Dept of Computer Science
Ahmadu Bello University

WEB APPLICATION ENGINEERING II

Lecture #4

Umar Ibrahim Enesi

Dept of Computer Science
Ahmadu Bello University

Objectives

- Gain understanding on:
  - Form structure
  - Form Handling
  - Form Validation with Filters and Pattern matching
  - Redirection
  - Sticky form

Forms: Review

- According to W3C specification:

A form (web form) is a component of a Web page that has named controls (input elements) which allows users to provide data that will be sent to the server for further processing.

- HTML form element represents the section that contains interactive controls that allows users to provide data.
- `<form>` and `</form>` tags are used to enclose such interactive controls.
- Interactive form controls may be text fields, password field, buttons, checkboxes, selectors, range controls, or colour pickers.

Forms: Review

- You can specify how a form is submitted using the *`method`* ***attribute*** on form element. Common values are:

```
GET (default)
POST
```

- You also have to specify the URL of the service that will handle the submitted data, using the *`action`* ***attribute*** (otherwise self referencing)
- To specify how form data is encoded, use the *`enctype`* ***attribute.*** Common values are:

```
application/x-www-form-urlencoded (default)
multipart/form-data
```

Dept of Computer Science
Ahmadu Bello University

Forms: Review

- Each form control is assigned a name using a `name` ***attribute***
- Initial values of form controls can be assigned using the `value` *attribute* (with the exception of `textarea`).
- When a form is submitted *name/value* pairs of all form controls are made available to the handler script.

Dept of Computer Science
Ahmadu Bello University

Handling Forms

- Information from forms are exposed to handler scripts via various *predefined superglobal* array variables;
  - `$_GET`
  - `$_POST`
  - `$_REQUEST`
  - `$_FILES`
- Each array key is mapped to a corresponding name of a form control.
- While the value of each key is mapped to the value of the corresponding named form control.

Dept of Computer Science
Ahmadu Bello University

Handling Forms

- It is possible for more than one form component to be assigned the same name suffixed with square bracket.

```
Mobile No1: <input type="text" name="mobilenos[]" />
Mobile No2: <input type="text" name="mobilenos[]" />
Mobile No3: <input type="text" name="mobilenos[]" />
```

- In such case, value from components that share the same name will be put together and made available in an array.

```
<?php
        //$mobilenos will be an array of all values submitted with mobilenos
        $mobilenos = $_REQUEST['mobilenos'];
?>
```

Dept of Computer Science
Ahmadu Bello University

Form Validation

- Form validation ensures that web applications are robust against all forms of input data.
- Form validation prevents malicious users from taking advantage of web forms to perform an attack.
- Form validation also ensures that valid data is supplied by users.
  - supplying a negative value as age.
  - omitting the username during login.
  - supplying an invalid email address or mobile number.
- Certain inconsistencies in user input can also be detected:
  - selecting a date in the future as date of birth.

Client side vs Server side form validation

- Client Side form validation:
  - Real time (or instant) validation.
  - Faster.
  - Improves user experience (interactivity).
  - May be bypassed.
  - Technology behind validation (*JavaScript*) may not be supported by the client.
- Server Side form validation:
  - More reliable
  - Independent of client functionality
  - Less convenient to users.(no real time validation)
  - Slower than client side validation

Basic PHP Form Validation

- Various string and variable handling functions can be used to perform basic check on the validity and consistency of external data.
- Many of these functions can also be used to sanitize data.

```
isset(), trim(), rtrim(), ltrim(), is_*(),
addslashes()
```

Dept of Computer Science
Ahmadu Bello University

PHP Form Validation: Filters

- Filter is a PHP’s core extension.
- Filter is used for validation or sanitization of data.
- A full description of all supported filters can be found in PHP documentation manual.
- Flags are also used to customize the behaviour of filters.
- Filter extension provides various functions for validation and sanitization.

Dept of Computer Science
Ahmadu Bello University

PHP Form Validation: Filters

List of supported filters:

Validate Filters	Sanitize Filters
FILTER_VALIDATE_BOOLEAN	FILTER_SANITIZE_EMAIL
FILTER_VALIDATE_EMAIL	FILTER_SANITIZE_ENCODED
FILTER_VALIDATE_FLOAT	FILTER_SANITIZE_MAGIC_QUOTES
FILTER_VALIDATE_INT	FILTER_SANITIZE_NUMBER_FLOAT
FILTER_VALIDATE_IP	FILTER_SANITIZE_NUMBER_INT
FILTER_VALIDATE_MAC	FILTER_SANITIZE_SPECIAL_CHARS
FILTER_VALIDATE_REGEXP	FILTER_SANITIZE_FULL_SPECIAL_CHARS
FILTER_VALIDATE_URL	

