



Projecte missatgeria

Crearem una app Android per gestionar una senzilla missatgeria. L'objectiu del projecte es aplicar tots els coneixements vists durant el curs.

La nostra app gestionarà la part de client d'un servei de missatgeria. La part de servidor està implementada amb un servidor REST, podem consultar les possibilitats del servidor a:

<https://iesmantpc.000webhostapp.com/public/ajuda/>

El primer que hem de fer es comprovar l'estat de la xarxa. Tindrem una funcionalitat mínima en el cas de que la xarxa no estigui disponible.

La segona passa serà la validació del usuari. Aquesta consisteix en recollir un nom i un usuari del usuari i validar-lo. Per fer això com es pot veure a l'ajuda de l'API haurem de fer una cridada POST i enviar dos paràmetres (nom i password). Haurem per tant de modificar el codi vist a la pràctica de JSON per tal d'incloure aquest fet.

```
public static String CridadaPost(String adrecaURL, HashMap<String,
String> parametres) {
    String resultat="";
    try {
        URL url = new URL(adrecaURL);
        Log.i("ResConnectUtils", "Connectant"+adrecaURL);
        HttpURLConnection httpConn = (HttpsURLConnection) url.openConnection();
        httpConn.setReadTimeout(15000);
        httpConn.setConnectTimeout(25000);
        httpConn.setRequestMethod("POST");
        httpConn.setDoInput(true);
        httpConn.setDoOutput(true);
        OutputStream os = httpConn.getOutputStream();
        BufferedWriter writer = new BufferedWriter(
            new OutputStreamWriter(os, "UTF-8"));
        writer.write(montaParametres(parametres));
        writer.flush();
        writer.close();
        os.close();
        int resposta = httpConn.getResponseCode();
        if (resposta == HttpURLConnection.HTTP_OK) {
            String line;
            BufferedReader br=new BufferedReader(new
InputStreamReader(httpConn.getInputStream()));
            while ((line=br.readLine()) != null) {
                resultat+=line;
            }
            Log.i("ResConnectUtils", resultat);
        }
        else {
            resultat="";
            Log.i("ResConnectUtils","Errors:"+resposta);
        }
    } catch (Exception ex) {
        ex.printStackTrace();
    }
    return resultat;
}
```

```
}  
private static String montaParametres(HashMap<String, String> params) throws  
UnsupportedEncodingException {  
    // A partir d'un hashmap clau-valor cream  
    //          clau1=valor1&clau2=valor2&...  
    StringBuilder result = new StringBuilder();  
    boolean first = true;  
    for(Map.Entry<String, String> entry : params.entrySet()){  
        if (first) { first = false;} else {result.append("&");}  
        result.append(URLEncoder.encode(entry.getKey(), "UTF-8"));  
        result.append("=");  
        result.append(URLEncoder.encode(entry.getValue(), "UTF-8"));  
    }  
    return result.toString();  
}
```