

Univerza v Ljubljani
Fakulteta *za računalništvo
in informatiko*



Elektronsko in mobilno poslovanje

VAJE 7

Jan Meznarič
Simon Vrhovc

Ljubljana
7. 12. 2016



REST

- Representational state transfer
- Tip spletnih storitev
- Omogoča interakcijo z uporabo virov
 - GET `streznik.com/uporabniki/1`
 - POST `streznik.com/uporabniki`
 - ...
- GET, POST, DELETE, PUT, ...
- Brez stanja
- Zahtevki in odgovori v obliki XML ali JSON
- Uporablja protokol HTTP
 - Uporaba statusnih kod (status codes)
 - npr.: 200 OK



JSON

- Format za tekstovno predstavitev podatkov
- Podatki so strukturirani v objekte
- Vsak objekt je množica parov *ime : vrednost*
- Posamezni pari so ločeni z vejicami
- Podatki so v objekte združeni z zavirami oklepaji
- Seznane pišemo v oglate oklepaje
- Vrednosti so lahko tipa
 - Število
 - Niz
 - Boolean
 - Seznam
 - Objekt
 - null



JSON

- Primer: objekt

```
{ "id": 34,  
  "ime": "Peter",  
  "priimek": "Klepec",  
  "uporabniskolme": "peterklepec",  
  "latitude": null,  
  "longitude": null,  
  "nakupovalniListeks": [ 146 ]}
```

- Primer: seznam objektov

```
"zaposleni": [  
  {"ime": "Janez", "priimek": "Novak"},  
  {"ime": "Peter", "priimek": "Klepec"},  
  {"ime": "Peter", "priimek": "Pan"}  
]
```



Volley

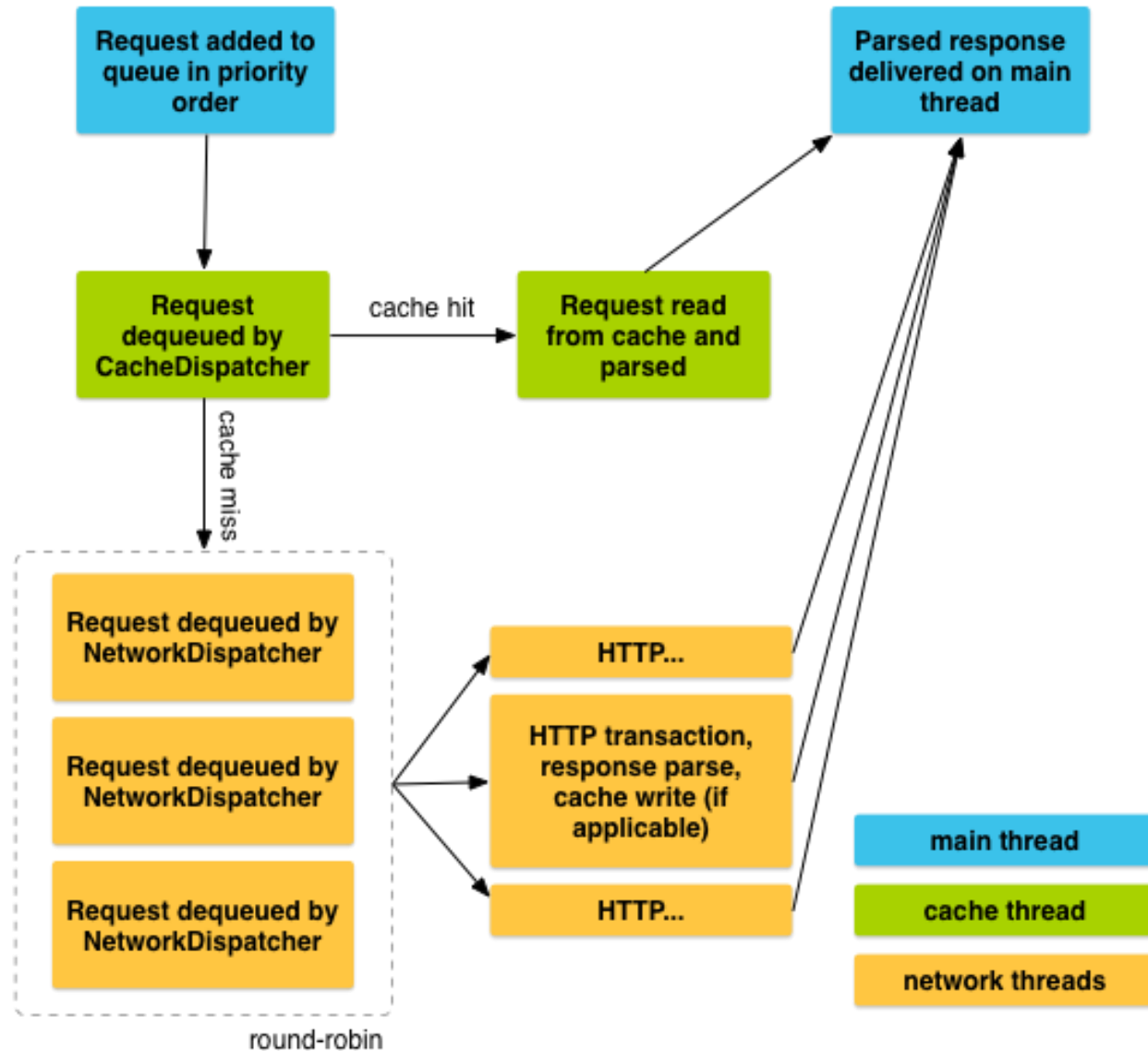
- Knjižnica za delo s spletnimi storitvami REST
- Priporoča se implementacija vrste zahtevkov (*RequestQueue*) po vzorcu edinca (*singleton*)
 - Ena instanca vrste je na voljo skozi celoten življenjski cikel aplikacije
- Knjižnico dodamo v projekt kot odvisnost z orodjem gradle

```
dependencies { ... compile 'com.android.volley:volley:1.0.0' }
```

