TD04 Correction

Matière : ATELIER DEVELOPPEMENT MOBILE AVANCE | **Classes : SEM31**

```
protected void ajouter() {
        RequestQueue queue = Volley.newRequestQueue(this);
        String url = " http://192.168.10.4 :80/MedSpec/Ajout.php";
        StringRequest sr = new StringRequest(Request.Method.POST, url,
            new Response.Listener<String>() {
              @Override
              public void onResponse(String response) {
         try{
             JSONObject json = new JSONObject(response);
               String reponse = json.getString("ETAT");
               if (reponse.equals("SUCCES"))
                  finish();
               else {
                Toast t = Toast.makeToast(Ajout.this,"Problème dans
                            Ajout!",Toast.LENGTH_LONG);
                     t.show() ;
                }
              } catch (JSONException error) {
                   Toast t = Toast.makeText(Ajout.this, "Problème d'analyse JSON: " +
                         error.getMessage(), Toast.LENGTH_LONG);
                    t.show();
             }
             }
            }, new Response.ErrorListener() {
                @Override
                public void onErrorResponse(VolleyError e) {
                   Toast t = Toast.makeText(Ajout.this, "Problème d'appel HTTP: " +
                         e.getMessage(), Toast.LENGTH_LONG);
                t.show();
                }
              }) {
         @Override
         public Map<String, String> getParams() throws AuthFailureError {
           HashMap<String, String> headers = new HashMap<String, String>();
           headers.put("nom", edNom.getText().toString());
           headers.put("prenom", edPrenom.getText().toString());
           headers.put("spec", edSpec.getText().toString());
           headers.put("adresse", edAdresse.getText().toString());
           headers.put("tel", "val" edTel.getText().toString());
           return headers;
         }
         };
         queue.add(sr);
}
```

```
protected void rechercher() {
        RequestQueue queue = Volley.newRequestQueue(this);
        String url = " http://192.168.10.4 :80/MedSpec/Recherche.php";
        StringRequest sr = new StringRequest(Request.Method.POST, url,
            new Response.Listener<String>() {
              @Override
              public void onResponse(String response) {
        try{
             JSONObject json = new JSONObject(response);
              adpMedecin.clear();
              JSONArray aPH = json.getJSONArray("medecins");
              for (int i = 0; i < aPH.length(); i++) {
                JSONObject o = aPH.getJSONObject(i);
                int id = Integer.parseInt(o.getString("id"));
                String nom = o.getString("nom");
                String prenom = o.getString("prenom");
                String spec = o.getString("spec");
                String adresse = o.getString("adresse");
                String tel = o.getString("tel");
                Medecin med = new Medecin(id, nom, prenom, spec, adresse, tel);
                adpMedecin.add(med);
             }
        } catch (JSONException error) {
                    Toast t = Toast.makeText(Recherche.this, "Problème d'analyse JSON: " +
        error.getMessage(), Toast.LENGTH_LONG);
                    t.show();
                  }
             }
           }, new Response.ErrorListener() {
                @Override
                public void onErrorResponse(VolleyError e) {
                   Toast t = Toast.makeText(Recherche.this, "Problème d'appel HTTP: " +
        e.getMessage(), Toast.LENGTH_LONG);
                t.show();
                }
              }) {
          @Override
          public Map<String, String> getParams() throws AuthFailureError {
            HashMap<String, String> headers = new HashMap<String, String>();
            headers.put("spec", edSpec.getText().toString());
            headers.put("adresse", edAdresse.getText().toString());
            return headers;
          }
          };
          queue.add(sr);
}
```