

Android: Basics on development

Desenvolvimento de Software e Sistemas Móveis (DSSMV)

Licenciatura em Engenharia de Telecomunicações e Informática

LETI/ISEP

2025/26

Paulo Baltarejo Sousa and Carlos Filipe Freitas

`{pbs,caf}@isep.ipp.pt`

Disclaimer

Material and Slides

Some of the material/slides are adapted from various:

- Presentations found on the internet;
- Books;
- Web sites;
- ...

Outline

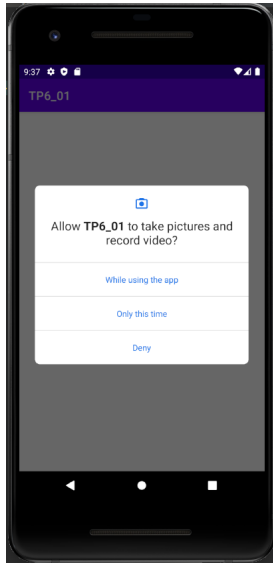
1 Permissions

2 Networking

8 Bibliography

Permissions

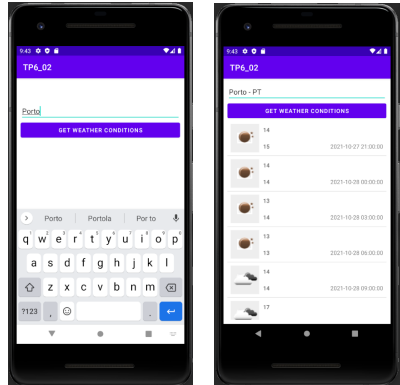
Multiple Permissions



Networking

Networking (I)

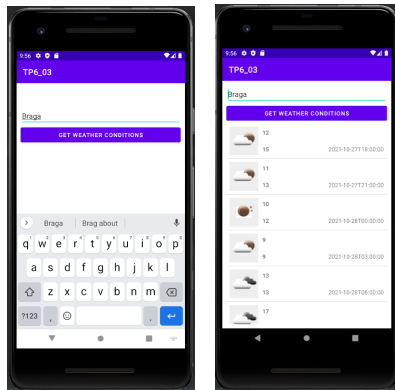
- Getting data from `http://api.openweathermap.org/`
- Data in JSON format



Check: TP6_02.zip

Networking (II)

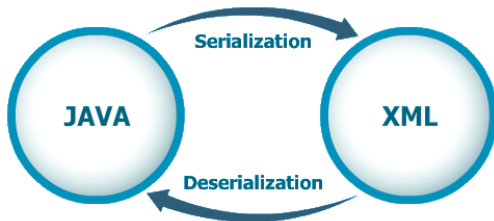
- Getting data from `http://api.openweathermap.org/`
- Data in XML format



Check: TP6_03.zip

Object to/from XML

- **Serialization** is the process of translating Java objects into XML documents
- **Deserialization** is the process of translating XML documents into Java objects .



XML: Example

```
public class User {  
    private int id;  
    private String name;  
    private String profession;  
    public User() {}  
    public int getId() {  
        return id;  
    }  
    public void setId(int id) {  
        this.id = id;  
    }  
    public String getName() {  
        return name;  
    }  
    public void setName(String name) {  
        this.name = name;  
    }  
    public String getProfession() {  
        return profession;  
    }  
    public void setProfession(String  
        profession) {  
        this.profession = profession;  
    }  
}
```

```
<?xml version="1.0" encoding="UTF-8"  
    standalone="yes"?>  
<users>  
    <user>  
        <id>1</id>  
        <name>Maria</name>  
        <profession>Teacher</profession>  
    </user>  
    <user>  
        <id>2</id>  
        <name>Pedro</name>  
        <profession>Teacher</profession>  
    </user>  
    <user>  
        <id>3</id>  
        <name>Joana</name>  
        <profession>Teacher</profession>  
    </user>  
</users>
```

