

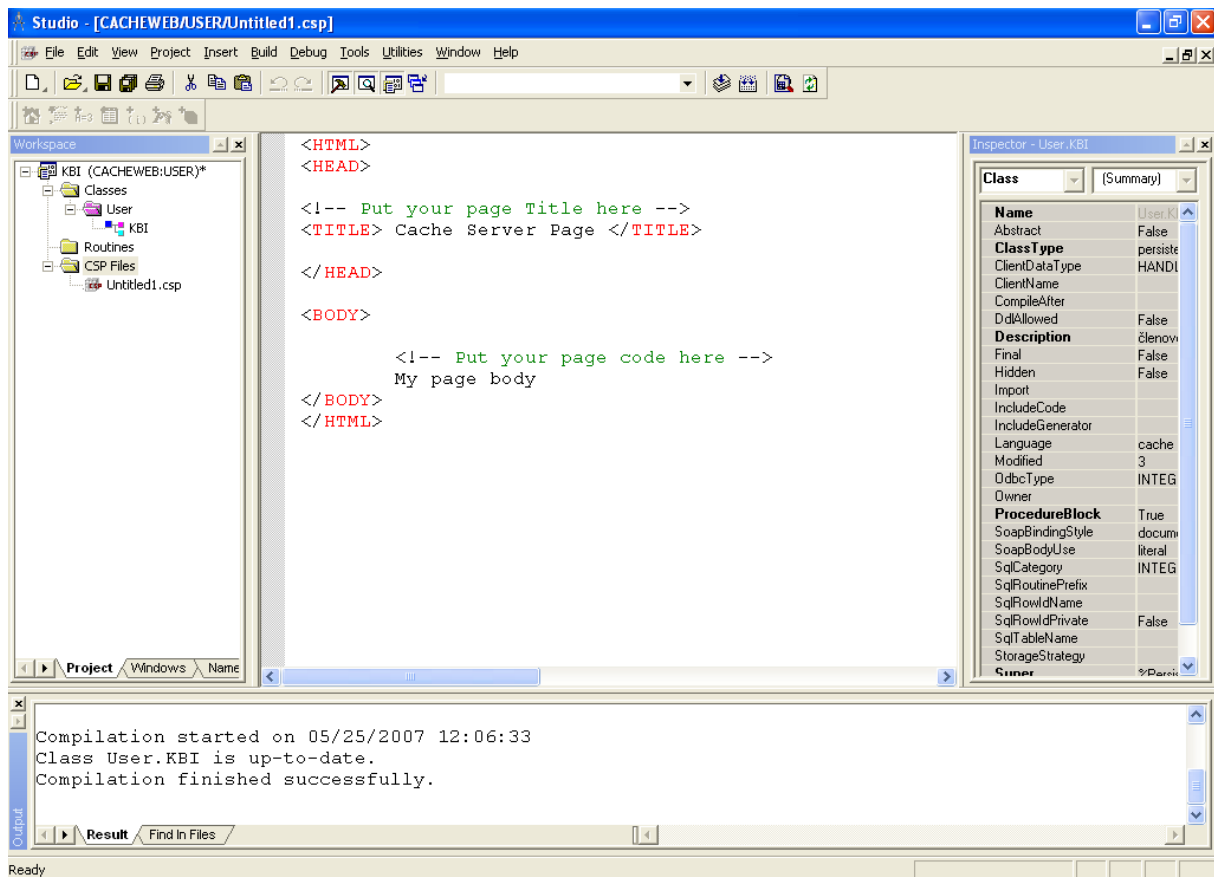
Cvičení 2

Vytvoření webového formuláře pro danou třídu pomocí jazyka *Caché Server Pages (CSP)* prostřednictvím *Caché Web Form Wizard*

Činnosti potřebné pro zvládnutí úlohy :

1. Mít z předchozího *cvičení 1* zkompilovanou (přeloženou třídu) *User.KBI*
2. Vytvoření Webového formuláře pro třídu pomocí *jazyka Caché Server Pages(CSP)*, pomocí průvodce *Caché Web Form Wizard*.