- <https://developer.android.com/training/volley/index.html>



Volley





Volley

```
public class VolleySingleton {  
  
    private static VolleySingleton mInstance;  
    private RequestQueue mRequestQueue;  
    private static Context mCtx;  
  
    private VolleySingleton(Context context) {  
        mCtx = context;  
        mRequestQueue = getRequestQueue();  
    }  
  
    public static synchronized VolleySingleton getInstance(Context context) {  
        if (mInstance == null) {  
            mInstance = new VolleySingleton(context);  
        }  
        return mInstance;  
    }  
  
    public RequestQueue getRequestQueue() {  
        if (mRequestQueue == null) {  
            mRequestQueue = Volley.newRequestQueue(mCtx.getApplicationContext());  
        }  
        return mRequestQueue;  
    }  
  
    public <T> void addToRequestQueue(Request<T> req) {  
        getRequestQueue().add(req);  
    }  
}
```



Volley

```
// Pridobi RequestQueue
RequestQueue queue = VolleySingleton.getInstance(this).getRequestQueue();

// ...

// Dodaj zahtevek na RequestQueue
VolleySingleton.getInstance(this).addToRequestQueue(stringRequest);
```

- Podpira različne tipe zahtevkov
 - StringRequest
 - ImageRequest
 - JsonObjectRequest
 - JsonArrayRequest
 - Custom Request



Volley

```
JsonObjectRequest jsonObjRequest = new JsonObjectRequest
    (Request.Method.GET, url, null, new Response.Listener<JSONObject>() {

        @Override
        public void onResponse(JSONObject response) {

            // koda, ki se izvede ob uspešnem klicu

        }
    }, new Response.ErrorListener() {

        @Override
        public void onErrorResponse(VolleyError error) {

            // koda, ki se izvede v primeru napake

        }
    });
```



JSONObject

- org.json.JSONObject
- Razred za razčlenjevanje objektov JSON
- Primeren za razčlenjevanje rezultatov, ki nam jih vračajo storitve REST

```
object = new JSONObject(response.toString());  
Log.i(TAG, "Title: " + object.getString("title"));
```



Vaja 13

- Izdelajte aplikacijo, ki bo prikazovala rezultate poizvedbe poljubne storitve REST
- Uporabite knjižnico Volley
- Implementirajte RequestQueue po vzorcu edinca (singleton)
- Uporabite JsonObjectRequest
- Na uporabniškem vmesniku prikažite celoten objekt JSON, ki ga vrača storitev
- Z uporabo JSONObject razčlenite dobljeni objekt
- Na uporabniškem vmesniku prikažite posamezna polja odgovora
- Opomba: aplikaciji morate dovoliti dostop do interneta