Lecture 7 OverView

- Day6/Lab6 Questions
- Review Student Videos
- Todays Topic (**API,SERVICE**)
- Todays Lab (Read XML/JSON and store it's contents)

Class interactivity

Please come up to the board and create groups.

- Each group is asked to design a site (tweeter, blogger, facebook ...)
- Design a flow
- Separate Views, Models, Controllers

API

What is API?

- API is a set of routines, protocols, and tools for building software applications.
- A good API makes it easier to develop a program by providing all the building blocks.
- A programmer then puts the blocks together and you can use all of the blocks or some of the blocks.

API

API FOR YOUR DESIGN?

- What can these sites on the board share?
- API should return a query or individual data.
- Remember AJAX?
- This time the parsing of data happening on the server side.

COLDFUSION CODE

LET's create and JSON/XML data with Coldfusion (day7.cfm)

```
<cfset me={
        "fname" = "orcun",
        "height"=64,
        "dt"=createDate(2003,07,07),
        "courses"=["ssl","bdf","sdi"]
}>

<cfdump var="#me#"/>
```

What do we see when we run the code?

COLDFUSION CODE

- We can serialize and de-serialize this data (Just like SFW1 Ajax Serialize)
- We have built-in methods in our SS languages for **JSON**

FOR Coldfusion we use **serializeJSON()** to create JSON Object.

```
<!--- CREATING JSON WITH COLDFUSION --->

<cfset myob=serializeJSON(myob)>
<cfset views.getView("../views/pages/content",myob)>
```

COLDFUSION CODE

JSON OUTPUT ON THE PAGE FOR AJAX

```
1  <cfcontent type="application/json" reset="yes">
2
3    <cfoutput>#data#</cfoutput>
4
5  </cfcontent>
6
```

When we load the page we come up with JSON Object

{"fname":"orcun"}

Your content code should always follow

<cfexit method="exittemplate">

COLDFUSION CODE

- We can create the XML version of our data for service or API uses also
- We have built-in methods in our SS languages for **XML**

FOR Coldfusion we use **cfml2ddx action** to create XML file.

```
<cfwddx action="cfml2wddx" input="#myob#" output="toxml">

<cfset views.getView("../views/pages/content",toxml)>
```

We can also output the file for a hard copy by adding this line after the cfwddx tag

```
<cffile action="write" file="/Applications/Railo/webroot/SSL/mycf.xml" output="toxml">
```

COLDFUSION CODE

To output the XML file with Coldfusion.

```
<cfcontent type="text/xml" reset="yes">
<cfoutput>#data#</cfoutput>
<cfexit method="exittemplate">
```

COLDFUSION CODE

To parse an XML file with Coldfusion:

```
<cfset x=xmlParse("#data#")>
<cfset b = x.xmlRoot.data.struct.var>
<cfdump var="#b#">
```

And to search for a node or loop through nodes:

```
<cfset h= xmlSearch(x, "//data/struct/var")>

<cfdump var="#h#">

<cfloop  from="1" to="1" index="i">

<cfoutput>#h[i]#</cfoutput>
</cfloop>
```

COLDFUSION CODE

To load external XML feed with Coldfusion:

```
<cffile action="read"
file="#expandPath('http://rss.cnn.com/rss/cnn_topstories.rss')#"
variable="data">
<cfdump var="#data#"/>
```

PHP CODE

LET's create and JSON/XML data with PHP (day7)

```
$r = array("name"=>array("name"=>"orcun"));
header("Content-Type: application/json");
echo json_encode($r);
?>
```

What do we see when we run the code?

PHP CODE

LET's create and XML data with PHP (day7)

Creating XML is little tricky with php but once you see the syntax it gets easier.

```php
header("Content-Type: text/xml");

$xmlDoc = new DOMDocument();

$root    = $xmlDoc->appendChild($xmlDoc->createElement("posts"));
$post    = $root->appendChild(
                $xmlDoc->createElement("post","sdfsdfs"))->appendChild(
                $xmlDoc->createAttribute("title"))->appendChild(
                $xmlDoc->createTextNode("3"));


$xmlDoc->formatOutput = true;

echo $xmlDoc->saveXML();
```

PHP CODE

LET's create and XML data with PHP (day7)

To output the XML you just created:

```
$xmlDoc->save("my.xml");
```

PHP CODE

LET's read the XML data with PHP

```
$xmlStr = file_get_contents("http://host/feed.rss");
$xml = new SimpleXMLElement($xmlStr);
$res = $xml->xpath("/field[@name='name']/string");

foreach($res as $r){
    echo $r[0]."<br>";

}
```

Lecture 7 Lab

Today's Lab XML parsing:

- Find a web service for XML or JSON.
- Get the data
- Store the data into a database by using the techniques we have covered today.
- Time stamp the stored information so that data would not get overwritten if there is no change.