

The background features a dark gray upper half and a white lower half, separated by a horizontal orange bar. Concentric circles in shades of gray are visible, centered on the left side of the slide.

WORKING WITH WEBSERVICE

HTTP REQUEST

- RESTful và cách làm việc với form HTML
- Tạo một form HTML
- POST – GET
- Phân tích dữ liệu nhận được
- Parse JSON / XML (SAX-DOM)

- RESTful hay RESTful webservice là một chuẩn tương tác giữa client và server trong đó sử dụng các phương thức của Html thay vì sử dụng SOAP envelop như trong giao thức SOAP
- Có thể dễ dàng chuyển đổi giữa Json và XML hoặc cả hai

- Để tạo một tài nguyên trên máy chủ, sử dụng POST.
- Để truy xuất một tài nguyên, sử dụng GET.
- Để thay đổi trạng thái một tài nguyên hoặc để cập nhật nó, sử dụng PUT.
- Để huỷ bỏ hoặc xoá một tài nguyên, sử dụng DELETE.

```
POST /users HTTP/1.1
Host: myserver
Content-Type: application/xml
<?xml version="1.0"?>
<user>
  <name>Robert</name>
</user>
```

```
PUT /users/Robert HTTP/1.1
Host: myserver
Content-Type: application/xml
<?xml version="1.0"?>
<user>
  <name>Bob</name>
</user>
```

```
GET /users/Robert HTTP/1.1
Host: myserver
Accept: application/xml
```

HTML FORM

- Tạo form Html login với GooglePlus thông qua API

- GetCode (post – get)

https://accounts.google.com/o/oauth2/auth?client_id=937282861584-88soevgnhtjjred8oetdq0fd425pomu1@developer.gserviceaccount.com&redirect_uri=http://yapasys.theyapa.com/&scope=https://www.google.com/m8/feeds/&response_type=code

- GetAccessToken (post)

https://accounts.google.com/o/oauth2/token?client_id=937282861584-88soevgnhtjjred8oetdq0fd425pomu1@developer.gserviceaccount.com&redirect_uri=http://yapasys.theyapa.com/&client_secret=LwqKLKg2-lbeGaCHzclZyKvZ&grant_type=authorization_code&code=

- RefreshToken (post)

https://accounts.google.com/o/oauth2/token?client_id=937282861584-88soevgnhtjjred8oetdq0fd425pomu1@developer.gserviceaccount.com&client_secret=LwqKLKg2-lbeGaCHzclZyKvZ&grant_type=refresh_token&refresh_token=

HTTP POST

```
HttpClient client = new DefaultHttpClient();
HttpPost post = new HttpPost(postURL);
if (!args.isEmpty()) {
    for (Map.Entry<String, String> pair : args.entrySet()) {
        params.add(new BasicNameValuePair(pair.getKey(), pair.getValue()));
    }
    UrlEncodedFormEntity ent = new UrlEncodedFormEntity(params, HTTP.UTF_8);
    post.setEntity(ent);
}
HttpResponse responsePOST = client.execute(post);
```

HTTP GET

```
HttpClient client = new DefaultHttpClient();  
HttpGet get = new HttpGet(getURL);  
HttpResponse responsePOST = client.execute(get);
```

- Kết luận :
- HttpPost và HttpGet đóng vai trò tương tự như method Post và Get trong Html form
- Các tham số trong HttpGet truyền thẳng trong getURL
- Các tham số trong HttpPost được truyền dưới dạng FormEntity và thông qua phương thức setEntity(FormEntity ent);

RESULT

- Dữ liệu trả về trong cả hai kiểu Post và Get đều là `HttpResponse`
- Cần phải chuyển dữ liệu về dạng `String` để có thể phân tích (parser)

```
HttpEntity resEntity = responsePOST.getEntity();  
String responString = EntityUtils.toString(resEntity);
```

- Dữ liệu thu về ở đây sẽ ở một trong hai chuẩn JSON hoặc XML

```
JSONObject jsonObject = new JSONObject(responString);
```

```
...
```

```
try{  
    InputStream is = new ByteArrayInputStream(responString.getBytes("UTF-8"));  
    }catch(Exception e){  
}
```


URL CONNECTION

- URL connection
- Status
- Property
- Response Code
-

- Ngoài cách sử dụng `HttpPost`, `HttpGet`, `GetMethod`, `PostMethod` chúng ta có thể sử dụng `URLConnection`, `HttpURLConnection`

```
HttpURLConnection con = (HttpURLConnection) url.openConnection();
```

```
con.setDoOutput(true);
```

```
con.setRequestProperty("Content-Length", ...);
```

```
OutputStream out = con.getOutputStream();
```

```
out.write(...);
```

```
out.close();
```

```
if (con.getResponseCode() != 200)
```

- URLconnection cho phép tác động trực tiếp vào các request property
- Ví dụ

```
URLConnection connection = new URL(url).openConnection();  
connection.setDoOutput(true); // Triggers POST.  
connection.setRequestMethod("PUT");  
connection.setRequestProperty("Accept-Charset", charset);  
connection.setRequestProperty("Range", charset);  
connection.setRequestProperty("Content-Type", "application/x-www-form-  
urlencoded; charset=" + charset);
```