



# American International University- Bangladesh

## CSC 3215: Web Technologies

### CO2 and CO4 Evaluation Finished Project Report Spring 2019-2020

**Project Title:** Content Rating Site  
**Section: I**

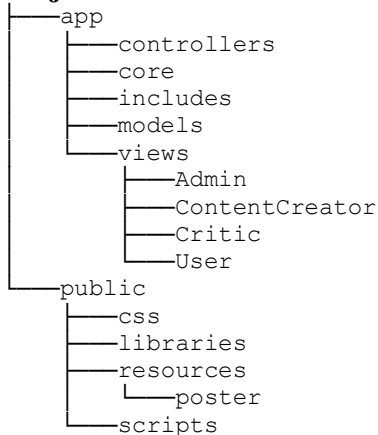
Student Name	Student Id
Hasidul Islam	17-33834-1
Sadia Farzana	17-33890-1
Nafis Mustafa	17-33855-1

# Content Rating Site

## Project Description:

Content rating site is a website where content creator can upload their content's profile to be rated. Normal user's and critics rates the content. Normal users and critics can also comment on the contents. One user or one critic can only comment and rate once on a content. That saves the system from being manipulated by anyone. The rating values are mean of the individual rating. User rating and comments reflects how much normal people like the content. Critic rating shows how critics are feeling about the content. There is an admin in this system who observe everything. The Admin can see every user and every content. Admin has the capability to delete content, user, critic and content creator. While doing this project MVC and OOP concepts were followed.

## Project Folder Structure:



## Core Modules of the Project:

### LoginModel.php

```
1 <?php
2
3 require_once INCLUDES . "DataAccess.php";
4 class LoginModel
5 {
6     public $id;
7     public $password;
8     public $permissionType;
9
10
11     function setId($id)
12     {
13         $this->id = $id;
14     }
15     function setPassword($password)
16     {
17         $this->password = $password;
18     }
19     function setLoginInfo($id, $password, $permissionType)
20     {
21         $this->id = $id;
22         $this->password = $password;
23         $this->permissionType = $permissionType;
24     }
25
26
27
```

# Content Rating Site

```
28     function insertUser($id, $password, $permission)
29     {
30
31         try {
32             //this function will be needing to insert user in login table
33             $sql = "insert into login (id,password,permission) values ('" . $id .
34 "' , '" . $password . "' , '" . $permission . "')";
35             var_dump($sql);
36             $db = new DataAccess();
37
38             $db->executeQuery($sql);
39         } catch (Exception $e) {
40             echo 'Caught exception: ', $e->getMessage(), "\n";
41         }
42     }
43
44
45     public function getLatestAutoId() //this will get latest auto id of login
46 table
47     {
48         $sql = "SELECT MAX(autoId) AS autoId FROM login";
49         $db = new DataAccess();
50         $result = $db->getData($sql);
51
52
53         while ($row = $result->fetch_assoc()) {
54             $data = $row['autoId'];
55         }
56         return $data;
57     }
58
59
60     function validateLogin($id, $password)
61     {
62         try {
63
64             //echo $id . " " . $password;
65             //this method will find if the user exists or not
66             $sql = "SELECT * FROM login WHERE id='" . $id . "' AND password='" .
67 $password . "'";
68             $db = new DataAccess();
69             $result = $db->getData($sql);
70
71             //these are for testing
72             //while ($row = mysqli_fetch_assoc($result)) {
73             // echo "id: " . $row["id"] . " - Pass: " . $row["password"] . " " .
74 $row["permission"] . "<br>";
75             //}
76
77
78             echo "I am in validateLogin";
79
80             if ($result->num_rows > 0) {
81                 echo "I am in row 1 er beshi";
82
83                 return true;
84             } else {
85                 return false;
```

# Content Rating Site

```
86     }
87     } catch (Exception $e) {
88         echo 'Caught exception: ', $e->getMessage(), "\n";
89     }
90 }
91
92 function getPermissionType($id, $password)
93 {
94     try {
95         //this method will find if the user exists or not
96         $sql = "SELECT permission FROM login WHERE id='" . $id . "' AND
97 password='" . $password . "'";
98         $db = new DataAccess();
99         $result = $db->getData($sql);
100         if ($result->num_rows == 1) {
101
102             while ($row = $result->fetch_assoc()) {
103                 $permissionType .= $row['permission'] . ""; //don't put
104 space between "" here. It ruined a lot of my time
105                 // echo "permission " . $row["permission"];
106                 //$permissionType = (string) $row["permission"];
107             }
108
109             //echo "I am in Permisson stuff";
110             //echo $permissionType;
111             //echo gettype($permissionType);
112             return $permissionType;
113         } else {
114             return false;
115         }
116     } catch (Exception $e) {
117         echo 'Caught exception: ', $e->getMessage(), "\n";
118     }
119 }
120
121 public function changePassword($id, $password)
122 {
123     try {
124         $sql = "UPDATE `login` SET `password`= '{$password}' WHERE id =
125 '{$id}'";
126         $db = new DataAccess();
127
128         $db->executeQuery($sql);
129     } catch (Exception $e) {
130         echo 'Caught exception: ', $e->getMessage(), "\n";
131     }
132 }
133
134
```

## LoginController.php

```
1 <?php
2 class LoginController extends Controller
3 {
4
5     public function index()
6     {
```

