

# Mapový lokátor

---

*Dokument specifikace požadavků*

*Tým SVK (Sloup, Vavříčka, Kořán)*

*Autor: Martin Sloup*

## Historie verzí

Datum	Verze	Popis	Autor
17. května 2009	1.0	Vytvoření dokumentu	Martin Sloup

## Obsah

1	Úvod .....	2
2	Obecné vlastnosti .....	2
3	Specifické požadavky .....	2
3.1	Funkční požadavky .....	2
3.1.1	Použití komponentového modelu .....	2
3.1.2	Načítání polohy objektů .....	2
3.1.3	Požadavky na způsob ovládání .....	2
3.2	Mimofunkční požadavky .....	2
3.2.1	Použitý programovací jazyk .....	2
3.2.2	Nezávislost na implementaci OSGi Frameworku .....	2
3.2.3	Výběr API pro zobrazení map .....	3
3.3	Rozhraní .....	3
3.3.1	Uživatelské rozhraní .....	3
3.3.2	Softwarové rozhraní .....	3
3.3.3	Hardwarové rozhraní .....	3
4	Glosář .....	3
5	Odkazy na zdroje .....	4
6	Přílohy .....	4

## 1 Úvod

Na univerzitě se pracuje v komponentovém modelu OSGi. Rádi by demonstrovali použití komponentového modelu OSGi v aplikaci umožňující uživateli sledování pohybujících se objektů na mapě. Předpokladem je možnost pohybu s mapou její přiblížení, oddálení a výběr zobrazených objektů. Objekty budou aktualizovat svoji polohu po několika vteřinách.

## 2 Obecné vlastnosti

Aplikace umožní uživateli sledovat skupiny objektů na mapě. Tyto skupiny objektů budou přednastaveny administrátorem systému. Uživatel bude moci zvolit, kterou skupinu objektů chce vidět na mapě a kterou ne. Zároveň uživatel bude moci provádět standardní operace s mapou, tj. pohybovat se zorným polem mapy, přibližovat a oddalovat mapu.

## 3 Specifické požadavky

### 3.1 Funkční požadavky

#### 3.1.1 Použití komponentového modelu

Zadavatelem projektu bylo navrženo rozčlenění objektů aplikace do komponent OSGi modelu a navržení rozhraní ke komunikaci mezi komponentami systému. Komponenty jsou navrženy tak, aby je bylo možné za běhu vyměňovat za jiné implementace komponent.

#### 3.1.2 Načítání polohy objektů

Jednou z komponent aplikace je komponenta sloužící k načítání informací o poloze objektů. Pro ukázkou funkční aplikace bylo zvoleno načítání souřadnic ze záznamu trasy ve formátu NMEA. Jedná se prakticky o záznam komunikace GPS zařízení a aplikace, která GPS zařízení používá.

#### 3.1.3 Požadavky na způsob ovládání

Aplikace bude tvořena mapou se skupinou objektů a postraní lištou, kde si bude moci uživatel zvolit, které skupiny objektů chce uživatel vidět a které ne. U mapy bude moci uživatel pohybovat se zorným polem mapy, přibližovat a oddalovat mapu. O každého zobrazeného objektu na mapě bude moci uživatel zobrazit informace.

### 3.2 Mimofunkční požadavky

#### 3.2.1 Použitý programovací jazyk

Jedním z požadavků je použití OSGi modelu při vývoji aplikace, proto jazyk, kterým je aplikace napsána je pevně určen. Programovacím jazykem je tedy Java od verze 1.6.

#### 3.2.2 Nezávislost na implementaci OSGi Frameworku

Zadavatelem projektu je pevně stanovena nezávislost na implementaci OSGi Frameworku. Aplikace je primárně vyvíjena pod implementací Apache Felix, ale není problém ji rozběhnout na implementaci Eclipse Equinox, případně Knopflerfish.

