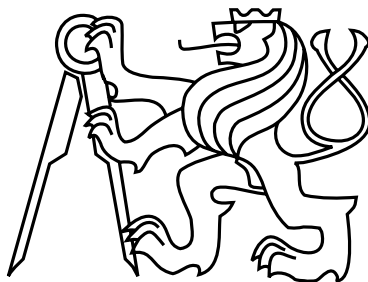
Budete-li zdrojový text zpracovávat pomocí programu `pdflatex`, očekávají se obrázky s příponou `*.pdf`¹, použijete-li k formátování `latex`, očekávají se obrázky s příponou `*.eps`.²

Příklad vložení obrázku:

```
\begin{figure}[h]
\begin{center}
\includegraphics[width=5cm]{figures/LogoCVUT}
\caption{Popiska obrazku}
\label{fig:logo}
```

¹pdflatex umí také formáty PNG a JPG.

²Vzájemnou konverzi mezi snad všemi typy obrázku včetně změn velikostí a dalších vymožeností vám může zajistit balík ImageMagic (<http://www.imagemagick.org/script/index.php>). Je dostupný pod Linuxem, Mac OS i MS Windows. Důležité jsou zejména příkazy `convert` a `identify`.



Obrázek B.1: Popiska obrázku

DTD	construction	elimination
	in1 A B a:sum A B in1 A B b:sum A B	case([_:A] a)([_:B] a)ab:A case([_:A] b)([_:B] b)ba:B
+	do_reg:A -> reg A	undo_reg:reg A -> A
*, ?	the same like and + with empty_el:empty	the same like and + with empty_el:empty
R(a,b)	make_R:A->B->R	a: R -> A b: R -> B

Tabulka B.1: Ukázka tabulky

```
\end{center}
\end{figure}
```

B.2 Kreslení obrázků

Zřejmě každý z vás má nějaký oblíbený nástroj pro tvorbu obrázků. Jde jen o to, abyste dokázali obrázek uložit v požadovaném formátu nebo jej do něj konvertovat (viz předchozí kapitola). Je zřejmě vhodné kreslit obrázky vektorově. Celkem oblíbený, na ovládání celkem jednoduchý a přitom dostatečně mocný je například program Inkscape.

Zde stojí za to upozornit na kreslicí programe Ipe [?], který dokáže do obrázku vkládat komentáře přímo v latexovském formátu (vzroce, stejné fonty atd.). Podobné věci umí na Linuxové platformě nástroj Xfig.

Za pozornost ještě stojí schopnost editoru Ipe importovat obrázek (jpg nebo bitmap) a krelit do něj latexovské popisky a komentáře. Výsledek pak umí exportovat přímo do pdf.

B.3 Tabulky

Existuje více způsobů, jak sázet tabulky. Například je možno použít prostředí `table`, které je velmi podobné prostředí `figure`.

Zdrojový text tabulky B.1 vypadá takto:

```

\begin{table}
\begin{center}
\begin{tabular}{|c|l|l|}
\hline
\textbf{DTD} & \textbf{construction} & \textbf{elimination} \\
\hline
 $\mid$  &  $\verb+in1|A|B a:sum A B+ & \verb+case([_:A]a)([_:B]a)ab:A+ \\
& \verb+in1|A|B b:sum A B+ & \verb+case([_:A]b)([_:B]b)ba:B+ \\
\hline
 $\$$  &  $\verb+do\_reg:A -> reg A+ & \verb+undo\_reg:reg A -> A+ \\
\hline
 $\$$ ,  $\$$  & the same like  $\mid$  and  $\$$  & the same like  $\mid$  and  $\$$  \\
& with  $\verb+empty\_el:empty+ & with \verb+empty\_el:empty+ \\
\hline
 $R(a,b)$  &  $\verb+make\_R:A->B->R+ & \verb+a: R -> A+ \\
& & \verb+b: R -> B+ \\
\hline
\end{tabular}
\end{center}
\caption{Ukázka tabulky}
\label{tab:tab1}
\end{table}
\begin{table}$$$$ 
```

B.4 Odkazy v textu

B.4.1 Odkazy na literaturu

Jsou realizovány příkazem `\cite{odkaz}`.