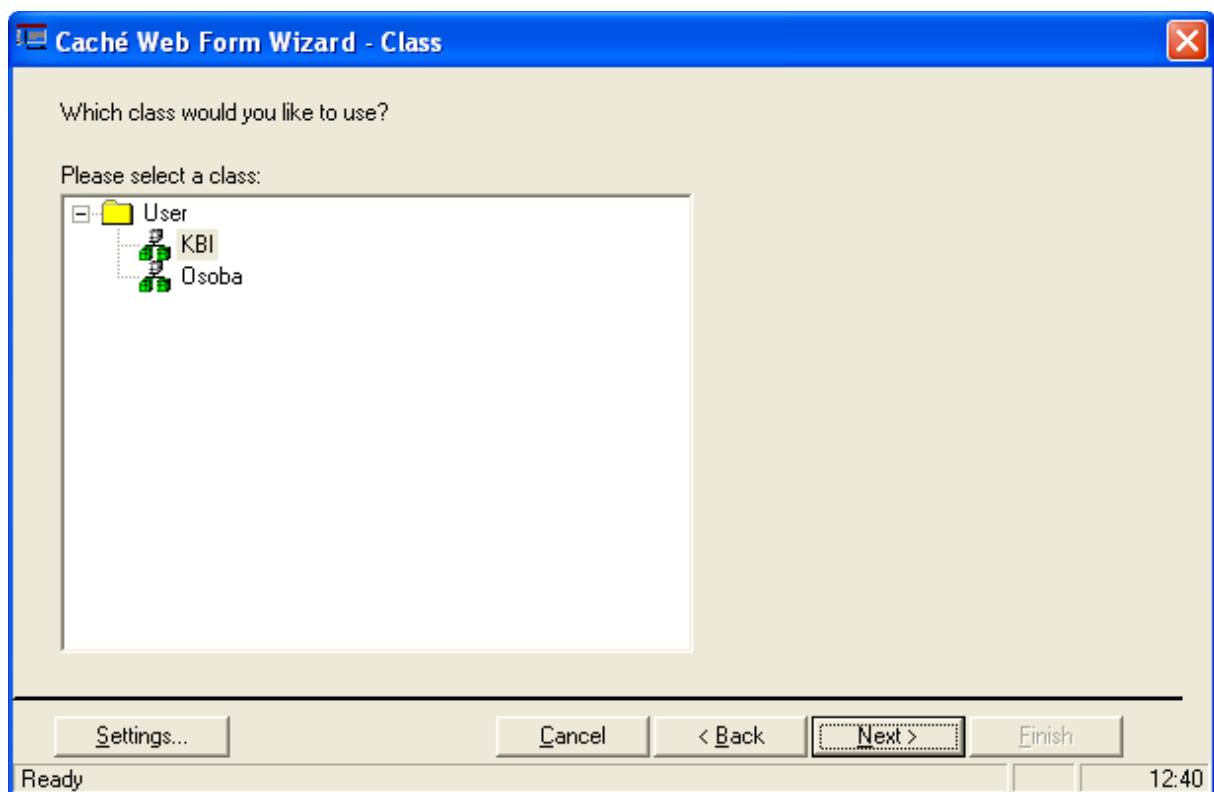
Jestliže máme úspěšně přeloženou třídu *User.KBI*, můžeme přikročit k tvorbě webového formuláře. Nejdříve vytvoříme webový formulář *CSP*, kliknutím na položku *CSP Files* ve struktuře projektu. Z nabídky vybereme položku *Create New CSP File*. Tím dojde k vytvoření nového formuláře *CSP* a jeho zobrazení viz, následující obrázek.