### 3.2.3 Výběr API pro zobrazení map

Při návrhu bylo nutné vybrat API, které bude vykreslovat, jak samotnou mapu, tak i příslušné objekty na mapě. Zadavatel nespécifikoval, jaké API použít. Byla určena pouze jedna podmínka – zobrazení map nejen pro Českou Republiku. Po prozkoumání různých API pro vizualizaci map byla zvolena implementace Google Maps z následujících důvodů:

- 1) podrobnost map
- 2) jednoduché ovládání
- 3) jednoduchá implementace v produktu
- 4) volné podmínky použití

Google Maps API je tvořeno komponentou ve webové stránce, proto bylo nutné zajistit komunikaci mezi webovou stránkou a aplikací v reálném čase. Uživatelské prostředí bude tvořeno webovou stránkou pomocí značkovacího jazyka HTML společně se skriptováním na straně klienta v jazyce JavaScript. Komunikace mezi webovou stránkou a zbytkem aplikace bude zajištěno pomocí REST nad protokolem HTTP v datovém formátu JSON.

## 3.3 Rozhraní

### 3.3.1 Uživatelské rozhraní

Aplikace je ovládána pomocí webového prohlížeče, v kterém je zobrazeno uživatelské prostředí aplikace. Tato webová stránka řeší všechny uživatelské funkce aplikace, tj. výběr viditelné skupiny objektů, zobrazení vybrané skupiny objektů na mapě, práce s mapou, zobrazení informace o objektu, ...

### 3.3.2 Softwarové rozhraní

Zadavatelem aplikace jsou pevně stanoveny rozhraní komponent aplikace. Ty je nutné pevně dodržet, aby bylo možné komponenty aplikace vyměňovat za běhu. Každá komponenta bude tím pádem bundle v OSGi modelu a rozhraní budou implementována pomocí OSGi service (přesněji declarative service).

### 3.3.3 Hardwarové rozhraní

Aplikace používá připojení k internetu, proto je ho nutné mít k dispozici pro správný chod aplikace.

## 4 Glosář

### JSON

JavaScript Object Notation je odlehčený formát pro výměnu dat. Je jednoduše čitelný i zapisovatelný člověkem a snadno analyzovatelný i generovatelný strojem. Je založen na podmnožině programovacího jazyka JavaScript. JSON je textový, na jazyce zcela nezávislý formát, využívající však konvence dobře známé programátorům jazyků rodiny C (C, C++, C#, Java, JavaScript, Perl, Python a dalších). Díky tomu je JSON pro výměnu dat opravdu ideálním jazykem.

### Klient

Klientská část aplikace. V tomto případě se jedná o webovou stránku aplikace, pomocí které je tvořeno uživatelské prostředí.

**NMEA formát**

Neboli NMEA 0183 je standard pro komunikaci s GPS přijímačem vytvořenou organizací National Marine Electronics Association

**OSGi**

OSGi je specifikace, která definuje standard pro vývoj, nasazení a správu aplikací v řízeném prostředí. Řízeným prostředím je OSGi kontejner a aplikací je takzvaný bundle. Zjednodušeně řečeno OSGi kontejner je vlastní prostředí vystavěné nad JVM, které řídí soužití aplikací (takzvaných bundles neboli balíčků) v tomto prostředí - viditelností určitého API počínaje a definicí závislostí konče.

**REST**

Representational State Transfer je koncept pro design distribuované architektury. Distribuovaná architektura v tomto smyslu znamená, že části programu běží na různých strojích a pro svoji komunikaci využívají síť. Pod programem si můžete představit například webovou aplikaci, kde internetový prohlížeč komunikuje s webovým serverem, aplikaci pro výměnu dat mezi finančními institucemi, kde dochází k vzájemnému volání mezi servery.

**Server**

Přijímá požadavky klientské části a předává zpět potřebná data. Kromě poskytovaného REST API, potřebného ke komunikaci klientské části se zbytkem aplikace, předává do prohlížeče potřebnou HTML stránku s uživatelským prostředím a skript, sloužící k popisu ovládání klientské části.

