Internal Assessment Test 3 – July 2022

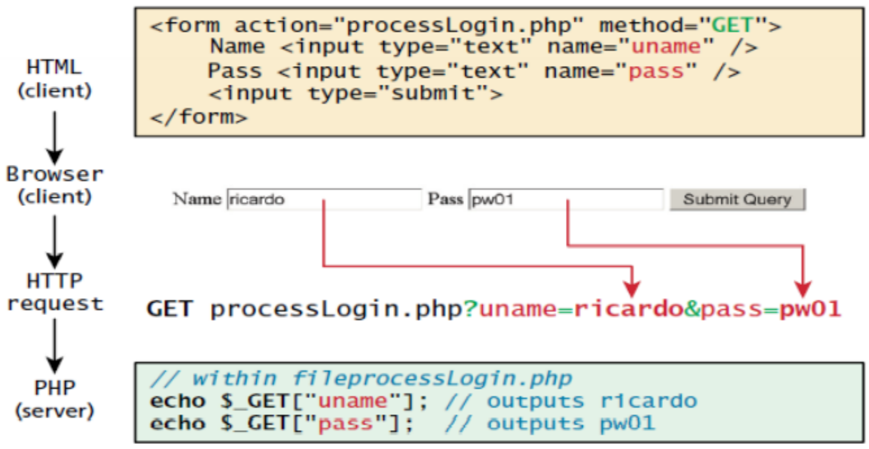
Solution

Sub:	Web Technology & its Applications					Sub Code:	18CS63		Branch:	CSE
Date:	09/07/2022	Duration:	90 mins	Max Marks:	50	Sem/Sec:	A, B & C	Time	8.30 - 10.00 am	OBE

	Answer any FIVE FULL Questions	MARKS	CO	RBT
1	Explain $_GET and $_POST SUPER GLOBAl/HYPER GLOBAL arrays with flow diagram	[10]	CO2	L2

Ans:

- The $_GET and $_POST arrays are the most important super global variables in PHP since they allow the programmer to access data sent by the client in a query string.



```
<form action="processLogin.php" method="GET">
    Name <input type="text" name="uname" />
    Pass <input type="text" name="pass" />
    <input type="submit">
</form>
```

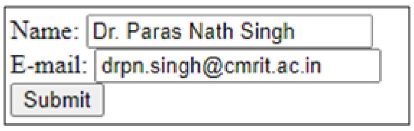
```
GET processLogin.php?uname=ricardo&pass=pw01
```

```
// within fileprocessLogin.php
echo $_GET["uname"]; // outputs ricardo
echo $_GET["pass"];  // outputs pw01
```

- An HTML form (or an HTML link) allows a client to send data to the server.
- That data is formatted such that each value is associated with a name defined in the form.
- If the form was submitted using an HTTP GET request, then the resulting URL will contain the data in the query string.
- PHP will populate the superglobal $_GET array using the contents of this query string in the URL.

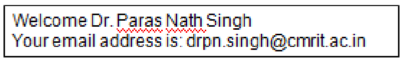
Flow from HTML to PHP

```
<html>
<body>
<form action="welcome.php" method="post">
Name: <input type="text" name="name"><br>
E-mail: <input type="text" name="email"><br>
<input type="submit">
</form>
</body>
</html>
```