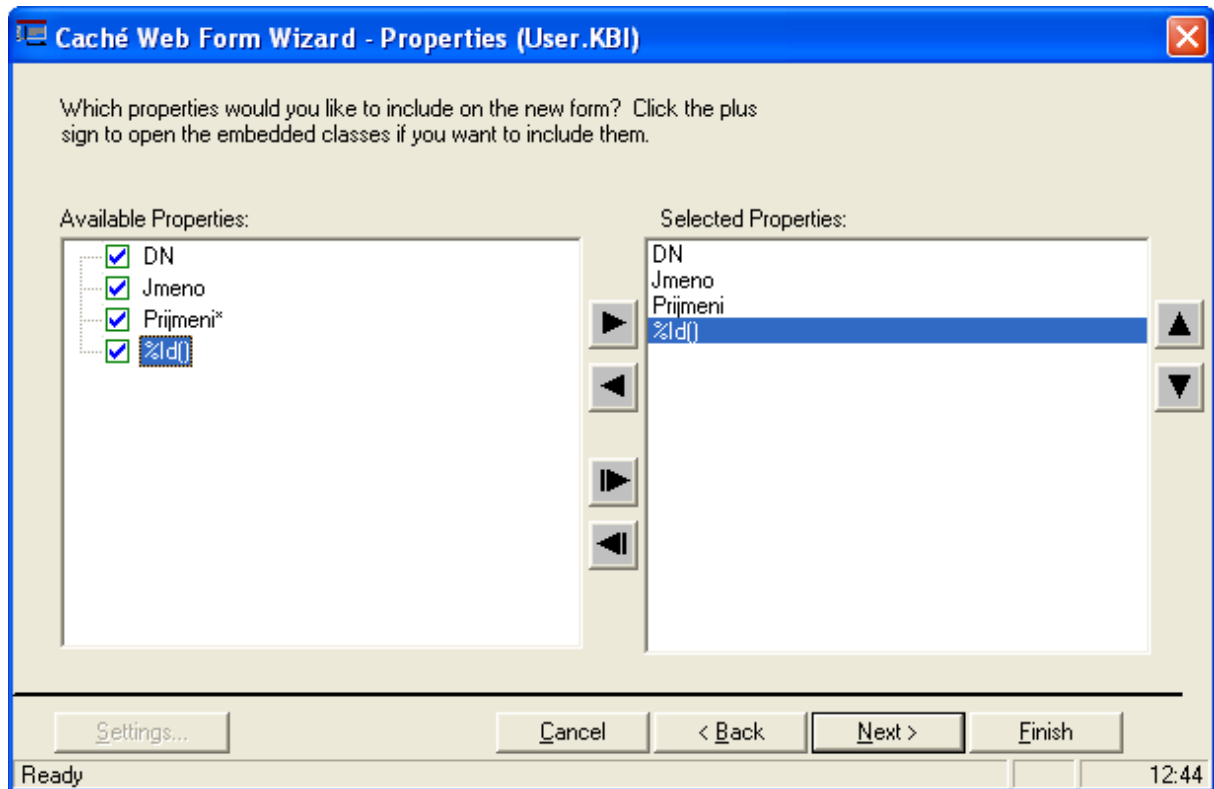
Označíme předpřipravený kód mezi značkami `<BODY>` a `</BODY>`, otevřeme nabídku *Insert* a vybereme položku *Form Wizard*. Spustí se dialog s průvodcem *Caché Web Form Wizard*.