## 5 Odkazy na zdroje

**Popis NMEA formátu**

<http://www.gpsinformation.org/dale/nmea.htm>

**Podrobný popis použití Google Maps API**

<http://code.google.com/intl/cs/apis/maps/>

**Informace o REST**

[http://en.wikipedia.org/wiki/Representational\\_State\\_Transfer](http://en.wikipedia.org/wiki/Representational_State_Transfer)

## 6 Přílohy

### A. Případy užití

**PU 001 Zobrazit mapu*****Aktéři***

Uživatel

***Omezení na stav systému před spuštěním případu užití***

(žádné)

***Kroky případu užití:***

- 1) Uživatel zobrazí v internetovém prohlížeči stránku s aplikací

- 2) Systém načte mapu
- 3) PU002 Načíst seznam skupin objektů

***Omezení na stav systému po ukončení případu užití***

Zobrazená mapa a načtený seznam skupin objektů v postranní liště.

---

**PU 002 Načíst seznam skupin objektů*****Aktéři***

Uživatel

***Omezení na stav systému před spuštěním případu užití***

(žádné)

***Kroky případu užití:***

- 1) Klient se dotáže webového serveru na existující skupiny
- 2) Server sestaví seznam skupin objektů a vrátí je klientovi
- 3) Klient přidá do postranní lišty seznam skupin se zaškrťovacím políčkem.

***Omezení na stav systému po ukončení případu užití***

V postraní liště se nachází seznam skupin objektů

---

**PU 003 Práce s mapou*****Aktéři***

Uživatel

***Omezení na stav systému před spuštěním případu užití***

Zobrazená mapa

***Kroky případu užití:***

- 1) Uživatel může pohybovat mapou prostřednictvím táhnutí mapou nebo přiblížit a oddálit pomocí ovládacích prvků na mapě, nebo:
- 2) Zobrazit informaci o objektu na mapě pomocí kliknutí na ikonku objektu v případě, že objekt je zobrazen na mapě.

***Omezení na stav systému po ukončení případu užití***

(žádné)

---

**PU 004 Vybrat zobrazené skupiny objektů*****Aktéři***

Uživatel

***Omezení na stav systému před spuštěním případu užití***

Zobrazená mapa a načtený seznam skupin objektů v postranní liště.

***Kroky případu užití:***

- 1) Uživatel zaškrtně nebo odškrtně skupinu, kterou chce zobrazit nebo schovat na mapě (lze mít zaškrtnuto více skupin)
- 2) Pokud je skupina odškrtnuta provede se PU 006 Odebrat objekty z mapy a ukončí se případ užití
- 3) Klient odešle požadavek na Server s názvem skupiny.
- 4) Server z rejstříku objektů získá skupinu včetně aktuálních poloh objektů skupiny a vrátí je zpět Klientovi.
- 5) PU 005 Přidat objekty na mapu

***Omezení na stav systému po ukončení případu užití***

Viditelné objekty na mapě.

---

**PU 005 Přidat objekty na mapu*****Aktéři***

Uživatel

***Omezení na stav systému před spuštěním případu užití***

Zobrazená mapa a je k dispozici skupina se seznamem objektů

***Kroky případu užití:***

- 1) Pro každý objekt:
  - a. Přidání objektu na mapu na základě souřadnice polohy objektu
  - b. Nastavení možnosti zobrazení informací o objektu pro 2. krok PU 003 Práce s mapou.

***Omezení na stav systému po ukončení případu užití***

Objekty skupiny jsou vidět na mapě.

---

**PU 006 Odebrat objekty z mapy*****Aktéři***

Uživatel

***Omezení na stav systému před spuštěním případu užití***

Zobrazená mapa a je k dispozici název odškrtnuté skupiny

***Kroky případu užití:***

- 1) Pro každý objekt skupiny: Odebrání objektu z mapy

***Omezení na stav systému po ukončení případu užití***

Objekty skupiny nejsou vidět na mapě.