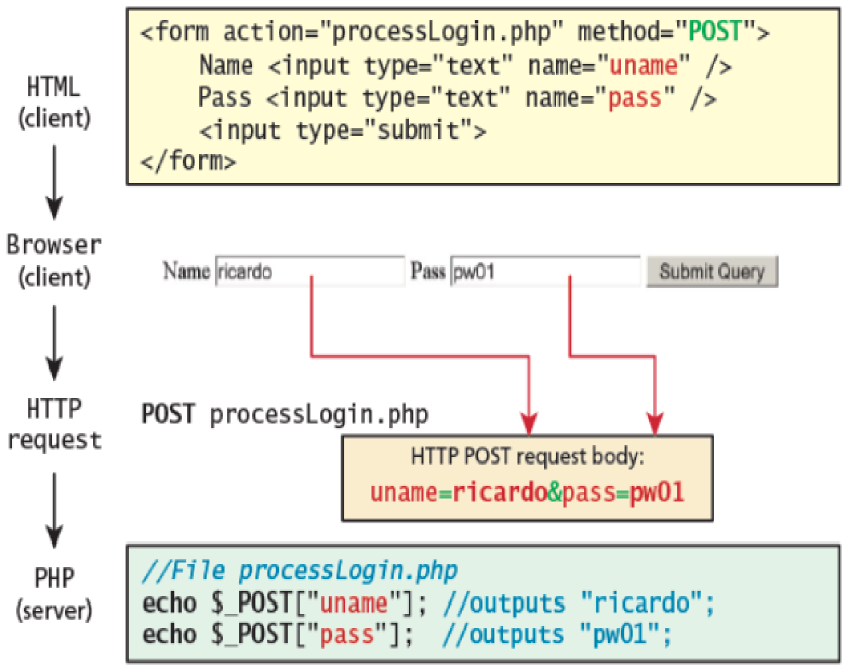


Name: Dr. Paras Nath Singh
E-mail: drpn.singh@cmrit.ac.in
Submit

```
//Welcome.php
<html>
<body>
Welcome <?php echo $_POST["name"]; ?><br>
Your email address is: <?php echo $_POST["email"]; ?>
</body>
</html>
```



Welcome Dr. Paras Nath Singh
Your email address is: drpn.singh@cmrit.ac.in



```
<form action="processLogin.php" method="POST">
    Name <input type="text" name="uname" />
    Pass <input type="text" name="pass" />
    <input type="submit">
</form>
```

```
POST processLogin.php
uname=ricardo&pass=pw01
```

```
//File processLogin.php
echo $_POST["uname"]; //outputs "ricardo";
echo $_POST["pass"];  //outputs "pw01";
```

2 Explain procedural error handling and object oriented exception handling with suitable code and segment. [05] CO2 L2

Ans:

- When a fatal PHP error occurs, program execution will eventually terminate unless it is handled.
- The PHP documentation provides two mechanisms for handling runtime errors:
- procedural error handling
- object-oriented exception handling.

Procedural Error handling:

- In the procedural approach to error handling, the programmer needs to explicitly test for error conditions after performing a task that might generate an error.
- While this approach might seem more straightforward, it does require the programmer to know ahead of time what code is going to generate an error condition.

```
$connection = mysqli_connect(DBHOST, DBUSER, DBPASS, DBNAME);

$error = mysqli_connect_error();
if ($error != null) {
   // handle the error
   ...
}
```

Object Oriented Error handling:

- When a runtime error occurs, PHP throws an exception.
- This exception can be caught and handled either by the function, class, or page that generated the exception or by the code that called the function or class.
- If an exception is not caught, then eventually the PHP environment will handle it by terminating execution with an “Uncaught Exception” message.

PHP uses the try .. .catch programming construct to programmatically deal with exceptions at runtime.

```
// Exception throwing function
function throwException($message = null,$code = null) {
  throw new Exception($message,$code);
}

try {
  // PHP code here
  $connection = mysqli_connect(DBHOST, DBUSER, DBPASS, DBNAME)
    or throwException("error");
 //...
}
catch (Exception $e) {
  echo ' Caught exception: ' .  $e->getMessage();
  echo ' On Line : ' .  $e->getLine();
  echo ' Stack Trace: '; print_r($e->getTrace());
} finally {
  // PHP code here that will be executed after try or after catch
}
```

```
function processArray($array)
{
   // make sure the passed parameter is an array with values
   if ( empty($array) ) {
      throw new Exception('Array with values expected');
   }
   // process the array code
   ...
}
```

```
public function setBirthDate($birthdate){
   // set variable only if passed a valid date string
   if ( $timestamp = strtotime($birthdate) ) {
      $this->birthDate=$timestamp;
   }
   else {
     throw new Exception("Invalid Date in Artist->setBirthDate()");
   }
}
```

```
try {
  // PHP code here
}
catch (Exception $e) {
     // do some application-specific exception handling here
     ...
     // now rethrow exception
     throw $e;
}
```

3	Explain classes, objects and properties with an example program in PHP.	[05]	CO2	L2

Ans:

Defining Classes

- The PHP syntax for defining a class uses the class keyword followed by the class name and { } braces.
- The properties and methods of the class are defined within the braces.

```
class Artist
{
public $firstName;
public $lastName;
public $birthDate;
public $birthCity;
public $deathDate;
}
```

- Each property in the class is declared using one of the keywords public, protected, or private followed by the property or variable name.