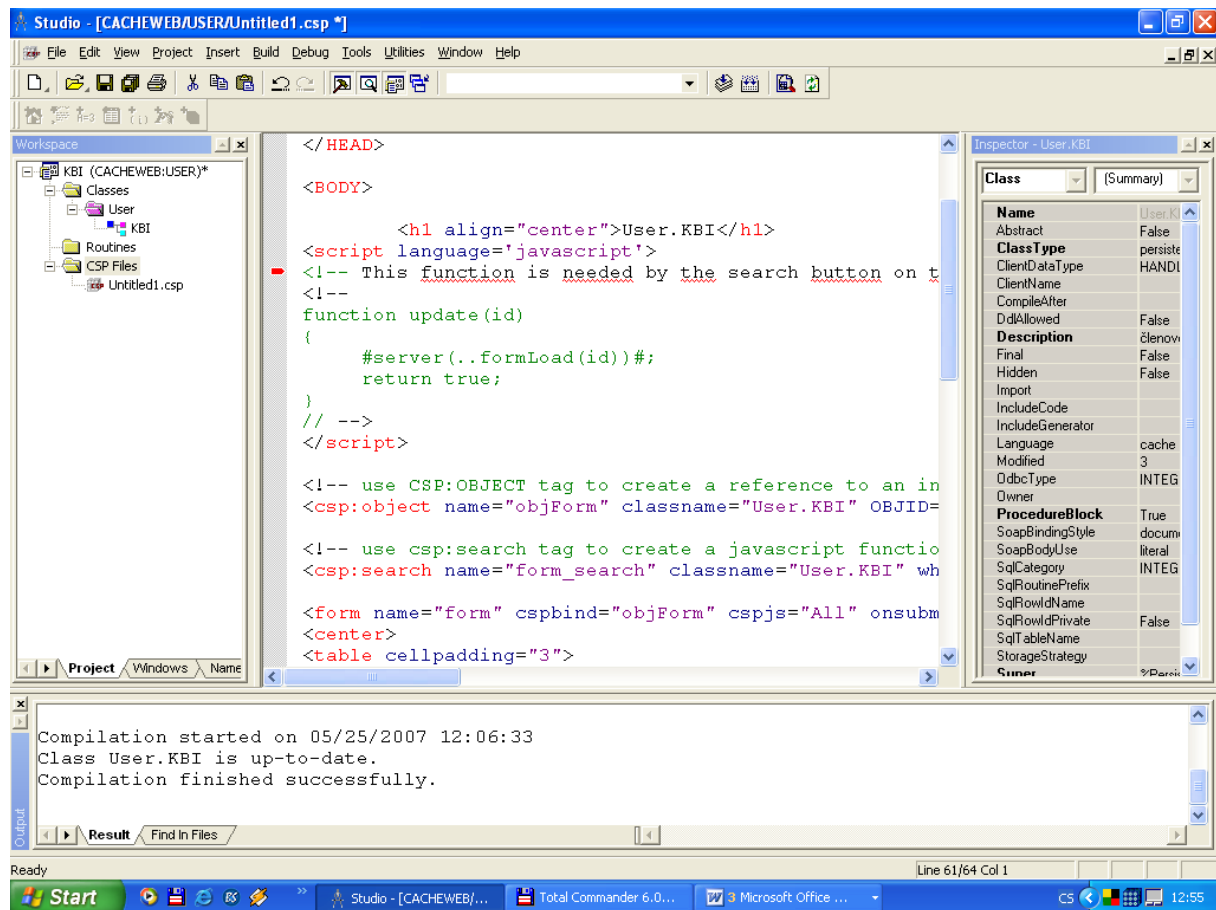
Po stisku *Next* se nám objeví seznam dostupných tříd (naše zkompileovaná třída *User.KBI*), viz následující obrázek.



