

# *Développement Application Android*

## *Webservice*

Anthony Chomienne

CPE Lyon

 2020

# DÉVELOPPEMENT APPLICATION ANDROID — PLAN

1 WEBSERVICE REST

2 CONCLUSION

- Récupérer des données

- Récupérer des données
- Synchroniser des données

- Récupérer des données
- Synchroniser des données
- Traiter des données

- Récupérer des données
- Synchroniser des données
- Traiter des données
- Envoyer des données

- Récupérer des données
- Synchroniser des données
- Traiter des données
- Envoyer des données
- Protocole Http/Https

- Ancien



# SOAP

- Ancien
- Message et réponse en XML

# SOAP

- Ancien
- Message et réponse en XML
- Pas facile à manipuler dans tous les langages

# SOAP

- Ancien
- Message et réponse en XML
- Pas facile à manipuler dans tous les langages
- Uniquement Requête POST

# SOAP

- Ancien
- Message et réponse en XML
- Pas facile à manipuler dans tous les langages
- Uniquement Requête POST
- Fort lien entre client et serveur

# REST

- Plus récent

# REST

- Plus récent
- Réponse dans le format du webservice JSON, CSV, RSS,...

# REST

- Plus récent
- Réponse dans le format du webservice JSON, CSV, RSS,...
- Requête basée sur http

# REST

- Plus récent
- Réponse dans le format du webservice JSON, CSV, RSS,...
- Requête basée sur http
- Requête GET/POST/PUT/DELETE/...



# URL ET URLConnection/HttpsURLConnection

- ```
URL url = new  
URL("http://webserviceurl.com/?param=12&id=22");
```

# URL ET URLConnection/HttpsURLConnection

- `URL url = new URL("http://webserviceurl.com/?param=12&id=22");`
- `URLConnection c;`

# URL ET URLConnection/HTTPSURLConnection

- `URL url = new URL("http://webserviceurl.com/?param=12&id=22");`
- `URLConnection c;`
- `c = (URLConnection) url.openConnection();`

# URL ET URLConnection/HttpsURLConnection

- `URL url = new URL("http://webserviceurl.com/?param=12&id=22");`
- `URLConnection c;`
- `c = (URLConnection) url.openConnection();`
- `URLConnection/HttpsURLConnection`

# URL ET URLConnection/HttpsURLConnection

- `URL url = new URL("http://webserviceurl.com/?param=12&id=22");`
- `URLConnection c;`
- `c = (URLConnection) url.openConnection();`
- `URLConnection/HttpsURLConnection`
- Méthode Get/Post/...

# URL ET URLConnection/HttpsURLConnection

- `URL url = new URL("http://webserviceurl.com/?param=12&id=22");`
- `URLConnection c;`
- `c = (URLConnection) url.openConnection();`
- `URLConnection/HttpsURLConnection`
- Méthode Get/Post/...
- Manipulation de la requête

# URL ET URLConnection/HttpsURLConnection

- `URL url = new URL("http://webserviceurl.com/?param=12&id=22");`
- `URLConnection c;`
- `c = (URLConnection) url.openConnection();`
- `URLConnection/HttpsURLConnection`
- Méthode Get/Post/...
- Manipulation de la requête
- `InputStream estream = c.getErrorStream();`

# MÉTHODE GET

- `c.setRequestMethod("GET");`



# MÉTHODE GET

- `c.setRequestMethod("GET");`
- `c.setDoInput(true);`

# MÉTHODE GET

- `c.setRequestMethod("GET");`
- `c.setDoInput(true);`
- `c.setRequestProperty("Accept", "*/*");`

# MÉTHODE GET

- `c.setRequestMethod("GET");`
- `c.setDoInput(true);`
- `c.setRequestProperty("Accept", "*/*");`
- `InputStream istream = c.getInputStream();`

# MÉTHODE POST

- `c.setRequestMethod("POST");`

# MÉTHODE POST

- `c.setRequestMethod("POST");`
- `c.setDoOutput(true);`

# MÉTHODE POST

- `c.setRequestMethod("POST");`
- `c.setDoOutput(true);`
- `c.setRequestProperty("Accept", "*/*");`

# MÉTHODE POST

- `c.setRequestMethod("POST");`
- `c.setDoOutput(true);`
- `c.setRequestProperty("Accept", "*/*");`
- `OutputStream ostream = c.getOutputStream();`

# INPUTSTREAM

- ```
InputStreamReader isr = new  
    InputStreamReader(istream);
```



# INPUTSTREAM

- `InputStreamReader isr = new  
InputStreamReader(istream);`
- `BufferedReader br = new BufferedReader(isr);`

# INPUTSTREAM

- `InputStreamReader isr = new  
InputStreamReader(istream);`
- `BufferedReader br = new BufferedReader(isr);`
- `String data = br.readLine(); //while data != null`

# INPUTSTREAM

- `InputStreamReader isr = new  
InputStreamReader(istream);`
- `BufferedReader br = new BufferedReader(isr);`
- `String data = br.readLine(); //while data != null`
- `istream.close();`

# OUTPUTSTREAM

- `DataOutputStream dos = new  
DataOutputStream(ostream);`

# OUTPUTSTREAM

- `DataOutputStream dos = new  
DataOutputStream(ostream);`
- `String data = "message to send";`

# OUTPUTSTREAM

- `DataOutputStream dos = new  
DataOutputStream(ostream);`
- `String data = "message to send";`
- `dos.writeBytes(data);`

# OUTPUTSTREAM

- `DataOutputStream dos = new  
DataOutputStream(ostream);`
- `String data = "message to send";`
- `dos.writeBytes(data);`
- `dos.flush();`

# OUTPUTSTREAM

- `DataOutputStream dos = new  
DataOutputStream(ostream);`
- `String data = "message to send";`
- `dos.writeBytes(data);`
- `dos.flush();`
- `ostream.close();`



# PARSER DU JSON

```
{ "arrayname" :  
  [  
    { "property":15, "other":"data"},  
    { "property" : 18, "other":"some"}  
  ]  
}
```

- `JSONObject rootObj = new JSONObject(jsonString);`

# PARSER DU JSON

- `JSONObject rootObj = new JSONObject(jsonString);`
- `JSONArray array = rootObj.getJSONArray("arrayname");`

# PARSER DU JSON

- `JSONObject rootObj = new JSONObject(jsonString);`
- `JSONArray array =  
rootObj.getJSONArray("arrayname");`
- `JSONObject obj = array.getJSONObject(0)`

# PARSER DU JSON

- `JSONObject rootObj = new JSONObject(jsonString);`
- `JSONArray array =  
rootObj.getJSONArray("arrayname");`
- `JSONObject obj = array.getJSONObject(0)`
- `int val = obj.getInt("property");`

# DÉVELOPPEMENT APPLICATION ANDROID — PLAN

1 WEBSERVICE REST

2 CONCLUSION

# SOURCES ET BIBLIO

- <http://www.idc.com/prodserv/smartphone-os-market-share.jsp>
- **Part de marché Mobile en France Janvier 2015**
- <http://opensignal.com/reports/2015/08/android-fragmentation/>
- <http://developer.android.com/>
- <http://www.statista.com/>
- <https://medium.com/@ankit.sinhal/mvc-mvp-and-mvvm-design-pattern-6e169567bbad>

Licence : 