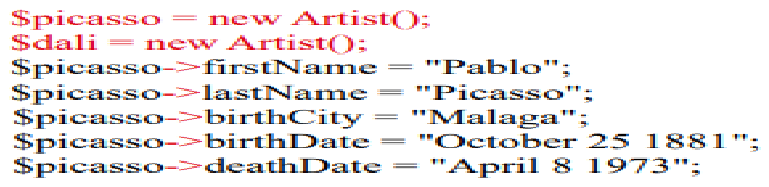
Instantiating Objects

- It’s important to note that defining a class is not the same as using it.
- To make use of a class, one must instantiate (create) objects from its definition using the new keyword.

```
$picasso = new Artist();
$dali = new Artist();
```

Properties

Once you have instances of an object, you can access and modify the properties of each one separately using the variable name and an arrow (->), which is constructed from the dash and greater than symbols.



```
$picasso = new Artist();
$dali = new Artist();
$picasso->firstName = "Pablo";
$picasso->lastName = "Picasso";
$picasso->birthCity = "Malaga";
$picasso->birthDate = "October 25 1881";
$picasso->deathDate = "April 8 1973";
```

4 How Query Strings are used in Hyperlinks? How Query string is sanitized? Write with example and diagram. [04] CO4 L2

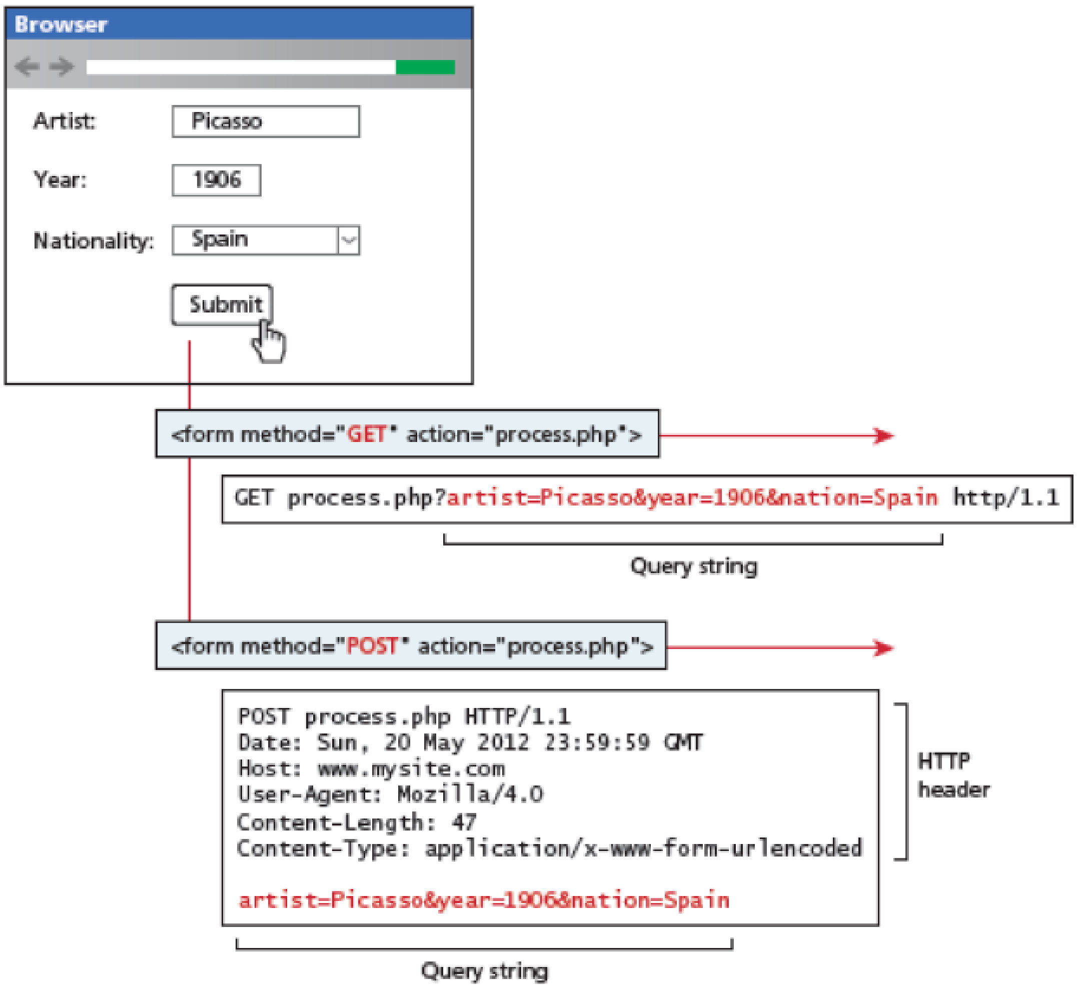
Ans:

To pass information is constrainedby the basic request-response interaction of the HTTP protocol. In HTTP,we can pass information using:

- Query strings
- Cookies

Passing Information via Query Strings

a web page can pass query string information
from the browser to the server using one of the two methods: a query string withinthe URL (GET) and a query string within the HTTP header (POST). Figure reviews these two different approaches.



```
<form method="GET" action="process.php">

GET process.php?artist=Picasso&year=1906&nation=Spain http/1.1

<form method="POST" action="process.php">

POST process.php HTTP/1.1
Date: Sun, 20 May 2012 23:59:59 GMT
Host: www.mysite.com
User-Agent: Mozilla/4.0
Content-Length: 47
Content-Type: application/x-www-form-urlencoded

artist=Picasso&year=1906&nation=Spain
```

<u>To Sanitize Query String</u>

jQuery escape html string:

```
function escapeHtml(str) {
   return str.replace(/&/g, "&amp;").replace(/</g, "&lt;").replace(/>/g, "&gt;").replace(/"/g, "&quot;").replace(/'/g, "&#039;");
}

//escaping HTML with jquery
var dangerousHTML = "<script>alert('Badabing Baby!');</script>";
$("#myElementID").text(dangerousHTML); //.text() function will escape and display text
```

jQuery escape html string:

```
//Alternatively, here is plain Javascript escape function
function escapeHtml(str) {
   return str.replace(/&/g, "&").replace(/</g, "<").replace(/>/g, ">").replace(/"/g, """").replace(/'/g, "'");
}
```

5	Define AJAX. Explain AJAX request by writing UML diagram.	[05]	CO3	L2

Ans:

- Asynchronous JavaScript with XML (AJAX) is a term used to describe a paradigm that allows a web browser to send messages back to the server without interrupting the flow of what's being shown in the browser.
- This makes use of a browser's multi-threaded design and lets one thread handle the browser and interactions while other threads wait for responses

to asynchronous requests.

Making Asynchronous request

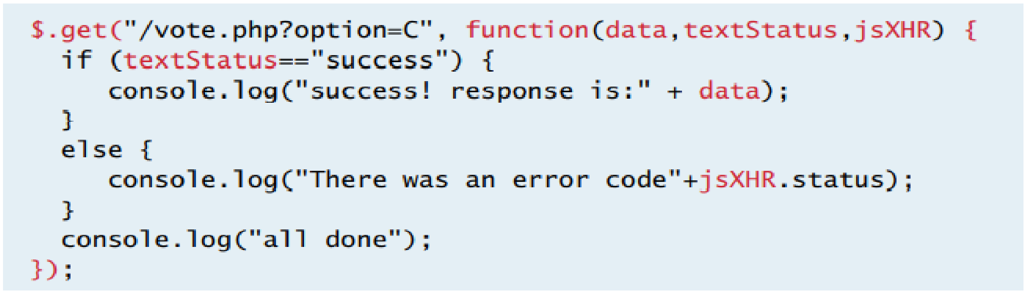
- jQuery provides a family of methods to make asynchronous requests.
- Consider for instance the very simple server time page described above.
- If the URL currentTime.php returns a single string and you want to load that value asynchronously into the element, you could write:
  $("#timeDiv").load("currentTime.php");