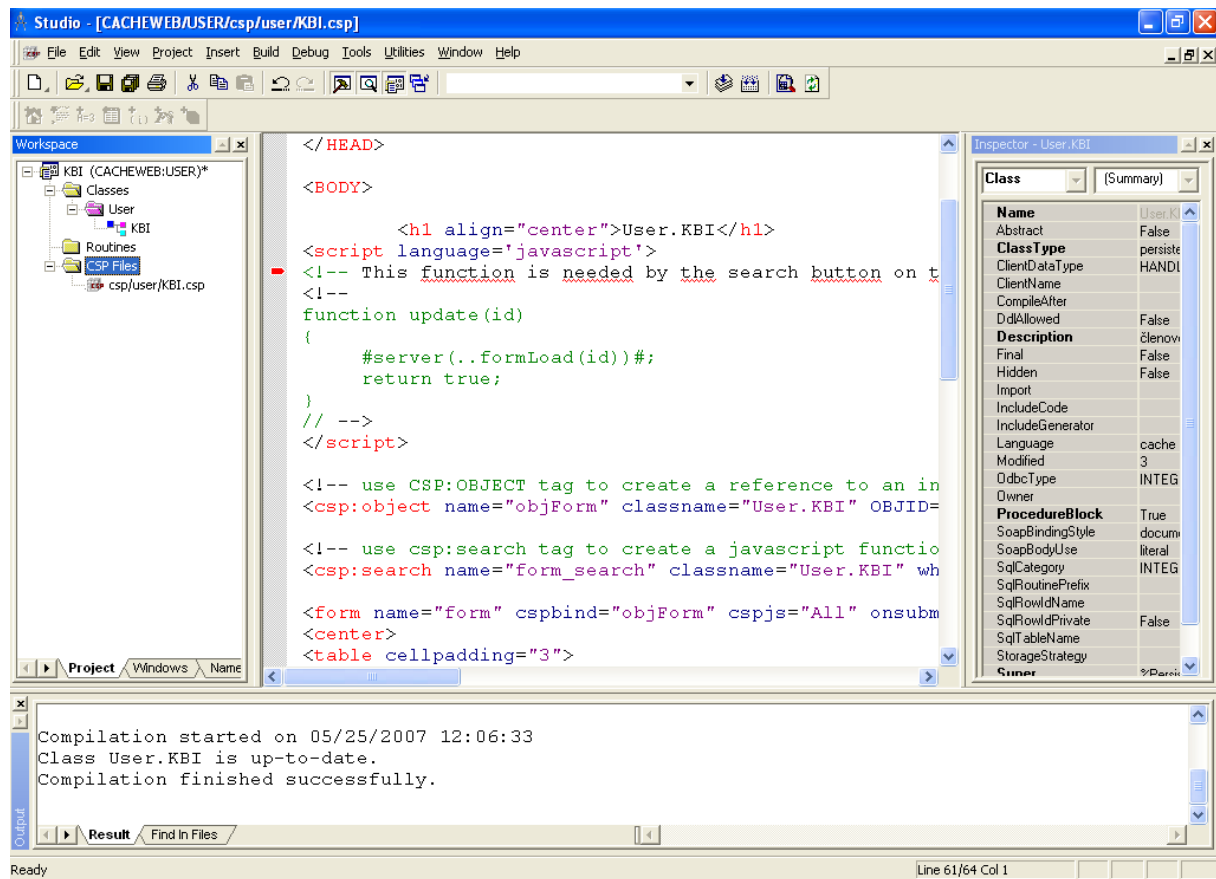
Vybereme třídu pro kterou chceme vytvořit webový formulář a stiskneme *Next*. Objeví se nám vlastnosti třídy, viz následující obrázek.



