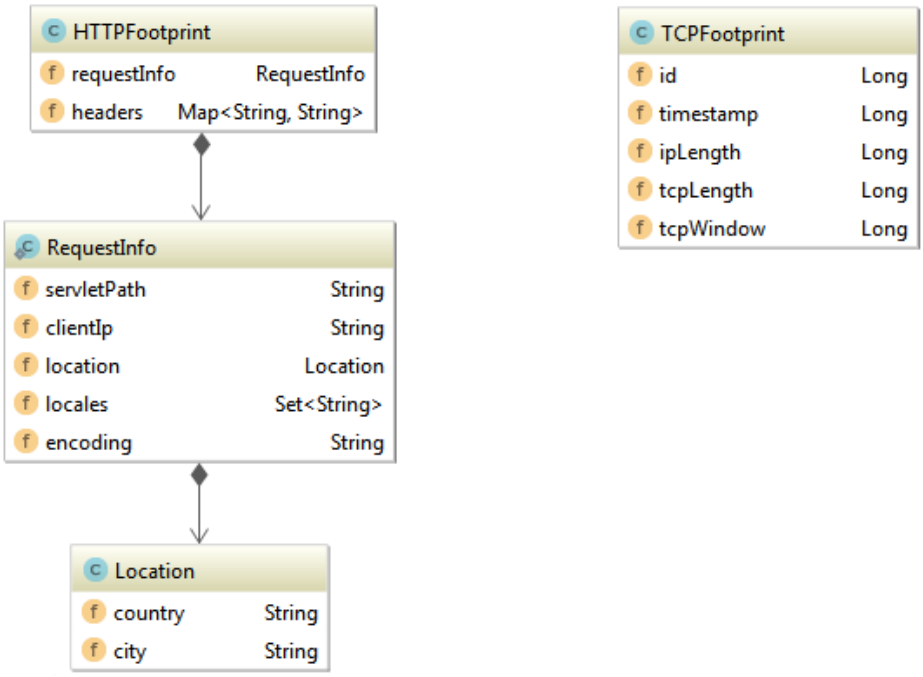
Štruktúra dát UFOO

Hlavná logika knižnice je v momentálnej Java implementácií sústredená v triede *UFooProcessor*. Táto trieda riadi spracovanie informácií získaných z *HTTP* a *TCP* protokolu, ktoré sú na začiatku procesu reprezentované triedami *HTTPFootprint* a *TCPFootprint.*

Nasledujúci diagram popisuje ich štruktúru:



Proces spracovania *requestu* pozostáva zo štyroch základných krokov:

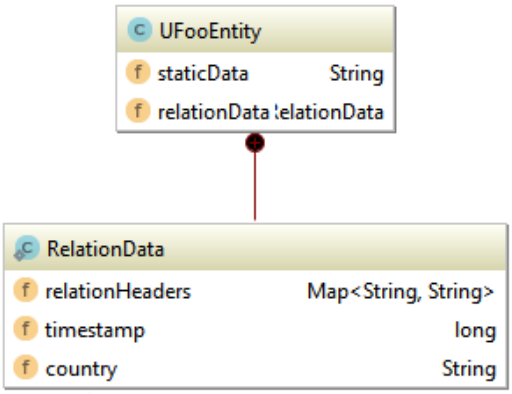
1. **Serializácia dát na reprezentáciu v podobe *UFooEntity***
2. **Analýza statických dát – hľadanie podobnosti**
3. Analýza relačných dát
4. Vyhodnotenie výsledkov

1 Serializácia dát

Ako som už spomínal, takzvaný unikátny odtlačok používateľa je v implementácií reprezentovaný triedou *UFooEntity.* Samotný odtlačok sa skladá z dvoch častí, z ktorých každá ďalej obsahuje dáta určené pre rozdielny typ funkcionality. Týmito časťami sú:

- Statické dáta
- Relačné dáta

Ich implementácia je znázornená na nasledujúcom diagrame:



1.1 Statické dáta

Statické dáta sú pre algoritmus informácie z HTTP a TCP komunikácie, ktorých hodnoty sú vhodné pre porovnávanie a vyhľadávanie podobností jednotlivých *requestov*. Ide teda najmä o hodnoty, ktoré sa nemenia vzhľadom k prebiehajúcej komunikácií jedného užívateľa.

Ich reprezentáciou je reťazec ktorý obsahuje vybrané informácie a HTTP hlavičky uvedené v nasledujúcej tabuľke:

- static headers
- unknown headers hash
- IP address
- country code
- city
- encoding
- locales
- servletPath
- tcpWindow
- tcpLength

```
private static String[] staticHeaders = {
        "Accept",
        "Authorization",
        "Cache-Control",
        "Cookie",
        "Content-Length",
        "Content-Type",
        "User-Agent"
};
```

Tento reťazec ma presne danú nasledujúcu štruktúru:

~

[h1|h2|h3|...|hn|hash(unknown headers)]|

|IP|countryCode|city|encoding|[locales]|path|tcpWindow|length

~

Začiatok tvoria zoradené hodnoty vybraných HTTP hlavičiek nasledované hashom hlavičiek, ktoré sa v *requeste* nachádzali.

Ďalej popisovač obsahuje informácie o IP adrese, lokácií a kódovaní.

Predposlednou informáciou je *servletPath* daného requestu.

Na konci štruktúry sú uvedené TCP informácie, konkrétne ide o *tcpWindow a tcpLength*.

Ako je znázornené vyššie jednotlivé informácie od seba delí oddeľovač – "|".

1.2 Relačné dáta

Relačné dáta slúžia na vyhodnotenie informácií, ktoré sú v priebehu analýzy *requestu* vyhodnocované samostatne, nie však na základe podobnosti z ostatnými dátami ale na základe iných aspektov ich hodnôt. Ide napríklad o určenie mieri bezpečnosti danej krajiny, alebo dĺžky časového rozmedzia jednotlivých *requestov*.

//TODO

• • •

2. Analýza statických dát

Algoritmus pre hľadanie podobností vyhodnotí statické dáta aktuálneho requestu a porovná ich zo statickými dátami odtlačkov, ktoré má uložené v pamäti. Jeho cieľom je nájsť prípadnu zhodu / podobnosť a vrátiť výsledok na ďalšie spracovanie.

• • •