Seznam literatury je dobré zapsat do samostatného souboru a ten pak zpracovat programem bibtex (viz soubor `reference.bib`). Zdrojový soubor pro bibtex vypadá například takto:

```

@Article{Chen01,
  author = "Yong-Sheng Chen and Yi-Ping Hung and Chiou-Shann Fuh",
  title = "Fast Block Matching Algorithm Based on
           the Winner-Update Strategy",
  journal = "IEEE Transactions On Image Processing",
  pages = "1212--1222",
  volume = 10,
  number = 8,
  year = 2001,
}

@Misc{latexdocweb,

```

```

author = "",
title  = "{\LaTeX} --- online manuál",
note   = "\verb|http://www.cstug.cz/latex/lm/frames.html|",
year   = "",
}
...

```

Pozor: Sazba názvů odkazů je dána BibTeX stylem (`\bibliographystyle{abbrv}`). BibTeX tedy obvykle vysází velké pouze počáteční písmeno z názvu zdroje, ostatní písmena zůstanou malá bez ohledu na to, jak je napíšete. Přesněji řečeno, styl může zvolit pro každý typ publikace jiné konverze. Pro časopisecké články třeba výše uvedené, jiné pro monografie (u nich často bývá naopak velikost písmen zachována).

Pokud chcete BibTeXu napovědět, která písmena nechat bez konverzí (viz `title = "{\LaTeX} --- online manuál"` v předchozím příkladu), je nutné příslušné písmeno (zde celé makro) uzavřít do složených závorek. Pro přehlednost je proto vhodné celé parametry uzavírat do uvozovek (`author = "..."`), nikoliv do složených závorek.

Odkazy na literaturu ve zdrojovém textu se pak zapisují:

Podívejte se na `\cite{Chen01}`,
další detaily najdete na `\cite{latexdocweb}`

Vazbu mezi soubory `*.tex` a `*.bib` zajistíte příkazem `\bibliography{}` v souboru `*.tex`. V našem případě tedy zdrojový dokument `thesis.tex` obsahuje příkaz `\bibliography{reference}`.

Zpracování zdrojového textu s odkazy se provede postupným voláním programů `pdflatex <soubor>` (případně `latex <soubor>`), `bibtex <soubor>` a opět `pdflatex <soubor>`.³

Níže uvedený příklad je převzat z dříve existujících pokynů studentům, kteří dělají svou diplomovou nebo bakalářskou práci v Grafické skupině.⁴ Zde se praví:

```

...
j) Seznam literatury a dalších použitých pramenů, odkazy na WWW stránky, ...
Pozor na to, že na veškeré uvedené prameny se musíte v textu práce
odkazovat -- [1].
Pramen, na který neodkazujete, vypadá, že jste ho vlastně nepotřebovali
a je uveden jen do počtu. Příklad citace knihy [1], článku v časopise [2],
statí ve sborníku [3] a html odkazu [4]:
[1] J. Žára, B. Beneš, and P. Felkel.

```

Moderní počítačová grafika. Computer Press s.r.o, Brno, 1 edition, 1998.

³První volání `pdflatex` vytvoří soubor s koncovkou `*.aux`, který je vstupem pro program `bibtex`, pak je potřeba znovu zavolat program `pdflatex` (`latex`), který tentokrát zpracuje soubory s příponami `.aux` a `.tex`. Informaci o případných nevyřešených odkazech (cross-reference) vidíte přímo při zpracovávání zdrojového souboru příkazem `pdflatex`. Program `pdflatex` (`latex`) lze volat vícekrát, pokud stále vidíte nevyřešené závislosti.

⁴Několikrát jsem byl upozorněn, že web s těmito pokyny byl zrušen, proto jej zde přímo necituji. Nicméně příklad sám o sobě dokumentuje obecně přijímaný konsensus ohledně citací v bakalářských a diplomových pracích na KP.

(in Czech).

- [2] P. Slavík. Grammars and Rewriting Systems as Models for Graphical User Interfaces. *Cognitive Systems*, 4(4--3):381--399, 1997.
- [3] M. Haendl, Š. Kment, and P. Slavík. Virtual Information Systems. In *WSCG'2000 -- Short communication papers*, pages 22--27, Pilsen, 2000. University of West Bohemia.
- [4] Knihovna grafické skupiny katedry počítačů:
<http://www.cgg.cvut.cz/Bib/library/>

... abychom výše citované odkazy skutečně našli v (automaticky generovaném) seznamu literatury tohoto textu, musíme je nyní alespoň jednou citovat: Kniha [?], článek v časopisu [?], příspěvek na konferenci [?], [www odkaz](#) [?].