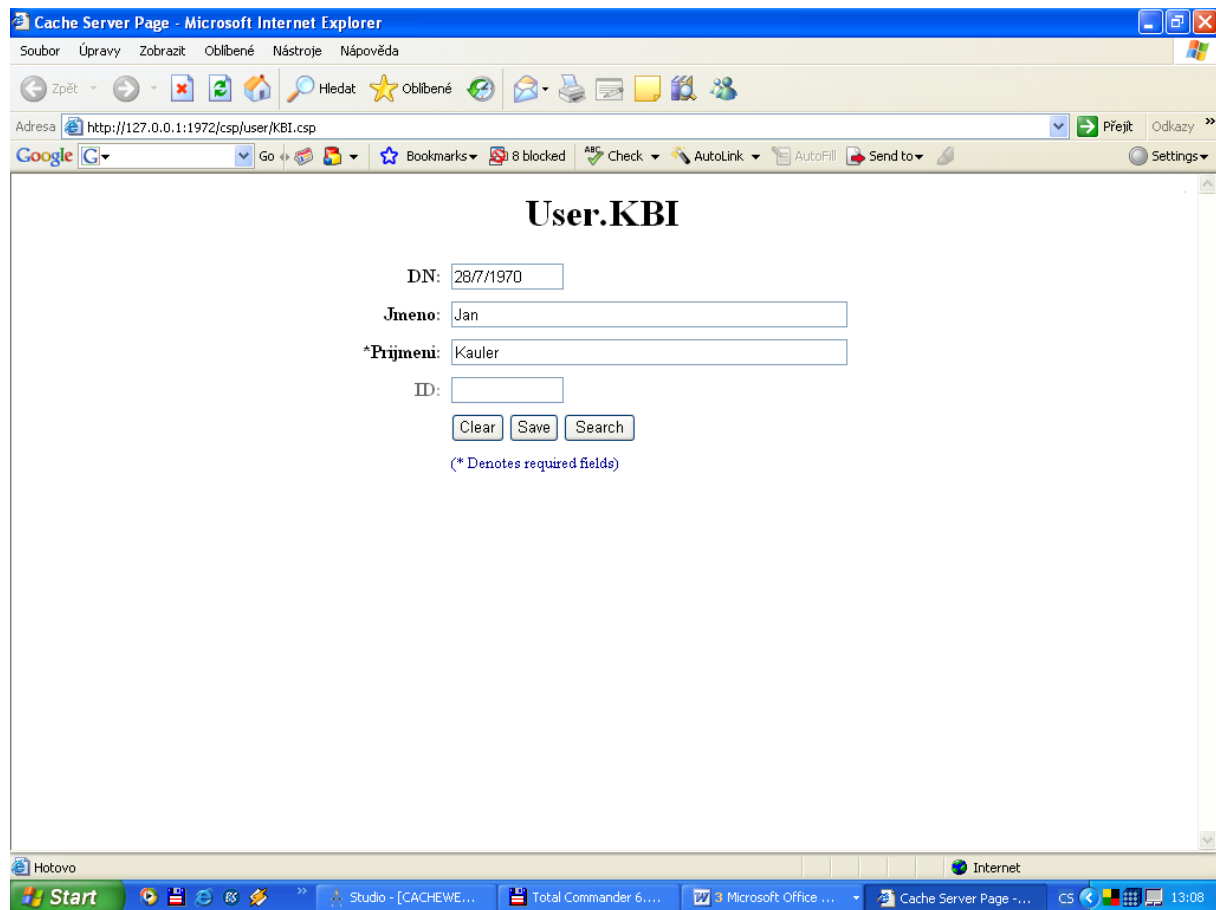
Vybereme všechny vlastnosti (budou použity pro vygenerování formuláře) a stiskneme *Finish*. Mezi značkami <BODY> a </BODY> se objeví vygenerovaný kód, viz. obrázek.