Dept of Computer Science
Ahmadu Bello University

PHP Form Validation: Filters

List of supported flags:

Supported Flags	
FILTER_FLAG_STRIP_LOW	FILTER_FLAG_ENCODE_AMP
FILTER_FLAG_STRIP_HIGH	FILTER_NULL_ON_FAILURE
FILTER_FLAG_ALLOW_FRACTION	FILTER_FLAG_ALLOW_OCTAL
FILTER_FLAG_ALLOW_THOUSAND	FILTER_FLAG_ALLOW_HEX
FILTER_FLAG_ALLOW_SCIENTIFIC	FILTER_FLAG_IPV4
FILTER_FLAG_NO_ENCODE_QUOTES	FILTER_FLAG_IPV6
FILTER_FLAG_ENCODE_LOW	FILTER_FLAG_NO_PRIV_RANGE
FILTER_FLAG_ENCODE_HIGH	FILTER_FLAG_NO_RES_RANGE
FILTER_FLAG_PATH_REQUIRED	FILTER_FLAG_QUERY_REQUIRED

Dept of Computer Science
Ahmadu Bello University

PHP Form Validation: Filters

List of available filter functions:

- `filter_has_var()`
- `filter_id()`
- `filter_input_array()`
- `filter_input()`
- `filter_list()`
- `filter_var_array()`
- `filter_var()`

PHP Form Validation: filter_var()

The most used functions are `filter_var()` and `filter_var_array()`

Usage:

```
filter_var($var, $filter, $options)
filter_var_array($data, $definition, $options)
```

Pattern Matching with Regular Expressions

- It is possible to match a pattern against values of variables.
- Such pattern is called regular expression.
- A match is passed if the presence of the pattern can be found in the sequence of tokens (value).
- Otherwise failed.
- Pattern matching is performed from left to right.
- Useful in data validation.
- Pattern matching can be performed in PHP with support of PCRE extension.

PCRE Syntax: Delimiter

- PCRE requires that all pattern be delimited with non-alphanumeric character. (backslash and white space not accepted)

```
~pattern~
#anotherpattern#
/yetanotherpattern/
```

- If delimiter is literarily part of the pattern, it must be escaped with backlash.

  #valid\#pattern#

- Pattern modifiers (*i, x, m,…*) may be added after the ending delimiter.

```
~pattern~i
```

PCRE Syntax: Meta-Characters

- Some characters have special meaning in regular expressions.

Meta-Character	Outside "[" and "]"	In-between "[" and "]"
\	General escape character	General escape character
^	Assert start of subject/line. True only if the matching point is at the start of the subject	Negate the class, but only if the first character
$	Assert end of subject. True only if the matching point is the last character of the subject.	
.	Match any char. except newline	
[ ]	Start & end class definition resp.	
|	Alternative branch (similar to OR)	
( )	Start & end sub-pattern resp.	

Dept of Computer Science
Ahmadu Bello University

PCRE Syntax: Meta-Characters

Meta-Character	**Outside "[" and "]"**	**In-between "[" and "]"**
?	0 or 1 occurrence	
+	1 or more occurrence	
*	0 or more occurrence	
{ }	Start & end of min/max quantifier	
-		Indicate character range

Dept of Computer Science
Ahmadu Bello University

PCRE Syntax: Character class

- "[" introduces a character class and terminated by "]".
- A character class defines a set of characters as one entity.
- A character class matches a single character in the subject.
- To match multiple characters use quantifiers

Dept of Computer Science
Ahmadu Bello University

PCRE Syntax: Repetition

- Use quantifiers to specify repetition
- Quantifiers can be placed after a single character, character class, sub-pattern or "." meta-character.
- Generally repetition can be specified thus:

`{a,b}`

Note:

- `a` must be less than or equal to `b`
- Both `a` and `b` must be less than 65536.
- `b` can be omitted. (in which case, there is not upper limit)
- Both `,` and `b` can be omitted (in which case, exact number of match is specified by `a`)

Dept of Computer Science
Ahmadu Bello University

PCRE Syntax: Repetition

- For convenience:

?	{0,1}
+	{1,}
*	{0,}

Dept of Computer Science
Ahmadu Bello University

PCRE Syntax: Sub-Patterns

- Sub-patterns are delimited by parenthesis.
- It is useful when demarcating a set of alternatives.
- Alternative patterns can be declared by vertical bar (|)