---

## PU 007 Aktualizovat stav objektů na mapě

### Aktéři

Timer

### Omezení na stav systému před spuštěním případu užití

Zobrazená mapa, jsou zaškrtnuté skupiny

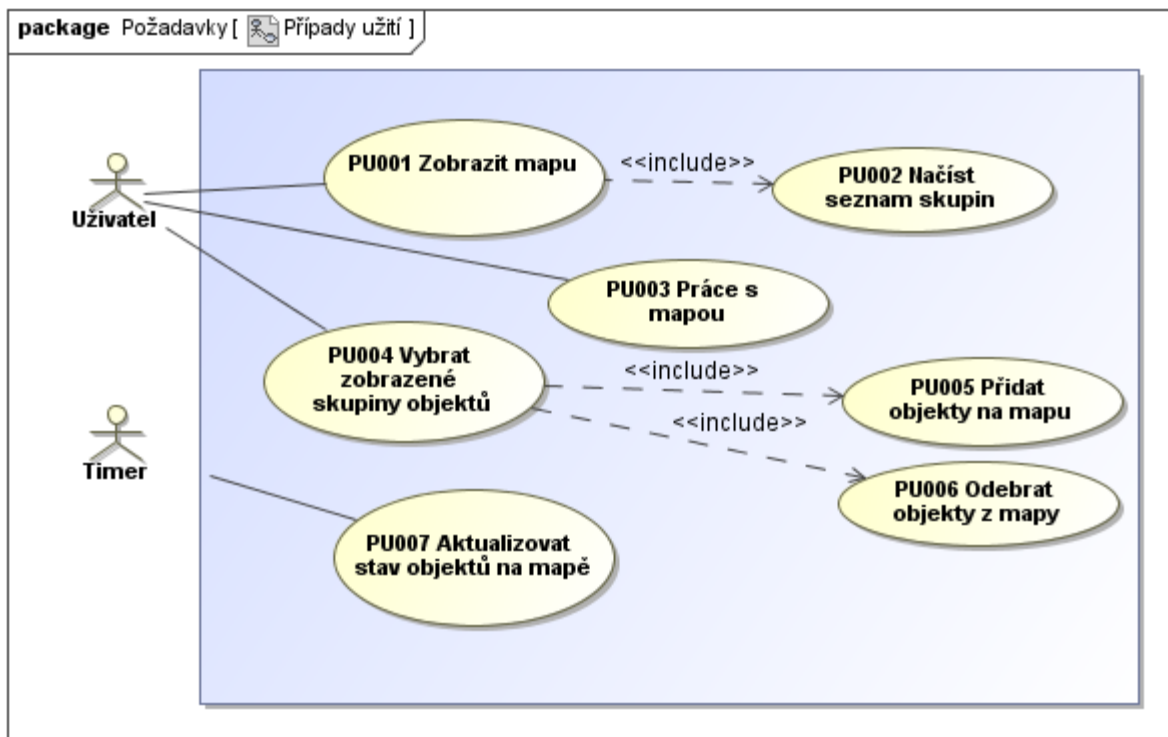
### Kroky případu užití:

- 1) Klient pošle Serveru seznam všech zaškrtnutých skupin
- 2) Server získá všechny skupiny odpovídající názvům zaškrtnutých skupin od rejstříku objektů
- 3) Server pro každou skupinu získá aktuální polohu všech objektů skupiny
- 4) Server vrátí Klientovi seznam skupin a aktuální polohy objektů u každé skupiny.
- 5) Po všechny vrácené skupiny: Klient vybere pro každý objekt skupiny jeho příslušný objekt na mapě a u objektu na mapě nastaví aktuální souřadnice.