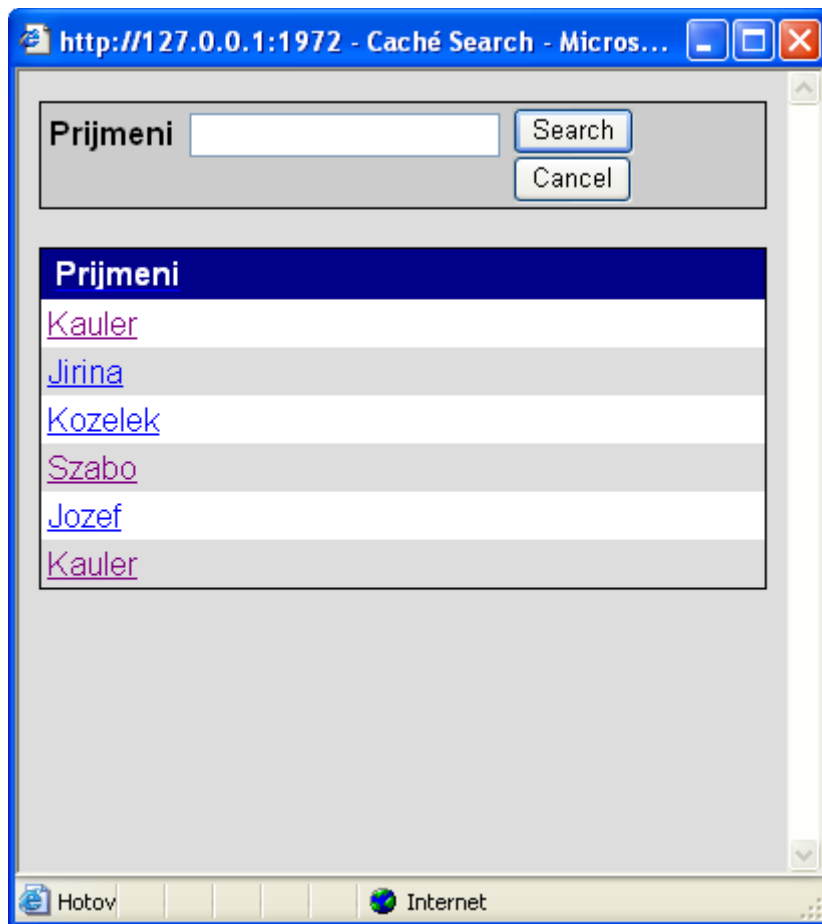
Změnu uložíme *Save As* z nabídky *File*. Formulář nazveme KBI.csp a uložíme jej do adresáře scp/user.



Pro prohlédnutí webové stránky v *Caché Server Pages* vyberte položku *Web Page* z nabídky *View*. Dojde ke spuštění webového prohlížeče a zobrazení automaticky vygenerovaného formuláře pro třídu *User.KBI*.



Vložte několik osob a klikněte na tlačítko *hledat(search)*, čímž vyberete určitou osobu, kterou pak můžete editovat nebo mazat.



Formulář vytvořený pomocí *Caché Web Form Wizard* představuje základní prototyp, který můžete použít pro své vlastní vývojové projekty. Kód je možné editovat a upravovat podle vašich potřeb.

Nyní si vysvětlíme funcce jednotlivých tagů ve stránce CSP :

```
<csp:object name="objForm" classname="User.KBI" OBJID=#(%request.Get("OBJID"))#>
```

Tento tag poskytuje data pro objekt Caché, tj. otevře databázový objekt "User.KBI" a přiřadí mu název "objForm" pod kterým se bude používat na webové stránce (CSP). OBJID je hodnota identifikátoru objektu "objForm"(třídy User.KBI) jež bude bude při volání načten z databáze. Tato hodnota je získána z objektu request (systémová třída %CSP.Request) voláním metody Get() s názvem parametru http dotazu. #()# reprezentuje část kódu vykonávaného na straně serveru, kdy výsledek volání je substituován místo výrazu mezi # #.

