Projekt SkautSIS

Informační systém pro skautská střediska

Návrhový model

verze 1.0  
8.12.2010  
**Tým SkautSIS**  
Vojtěch Vít, Marek Kožnar, Zbyšek Záliš, Jiří Matoušek, Roman Pivoňka

Cvičící: M. Komárek E-mail: [projekt-is@skaut-ricany.cz](mailto:projekt-is@skaut-ricany.cz)  
Cvičení: ZS-2010 čt 11:00 Web: <https://rabbit.felk.cvut.cz/trac/ISSS>

Historie revizí

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Verze** | **Popis** | **Autor** |
| 8. 12. 2010 | 1.0 | Sestavení dokumentu. | Vojtěch Vít |

Obsah

[1 Předmluva 4](#_Toc279616797)

[2 Model architektury 5](#_Toc279616798)

[2.1 Cross-Cutting Infrastructure Layer 5](#_Toc279616799)

[2.2 SkautIS Services 5](#_Toc279616800)

[2.3 SkautSIS Services 5](#_Toc279616801)

[2.4 SkautSIS WebSite 7](#_Toc279616802)

[3 Návrhový model 9](#_Toc279616803)

[3.1 Balíček "Oznámení" 10](#_Toc279616804)

[3.2 Balíček "Uživatelé" 13](#_Toc279616805)

[4 Model nasazení 15](#_Toc279616806)

[4.1 Model artefaktů 16](#_Toc279616807)

[4.2 Model zařízení 17](#_Toc279616808)

[5 Příloha: Operační kontrakty 18](#_Toc279616809)

[5.1 Oznámení 18](#_Toc279616810)

[5.2 Fotogalerie 19](#_Toc279616811)

# Předmluva

Tato dokumentace obsahuje návrh pouze několika vybraných částí systému SkautIS, neboť v rámci předmětu Y36SIN bude implementována jen malá podmnožina celkově zamýšleného systému.

V dokumentu uvedené návrhové diagramy jsou zamýšleny pouze jako příkladné ukázky fungování a architektury systému, nikoli jako podklad pro poloautomatický přepis do kódu. Není jich proto tolik, a v některých částech jsou modelovány na vyšší úrovni abstrakce, než jaká je typická pro ukázkové diagramy UML 2.0.

# Model architektury

Balíček obsahuje popis logického členění navrhované aplikace na vrstvy, které umožňují snadnou rozšiřitelnost a udržovatelnost aplikace. Jednotlivé vrstvy obsahují další balíčky, do kterých jsou seskupeny třídy, podle jejich logických souvislostí.



Obrázek: 1

## Cross-Cutting Infrastructure Layer

Infrastruktura je vrstva knihoven použitých na všech úrovních systému SkautSIS. Jedná se o podpůrné knihovny, konfigurace součástí .NET Frameworku, externí knihovny a frameworky aj.

## SkautIS Services

Skautský Informační Systém (SkautIS) je celostátní informační systém, jehož data jsou zpřístupněna prostřednictvím webových služeb (protokol SOAP). Tento systém není součástí systému SkautSIS.

SkautSIS bude z tohoto systému čerpat informace o organizačních jednotkách, o členech organizace, kontaktní údaje apod.

## SkautSIS Services

Komponenta "obalující" jádro informačního systému SkautSIS, které se stará např. o správu dat, řízení a sledování business procesů, autorizaci aj.

Veškerý přístup k jádru je zprostředkován prostřednictvím webových služeb, takže tato komponenta může běžet fyzicky na jiném serveru a s jinou technologií implementace, než prezentační vrstva (např. webové stránky).



Obrázek:SkautSIS Services

### Application Service Layer

Vrstva aplikačních služeb je primárním externím rozhraním komponenty SkautSIS Services. Obsahuje webové služby postavené na protokolu SOAP a technologii Windows Communication Foundation, pomocí nichž budou volající pracovat s jádrem systému SkautSIS.

Aplikační služby SkautSIS Services se rovněž starají o správu a řízení Workflows spjatých s tou kterou operací. Tato funkce je implementována technologií Windows Workflow Foundation.

### Domain Layer

Doménová vrstva modeluje entity a hodnotové objekty problémové domény, zachycuje vztahy mezi nimi a modeluje operace, které lze s doménovými objekty provádět.

Třídy v této vrstvě by měly být typu POCO, tj. neměly by být závislé (např. dědičností) na jiné technologii, než na jádru .NET Frameworku.



Obrázek:Domain Layer

#### Domain Objects

Doménové objekty se sestávají z POCO (Plain old CLR object) tříd entit a hodnotových objektů, zachycujících objekty problémové domény a vztahy mezi nimi.

#### Domain Services

Doménové služby jsou třídy obsahující operace, které není vhodné přiřadit přímo k některé ze tříd entit. Obvykle jsou to operace pracující s více než jednou entitou.