### Omezení na stav systému po ukončení případu užití

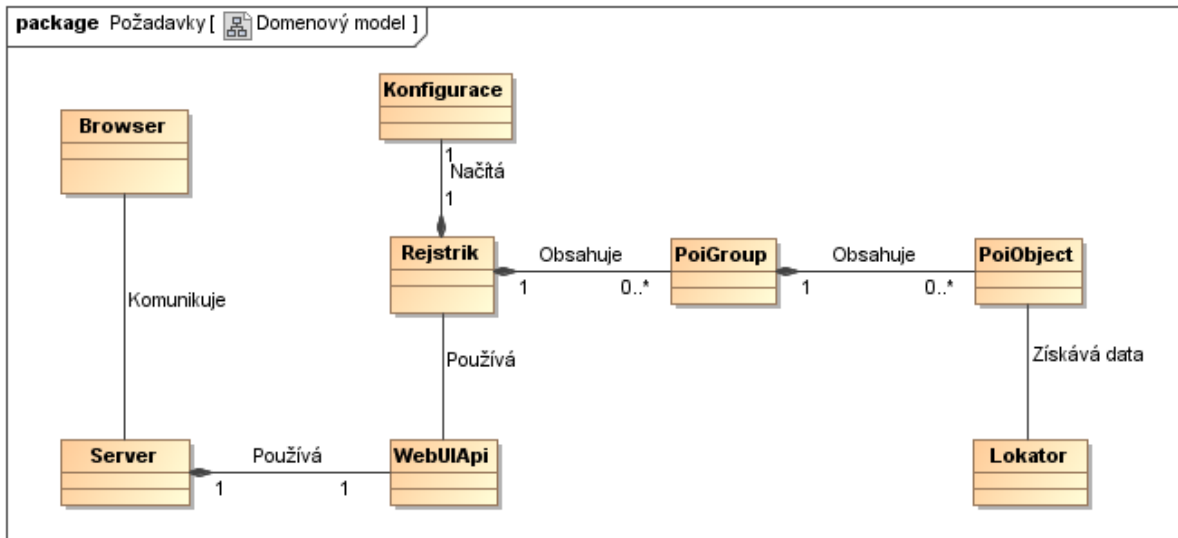
Objekty na mapě změnili svoji pozici.