Ještě přidáme další ukázkou citací online zdrojů podle české normy. Odkaz na wiki o frameworkích [?] a ORM [?]. Použití viz soubor `reference.bib`. V seznamu literatury by nyní měly být živé odkazy na zdroje. V `reference.bib` je zcela nový typ publikace. Detaily dohledal a dodal Petr Dlouhý v dubnu 2010. Podrobnosti najdete ve zdrojovém souboru tohoto textu v komentáři u příkazu `\thebibliography`.

B.4.2 Odkazy na obrázky, tabulky a kapitoly

- Označení místa v textu, na které chcete později čtenáře práce odkázat, se provede příkazem `\label{navesti}`. Lze použít v prostředích `figure` a `table`, ale též za názvem kapitoly nebo podkapitoly.
- Na návěští se odkážeme příkazem `\ref{navesti}` nebo `\pageref{navesti}`.

B.5 Rovnice, centrováná, číslovaná matematika

Jednoduchý matematický výraz zapsaný přímo do textu se vysází pomocí prostředí `math`, resp. zkrácený zápis pomocí uzavření textu rovnice mezi znaky `$`.

Kód `$ S = \pi * r^2 $` bude vysázen takto: $S = \pi * r^2$.

Pokud chcete nečíslované rovnice, ale umístěné centrovane na samostatné řádky, pak lze použít prostředí `displaymath`, resp. zkrácený zápis pomocí uzavření textu rovnice mezi znaky `$$`. Zdrojový kód: `$$$ S = \pi * r^2 $$$` bude pak vysázen takto:

$$S = \pi * r^2$$

Chcete-li mít rovnice číslované, je třeba použít prostředí `equation`. Kód:

```
\begin{equation}
  S = \pi * r^2
\end{equation}
```

```
\begin{equation}
  V = \pi * r^3
\end{equation}
```

je potom vysázen takto:

$$S = \pi * r^2 \quad (\text{B.1})$$

$$V = \pi * r^3 \quad (\text{B.2})$$

B.6 Kódy programu

Chceme-li vysázet například část zdrojového kódu programu (bez formátování), hodí se prostředí verbatim:

```
(* nickname2 *)
Lego> Refine in1
      (do_reg (nickname1 h));
Refine by in1 (do_reg (nickname1 h))
?4 : pcddata
?5 : pcddata
      (* surname2 *)
Lego> Refine surname1 h;
Refine by surname1 h
?5 : pcddata
      (* email2 *)
Lego> Refine undo_reg (email1 h);
Refine by undo_reg (email1 h)
*** QED ***
```

B.7 Další poznámky

B.7.1 České uvozovky

V souboru `k336_thesis_macros.tex` je příkaz `\uv{}` pro sázení českých uvozovek. „Text uzavřený do českých uvozovek.“

Příloha C

Seznam použitých zkratek

2D Two-Dimensional

ABN Abstract Boolean Networks

ASIC Application-Specific Integrated Circuit

⋮

Příloha D

UML diagramy

Tato příloha není povinná a zřejmě se neobjeví v každé práci. Máte-li ale větší množství podobných diagramů popisujících systém, není nutné všechny umísťovat do hlavního textu, zvláště pokud by to snižovalo jeho čitelnost.

Příloha E

Instalační a uživatelská příručka

Tato příloha velmi žádoucí zejména u softwarových implementačních prací.

Příloha F

Obsah přiloženého CD

Tato příloha je povinná pro každou práci. Každá práce musí totiž obsahovat přiložené CD. Viz dále.

Může vypadat například takto. Váš seznam samozřejmě bude odpovídat typu vaší práce. (viz [?]):



Obrázek F.1: Seznam přiloženého CD — příklad

Na GNU/Linuxu si strukturu přiloženého CD můžete snadno vyrobit příkazem:
`$ tree . >tree.txt`

Ve vzniklém souboru pak stačí pouze doplnit komentáře.

Z **README.TXT** (případně **index.html** apod.) musí být rovněž zřejmé, jak programy instalovat, spouštět a jaké požadavky mají tyto programy na hardware.

Adresář **text** musí obsahovat soubor s vlastním textem práce v PDF nebo PS formátu, který bude později použit pro prezentaci diplomové práce na WWW.