#### Repositories

Komponenta repozitářů v doménové vrstvě obsahuje rozhraní předepisující třídy pro persistenci entit. Konkrétní implementace je pak záležitostí vrstvy infrastrukurní.

### Infrastructure Layer for Data Persistence

Infrastrukturní vrstva pro perzistenci dat je soubor implementací rozhraní repozitářů definovaných v doménové vrstvě.

V systému SkautSIS budou tyto implementace volat kombinovaně webové služby systému SkautIS a vlastní databáze (prostřednictvím ORM frameworku) systému SkautSIS. Úkolem implementací repozitářů je tyto dva datové zdroje v případě potřeby agregovat.

V této vrstvě by také měla probíhat autorizace přístupu k datům na základě přiřazených oprávnění a závěrečná validace změn na entitách.

## SkautSIS WebSite

Komponenta dodávané prezentační vrstvy systému SkautSIS - webových stránek postavených na technologii ASP.NET MVC 3.



Obrázek:Presentation Layer

### Controller

Řadiče jsou třídy reagující na události, typicky požadavky uživatele z webových stránek. Pro zpracování požadavku obvykle používají webové služby SkautSIS Services, a následně uživateli vrací pohled (View) naplněný daty z modelu pohledu (ViewModel).

### View

Pohled je třída předpisující podpobu nějaké části primárního výstupu aplikace.

V aplikaci SkautSIS Website bude pro zápis pohledů použita syntaxe ViewEngine Razor, dodávaná v ASP.NET MVC 3.

### ViewModel

Model pohledu je DTO (data transfer object) sloužící pro přenos informací v požadovaném rozsahu a kvalitě směrem od řadiče (controller) k pohledu (view).

# Návrhový model

Balíček obsahuje popis návrhových tříd, jejich vzájemných vztahů, přiřazení zodpovědností a popis spolupráce pro realizaci požadovaných funkčností. Pro grafickou reprezentaci statického modelu využívá UML diagram tříd, kde jsou jednotlivé třídy doplněny o metody, atributy o datové typy (dle zvoleného implementačního jazyka nebo s využitím obecných datových typů), konce asociací jsou doplněny o popisy. Pro grafickou reprezentaci dynamického chování využívá UML sekvenční diagramy, UML diagramy spolupráce, UML diagramy aktivit a UML stavové diagramy.



Obrázek: 1

## Balíček "Oznámení"



Obrázek: 2

### Balíček "Přidání oznámení"



Obrázek: 3

### Balíček "Zobrazení historie oznámení"



Obrázek: 4

## Balíček "Uživatelé"

### Balíček "Odhlášení uživatele"



Obrázek: 5

### Balíček "Přihlášení uživatele"



Obrázek: 6

# Model nasazení

Balíček obsahuje popis umístění jednotlivých částí aplikace na jednotlivé fyzické uzly a způsob jejich komunikace.



Obrázek: 1

## Model artefaktů



Obrázek: 2

## Model zařízení



Obrázek: 3

### ClientDevice

Klientské zařízení může být jakékoli zařízení schopné spustit webový prohlížeč s podporou standardů.

### SkautISServer

Server hostující celostátní informační systém SkautIS.

### SkautSISWebServer

Webový server na kterém běží komponenta SkautSIS Services a/nebo komponenta SkautSIS Website.

### SomeWebServer

Webový server na kterém běží libovolná prezentační vrstva čerpající z webových služeb systému SkautSIS.

# Příloha: Operační kontrakty

## Oznámení

### Přidat oznámení



|  |  |
| --- | --- |
| Název: | EnterData(name,text,date) |
| Vztahy: | Případ užití – přidání oznámení |

#### Vstupní podmínky:

* Přihlášený uživatel přidává nové oznámení

#### Výstupní podmínky:

* Bylo založeno nové oznámení.
* Byl zadán název oznámení v parametru name.
* Byl vložen obsah nového oznámení v parametru text.
* Bylo vyplněno datum nového oznámení v parametru date.
* Oznámení bylo potvrzeno a uloženo v systému.

## Fotogalerie

### Nahrát fotografii

****

|  |  |
| --- | --- |
| Název: | uploadPhoto(photo) |
| Vztahy: | Use Case GAL05 Přidat Fotografii |

#### Vstupní podmínky:

* Uživatel nahrává fotografii

#### Výstupní podmínky:

* Nová instance třídy fotografie F byla vytvořena.
* F.ID byl nastaven
* F.uploadDate byl nastaven na aktuální datum
* F.uploadUserID bylo nastaveno na ID uživatele, který fotografii nahrál
* F.name byl nastaven
* F.description byl nastaven
* F byl asociován s fotogalerií FG podle F.fotogallery = FG.name
* F.eventName byl nastaven podle asociace s FG
* F.eventDate byl nastaven podle asociace s FG