  `(boy|girl)` matches boy or girl

  `(cosc|math|stat)204` matches cosc204 or math204 or stat204

Dept of Computer Science
Ahmadu Bello University

PCRE Syntax: Pattern Modifiers

- Pattern modifiers are used to alter the default behaviour of search

Pattern Modifier	Effect
i	Case insensitive search
m	Matches multiple line
s	Dot (.) metacharacter matches newline
x	Ignore whitespace characters

Dept of Computer Science
Ahmadu Bello University

PCRE Syntax: Examples

Four digit year	[1-9][0-9]{3}
Age	[1-9][0-9]*
Any character	.
Non negative numbers	(0)|(+)?[1-9][0-9]{0,}
Decimal Number	[+-]?([0-9]*[\.] [0-9]+) | ([0-9]+[\.][0-9]*)

Dept of Computer Science
Ahmadu Bello University

PCRE Syntax: PCRE Functions

- preg_filter— Perform a regular expression search and replace
- preg_grep— Return array entries that match the pattern
- preg_last_error — Returns the error code of the last PCRE regex execution
- preg_match_all — Perform a global regular expression match
- preg_match — Perform a regular expression match
- preg_quote — Quote regular expression characters
- preg_replace_callback_array — Perform a regular expression search and replace using callbacks
- preg_replace_callback — Perform a regular expression search and replace using a callback
- preg_replace — Perform a regular expression search and replace
- preg_split — Split string by a regular expression

preg_match()

In its simplest form:

```
    preg_match($pattern, $subject)
```

`$pattern` must be a string containing a PCRE.

`$subject` is a string to match the pattern against.

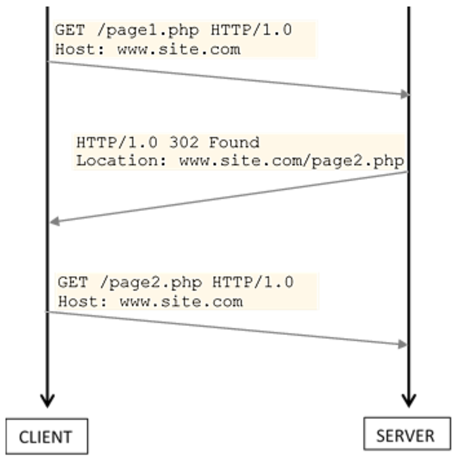
`preg_match()` returns 1 if a match occurs and 0 otherwise. *False* may be returned if an error occurred.

Redirection

- Depending on the situation, a redirection may be required in other to complete a process.
- During a redirect, the server responds with status code of 302 (or equivalent) and will additionally provide a URL in the `location` header field.
- The client (browser) on the other hand makes a second request to the specified URL.

Dept of Computer Science
Ahmadu Bello University

Redirection



Dept of Computer Science
Ahmadu Bello University

Redirection

- To cause the server to respond with status code 302, use:

```
header(“location:  new_location_url”);
```

- It is advisable to always exit script execution immediately after calling on the `header()` function.

```
header(“location:  new_location_url”);
exit();
```

Sticky Form

- Normally, when a form fails validation test, the page containing the form is displayed again for the user to have another chance to refill and re-submit the form with appropriate entries.
- The form is cleared off its previous entries and the user will have to fill in the form afresh. This can be frustrating and daunting especially for large forms.
- It is possible to retain previous entries after a form is reloaded.
- Such a form is called sticky form.

Sticky Form

- There are various techniques for implementing a sticky form.
  - Sessions
  - Cookies
  - URL parameters
  - Variables (self-referencing forms)
- In most cases, use `isset()` method to find out if value has been submitted previously.

```
<input type ="text" name="usn"
<?php echo ((isset($_POST['usn'])?("value='$_POST["usn"]'"):("")); ?> />
```

Key Points

- When a form is submitted, name/value pairs of each form component is made available through PHP's super global array.
- The method of form submission (GET or POST) determines which super global array holds the submitted data.
- It is always necessary to validate external data.
- Server side validation is more reliable but less convenient over client side validation.
- PHP Filter and PCRE extensions can be used for input validation.
- Sticky forms may be used to improve users' experience.

References

- PHP Documentation Manual
- Murach PHP and MySQL
- http://www.phpro.org/tutorials/Filtering-Data-with-PHP.html
- http://webcheatsheet.com/php/regular_expressions.php
- http://www.regextester.com/pregsyntax.html
- http://thenewcode.com/199/Pattern-Matching-in-PHP
- http://www.tutorialcode.com/php/sticky-forms/