XML: Create a Serializer

```
public String serializer(ArrayList<User> users) {  
    StringWriter writer = new StringWriter();  
    XmlPullParserFactory factory = XmlPullParserFactory.newInstance();  
    XmlSerializer serializer = factory.newSerializer();  
    serializer.setOutput(writer);  
    serializer.startDocument("UTF-8", true);  
    serializer.startTag("", "users");  
    for (User user: users){  
        serializer.startTag("", "user");  
        serializer.startTag("", "id");  
        serializer.text(Integer.toString(user.getId()));  
        serializer.endTag("", "id");  
        serializer.startTag("", "name");  
        serializer.text(user.getName());  
        serializer.endTag("", "name");  
        serializer.startTag("", "profession");  
        serializer.text(user.getProfession());  
        serializer.endTag("", "profession");  
        serializer.endTag("", "user");  
    }  
    serializer.endTag("", "users");  
    serializer.endDocument();  
  
    return writer.toString();  
}
```

XML: Create a Parser (Deserializer)

```
public ArrayList<User> parse(InputStream is) {
    ArrayList<User> users= new ArrayList<User>();
    User user = null; String text="";
    XmlPullParserFactory factory = XmlPullParserFactory.newInstance();
    XmlPullParser parser = factory.newPullParser();
    parser.setInput(is, null);
    int eventType = parser.getEventType();
    while (eventType != XmlPullParser.END_DOCUMENT) {
        String tagname = parser.getName();
        switch (eventType) {
            case XmlPullParser.START_TAG:
                if (tagname.equalsIgnoreCase("user")) { user = new User(); } break;
            case XmlPullParser.TEXT: text = parser.getText(); break;
            case XmlPullParser.END_TAG:
                if (tagname.equalsIgnoreCase("user")) {
                    users.add(user);
                } else if (tagname.equalsIgnoreCase("id")) {
                    user.setId(Integer.parseInt(text));
                } else if (tagname.equalsIgnoreCase("name")) {
                    user.setName(text);
                } break;
        }
        eventType = parser.next();
    }
    return users;
}
```

JSON: Example

```
{ "users": [{ "user": { "id": 1, "profession": "Teacher", "name": "Maria" } },  
  { "user": { "id": 2, "profession": "Teacher", "name": "Pedro" } },  
  { "user": { "id": 3, "profession": "Teacher", "name": "Joana" } } ] }
```

JSON: Create a Serializer

```
public static String serializer (ArrayList<User> data) {  
    try {  
        JSONObject jsonObj = new JSONObject();  
        JSONArray jsonArr = new JSONArray();  
        for (User user : data ) {  
            JSONObject obj = new JSONObject();  
            obj.put("id", user.getId());  
            obj.put("name", user.getName());  
            obj.put("profession", user.getProfession());  
            JSONObject obj1 = new JSONObject();  
            obj1.put("user", obj);  
            jsonArr.put(obj1);  
        }  
        jsonObj.put("users", jsonArr);  
        return jsonObj.toString();  
    } catch (JSONException ex) {  
        ex.printStackTrace();  
    }  
    return null;  
}
```

JSON: Create a Parser (Deserializer)

```
public static ArrayList<User> parser(String data) {
    ArrayList<User> array = new ArrayList<>();
    User user;
    try {
        JSONObject jsonObj = new JSONObject(data);
        JSONArray jsonArr = jsonObj.optJSONArray("users");
        for (int i=0; i < jsonArr.length(); i++) {
            JSONObject obj = jsonArr.getJSONObject(i);
            JSONObject obj1 = obj.getJSONObject("user");
            user = new User();
            user.setId(obj1.getInt("id"));
            user.setName(obj1.getString("name"));
            user.setProfession(obj1.getString("profession"));
            array.add(user);
        }
        return array;
    } catch (JSONException ex) {
        ex.printStackTrace();
    }
    return null;
}
```

Bibliography

Resources

- "Mastering Android Application Development", by Antonio Pachon Rui, 2015
- <https://developer.android.com/index.html>
- <http://simple.sourceforge.net/home.php>
<http://simple.sourceforge.net/download/stream/doc/tutorial/tutorial.php>