GET Requests

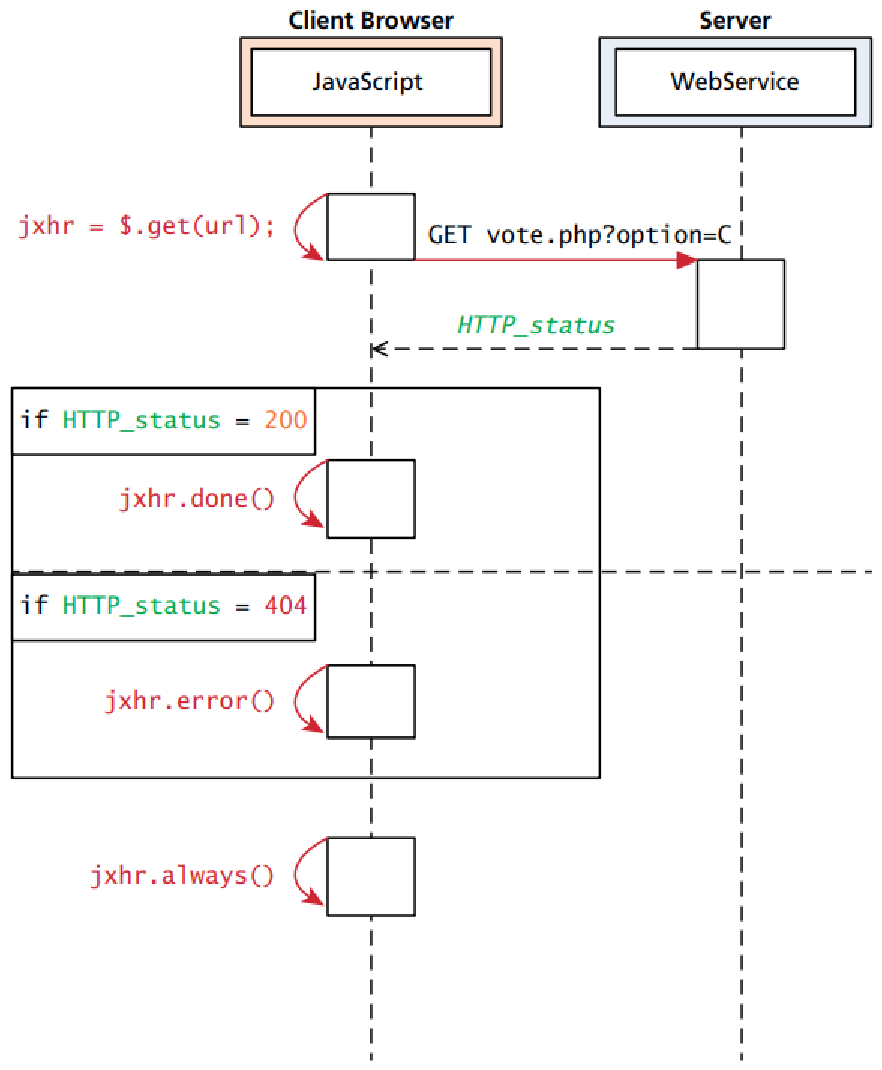
- get() method can request a resource very easily, handling the response from the request requires that we revisit the notion of the handler and listener.
- The formal definition of the get() method lists one required parameter url and three optional ones: data, a callback to a success() method, and a dataType.

jQuery.get(url [, data] [, success(data, textStatus, jqXHR)] [, dataType])

■ url is a string **that holds the location to send the request**.
■ data is an optional parameter that is a query string or a Plain Object.
■ success(data,textStatus,jqXHR) is an optional callback function that executes when the response is received. Callbacks are the programming term given to placeholders for functions so that a function can be passed into another function and then called from there (called back). This callback function can take three optional parameters
° data holding the body of the response as a string.
° textStatus holding the status of the request (i.e., "success").
° jqXHR holding a jqXHR object, described shortly.
■ dataType is an optional parameter to hold the type of data expected from the server. By default jQuery makes an intelligent guess between xml, json, script, or html.



```
$.get("/vote.php?option=C", function(data,textStatus,jsXHR) {
  if (textStatus=="success") {
     console.log("success! response is:" + data);
  }
  else {
     console.log("There was an error code"+jsXHR.status);
  }
  console.log("all done");
});
```

	 Client Browser Server JavaScript WebService jxhr = $.get(url); GET vote.php?option=C HTTP_status if HTTP_status = 200 jxhr.done() if HTTP_status = 404 jxhr.error() jxhr.always() FIGURE 15.12 Sequence diagram depicting how the jqXHR object reacts to different response codes			
6	With suitable code segments, explain converting a JSON object in JavaScript and a PHP object in PHP.	[10]	CO4	L2

Ans:

Using JSON object in Java Script:

- Since the syntax of JSON is the same used for creating objects in JavaScript, it is easy to make use of the JSON format in JavaScript:
- the JSON information will be contained within a string, and the JSON.parse() function can be used to transform the string containing the JSON data into a JavaScript object.

```
var text = '{"artist": {"name":"Manet","nationality":"France"}}';
var a = JSON.parse(text);
alert(a.artist.nationality);
```

The jQuery library also provides a JSON parser that will work with all browsers (the `JSON.parse()` function is not available on older browsers):

```
var artist = jQuery.parseJSON(text);
```

JavaScript also provides a mechanism to translate a JavaScript object into a JSON string:

```
var text = JSON.stringify(artist);
```

Using JSON in PHP

- PHP comes with a JSON extension and as of version 5.2 of PHP, the JSON extension is bundled and compiled into PHP by default.7 Converting a JSON string into a PHP object is quite straightforward:

```
<?php
  // convert JSON string into PHP object
  $text = '{"artist": {"name":"Manet","nationality":"France"}}';
  $anObject = json_decode($text);
  echo $anObject->artist->nationality;

  // convert JSON string into PHP associative array
  $anArray = json_decode($text, true);
  echo $anArray['artist']['nationality'];
?>
```

- the json_decode() function can return either a PHP object or an associative array. Since JSON data is often coming from an external source, one should always check for parse errors before using it, which can be done via the json_last_error() function:

```
<?php
  // convert JSON string into PHP object
  $text = '{"artist": {"name":"Manet","nationality":"France"}}';
  $anObject = json_decode($text);
  // check for parse errors
  if (json_last_error() == JSON_ERROR_NONE) {
    echo $anObject->artist->nationality;
  }
?>
```

- To go the other direction (i.e., to convert a PHP object into a JSON string), you can use the json_encode() function. // convert PHP object into a JSON string $text = json_encode($anObject);

CO PO Mapping

Course Outcomes		Modules covered	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3	PSO4
CO1	Apply appropriate technologies to create an interactive dynamic webpage	1,2,3,4, 5	2	2	2	-	3	2	-	-	2	-	2	2	3	-	-	2
CO2	Summarize advanced dynamic web projects using client-Server technologies.	1,2,3,4, 5	2	2	2	-	3	2	-	-	2	-	2	2	3	-	-	2
CO3	Demonstrate ability to adapt to changing web development and design skills.	1,2,3,4, 5	2	2	2	-	3	2	-	-	2	-	2	2	3	-	-	2
CO4	Analyze and develop a client server application using appropriate technologies considering performance.	1,2,3,4, 5	2	2	2	-	3	2	-	-	2	-	2	2	3	-	-	2
CO5	Inspect JavaScript frameworks like jQuery and Backbone which facilitates developer to focus on core features	5	2	2	2	-	3	2	-	-	2	-	2	2	3	-	-	2

COGNITIVE LEVEL	REVISED BLOOMS TAXONOMY KEYWORDS
L1	List, define, tell, describe, identify, show, label, collect, examine, tabulate, quote, name, who, when, where, etc.
L2	summarize, describe, interpret, contrast, predict, associate, distinguish, estimate, differentiate, discuss, extend

L3	Apply, demonstrate, calculate, complete, illustrate, show, solve, examine, modify, relate, change, classify, experiment, discover.
L4	Analyze, separate, order, explain, connect, classify, arrange, divide, compare, select, explain, infer.
L5	Assess, decide, rank, grade, test, measure, recommend, convince, select, judge, explain, discriminate, support, conclude, compare, summarize.

PROGRAM OUTCOMES (PO), PROGRAM SPECIFIC OUTCOMES (PSO)				CORRELATION LEVELS	
PO1	Engineering knowledge	PO7	Environment and sustainability	0	No Correlation
PO2	Problem analysis	PO8	Ethics	1	Slight/Low
PO3	Design/development of solutions	PO9	Individual and team work	2	Moderate/ Medium
PO4	Conduct investigations of complex problems	PO10	Communication	3	Substantial/ High
PO5	Modern tool usage	PO11	Project management and finance		
PO6	The Engineer and society	PO12	Life-long learning		
PSO1	Develop applications using different stacks of web and programming technologies				
PSO2	Design and develop secure, parallel, distributed, networked, and digital systems				
PSO3	Apply software engineering methods to design, develop, test and manage software systems.				
PSO4	Develop intelligent applications for business and industry				