## B. Diagramy

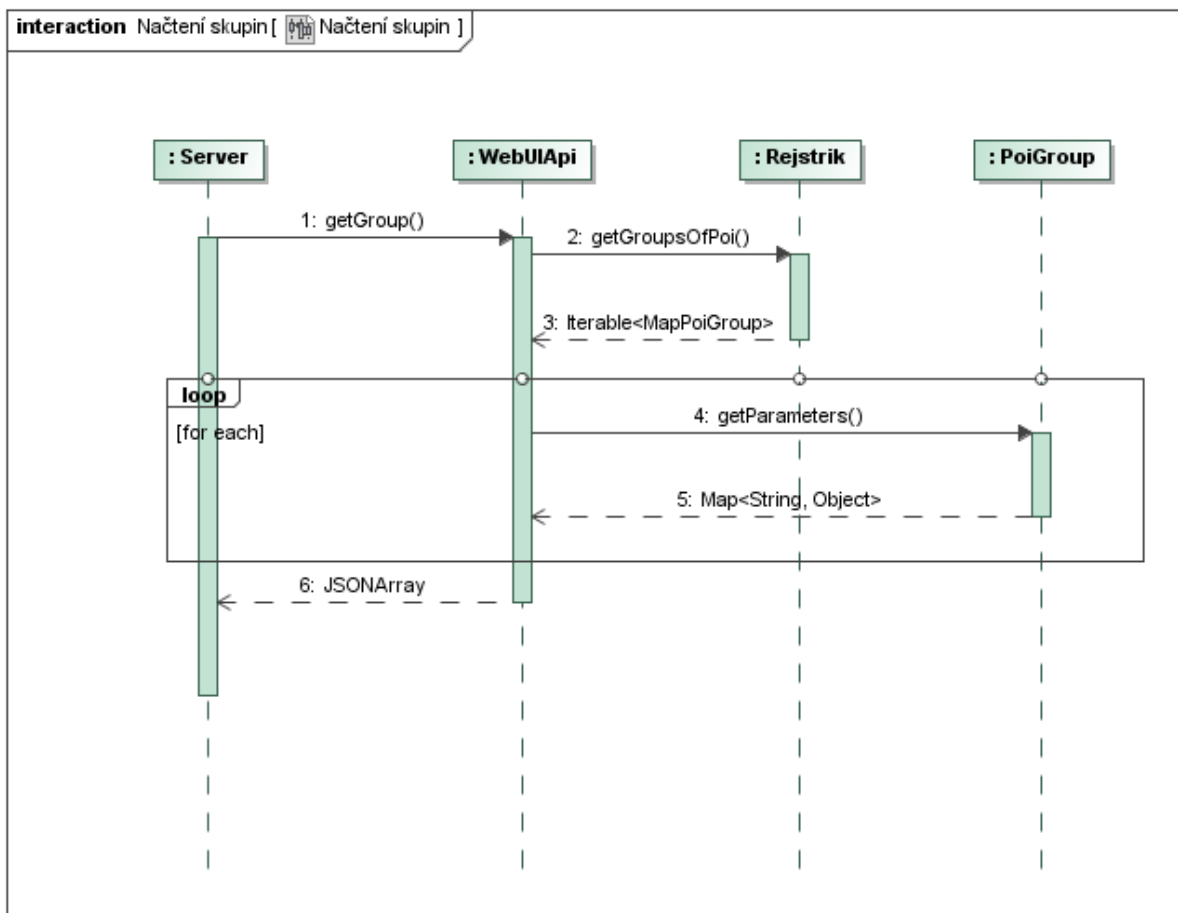


Obr. 1: Případy užití

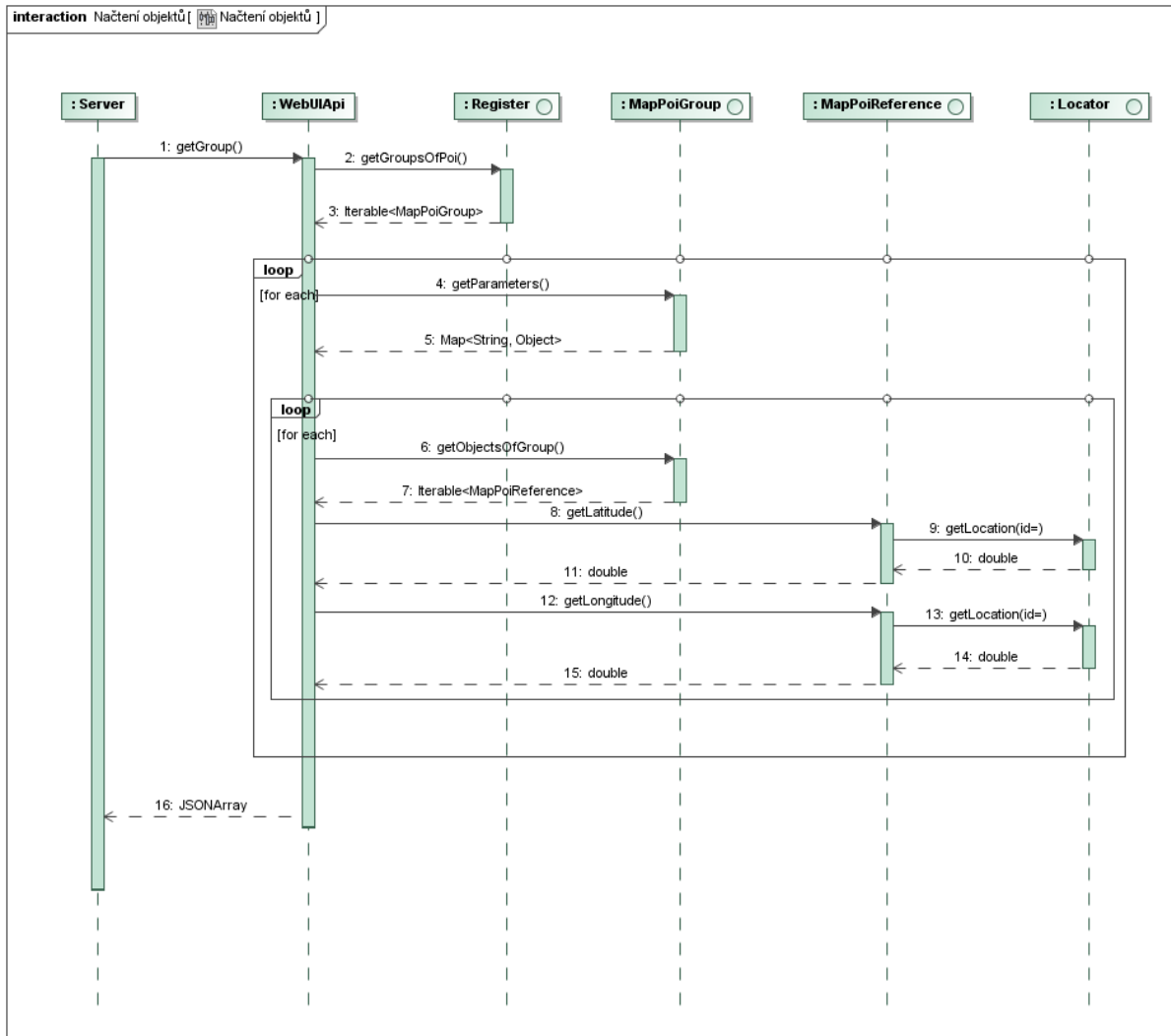




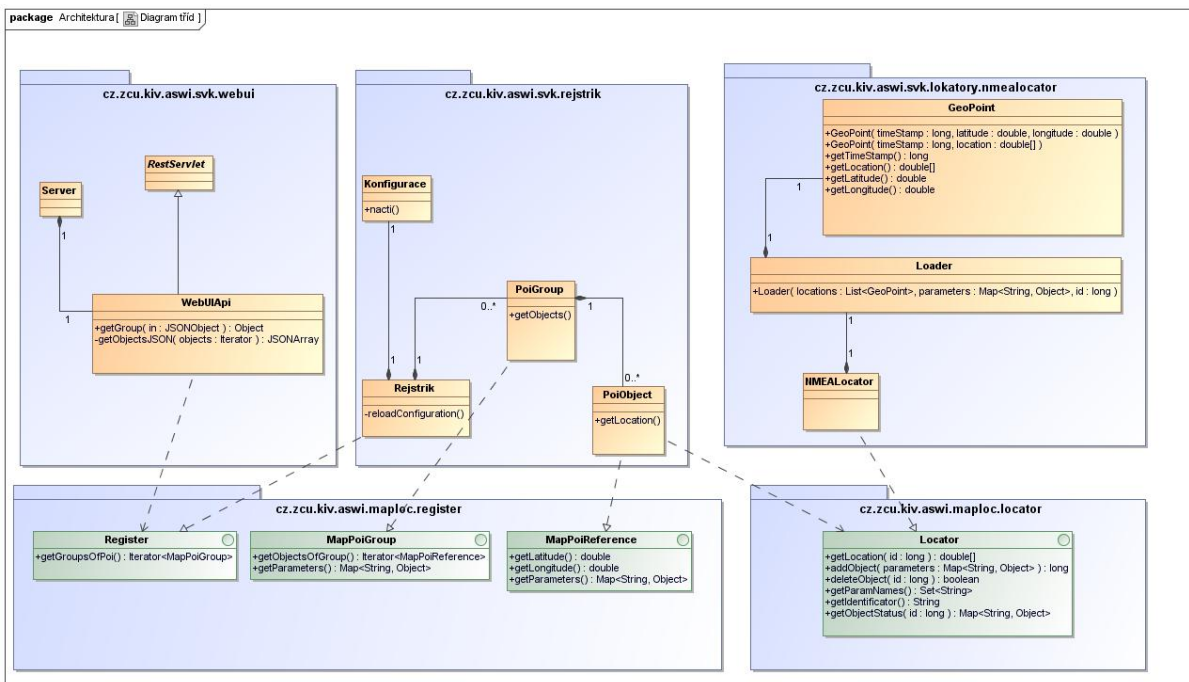
Obr. 2: Doménový model



Obr. 3: Sekvenční diagram - načtení skupin



Obr. 4: Sekvenční diagram - načtení objektů



Obr. 5: Diagram tříd