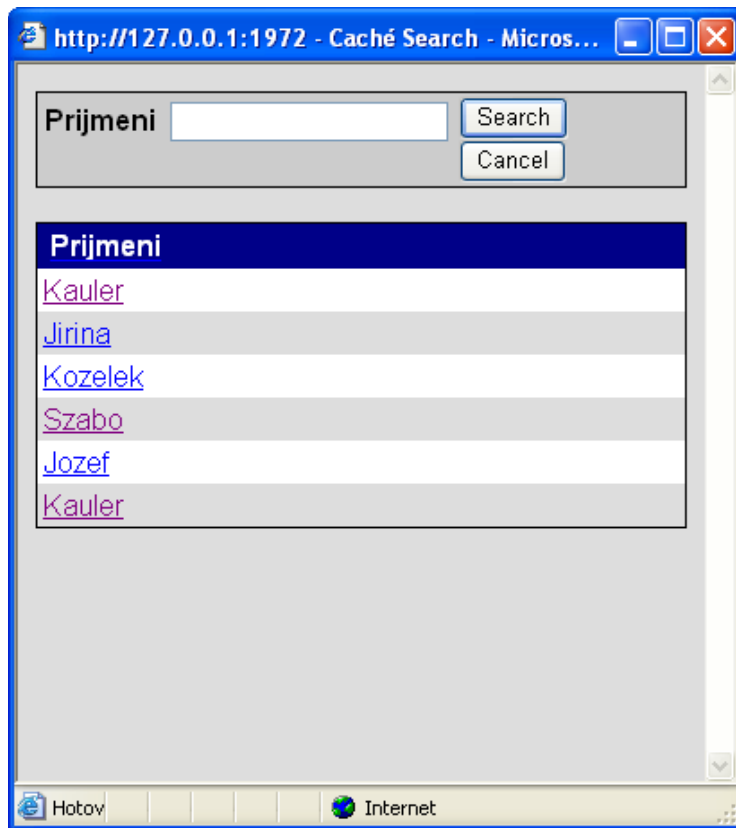
```
<csp:object name="objForm" classname="User.KBI" OBJID=#2+5# >
```

je ekvivalentní zápisu

```
<csp:object name="objForm" classname="User.KBI" OBJID=7>
```

Použitý výraz nám specifikuje napevno hodnotu identifikátoru objektu objForm v tomto případě na hodnotu 7. To by mělo za následek omezení funkčnosti vzorové aplikace na možnost editace pouze jednoho objektu, tj. objekt s ID=7.

```
<csp:search      name="form_search"      classname="User.KBI"      where="Prijmeni"
options="popup,nopredicates" onselect="update">
```



Vytváří Java skriptovou funkci pro vyvolání vyhledávací stránky. Atribut `name="form_search"` představuje její název. Další atributy specifikují parametry vyhledávání. Atribut `classname` určuje databázovou třídu nad jejíž instancemi je prováděno vyhledávání (v tomto případě `"User.KBI"`). Atribut `where` umožňuje specifikovat seznam (oddělený čárkami) vlastností objektu podle nichž bude vyhledávání probíhat. Atribut `options` je seznam přepínačů oddělených čárkami, které blíže konkretizují chování vyhledávací stránky (možnost řazení výsledků, vyhledávací stránka jako nové okno, zobrazování počtu výsledků, zapnutí/vypnutí porovnávacích operátorů)

```
<form name="form" cspbind="objForm" cspjs="All" onsubmit='return form_validate();'>
```

....

```
Jmeno <input type="text" name="Jmeno" cspbind="Jmeno" size="50">
```

...

```
<input type="button" name="btnClear" value="Clear" onclick='form_new();'>
```

```
<input type="button" name="btnSave" value="Save" onclick='form_save();'>
```

```
<input type="button" name="btnSearch" value="Search" onclick='form_search();'>
```

```
</form>
```


Poslední část definuje podobu a chování editačního formuláře. Využívá HTML tag `<form>` do kterého jsou přidány další atributy, umožňující propojení s editovaným databázovým objektem. Atribut `cspbind` specifikuje mapování formuláře na konkrétní instanci editované třídy, jež byla dříve definována tagem `<csp:object>`. Atribut `cspjs` definuje rozsah funkcí, které bude formulář schopen s objektem vykonávat (např. validace hodnot objektu, uložení objektu, vytvoření nové instance třídy).

Jednotlivé vlastnosti objektu jsou mapovány na HTML tag `<input type="text" >` atributem `cspbind` s hodnotou názvu vlastnosti objektu. Ve formuláři jsou deklarovány tlačítka na vytvoření nového objektu, uložení objektu na server a vyvolání vyhledávací stránky.