# Content Rating Site

```
7      // Return a 'view' or do nothing.
8      $this->view('LoginView');
9  }
10 public function phpAlert($msg)
11 {
12     echo '<script type="text/javascript">alert("'" . $msg . '"</script>';
13 }
14
15 public function loginUser()
16 {
17
18     if ($_SERVER['REQUEST_METHOD'] === 'POST') {
19
20         if (empty($_POST['id']) || empty($_POST['password'])) {
21             // echo "Incorrect password.";
22             header("Location: ../index");
23             //$this->phpAlert($error);
24         } else {
25             $user = $this->model('LoginModel');
26             $user->setId($_POST['id']);
27             $user->setPassword($_POST['password']);
28
29
30             if ($user->validateLogin($user->id, $user->password)) {
31
32                 // echo "tesing validation";
33                 //not here yet
34                 $_SESSION["id"] = $user->id;
35                 $_SESSION["password"] = $user->password;
36                 $_SESSION["First"] = "";
37
38
39
40                 // echo $_SESSION["id"];
41                 //echo $_SESSION["id"];
42
43                 if (isset($_SESSION["id"])) {
44                     echo gettype($user->getPermissionType($user->id, $user-
45 >password));
46                     switch ($user->getPermissionType($user->id, $user-
47 >password)) {
48                         case "admin":
49                             //echo " after permission done";
50                             header("Location: ../AdminController/index");
51                             break; //break is needed.Otherwise it will break
52                         case "user":
53                             header("Location: ../UserController/index");
54                             break;
55                         case "critic":
56                             header("Location: ../CriticController/index");
57                             break;
58                         case "contentCreator":
59                             header("Location:
60 ../ContentCreatorController/index");
61                             break;
62                         }
63                     // header("Location: ../views/admin.php");
64                 }
```

## Content Rating Site

```
65         } else {
66             // $this->index();
67             // $this->view('LoginView');
68             // header("Location: ../index");
69             header("Location: ../index");
70
71             // echo "Username or Password is invalid";
72         }
73     }
74 }
75 }
76
77 public function makeId($userType)
78 {
79     $idVal = 1 + $this->model('LoginModel')->getLatestAutoId();
80     if ($userType == "admin") {
81         //get autoIncrement value of login table add some prefix then return
82         $id = "a-" . $idVal;
83         return $id;
84     } else if ($userType == "user") {
85         //get autoIncrement value of login table add some prefix then return
86         $id = "u-" . $idVal;
87         return $id;
88     } else if ($userType == "critic") {
89         //get autoIncrement value of login table add some prefix then return
90         $id = "c-" . $idVal;
91         return $id;
92     } else if ($userType == "contentCreator") {
93         //get autoIncrement value of login table add some prefix then return
94         $id = "cc-" . $idVal;
95         return $id;
96     }
97 }
98
99 public function signUpUser()
100 {
101     if ($_SERVER['REQUEST_METHOD'] === 'POST') {
102         if ($_POST["user_type"] == "user") {
103             //write user object using post data and send it there
104             //echo "I am at user";
105             $userObj = $this->model('UserModel');
106             $id = $this->makeId($_POST['user_type']);
107             //insertUser($name, $id, $email, $phone)
108             $userObj->insertUser($_POST['user_name'], $id,
109 $_POST['user_email'], $_POST['user_phone']); //userTable
110             $loginObj = $this->model('LoginModel');
111             //insertUser($id, $password, $permissionType)
112             $loginObj->insertUser($id, $_POST['user_password'], "user");
113 //login table
114
115         } else if ($_POST["user_type"] == "contentCreator") {
116             //write content Creator object using post data
117
118             $contentCreatorObj = $this->model('ContentCreatorModel');
119             //function insertContentCreator($name, $id, $email, $phone)
120             $id = $this->makeId($_POST['user_type']);
121             $loginObj = $this->model('LoginModel');
122             //insertUser($id, $password, $permissionType)
```

## Content Rating Site

```
123         $loginObj->insertUser($id, $_POST['user_password'],
124 "contentCreator");
125         //insertUser($name, $id, $email, $phone)
126         $contentCreatorObj->insertContentCreator($_POST['user_name'],
127 $id, $_POST['user_email'], $_POST['user_phone']);
128     } else if ($_POST["user_type"] == "critic") {
129         //write content critic object using post data
130         $userObj = $this->model('CriticModel');
131         //function insertContentCreator($name, $id, $email, $phone)
132         $id = $this->makeId($_POST['user_type']);
133         //insertUser($name, $id, $email, $phone)
134         $userObj->insertCritic($_POST['user_name'], $id,
135 $_POST['user_email'], $_POST['user_phone']);
136         $loginObj = $this->model('LoginModel');
137         //insertUser($id, $password, $permissionType)
138         $loginObj->insertUser($id, $_POST['user_password'], "user");
139     }
140     echo "Registration successful. Login you in automatically...";
141     echo "Credentials: " . $id . " Password: " . $_POST['user_password'];
142     //sleep(5);
143
144     header("Location: autoLogin/" . $id . '/' . $_POST['user_password']);
145 }
146 }
147
148 public function autoLogin($id, $password)
149 {
150     $user = $this->model('LoginModel');
151     $user->setId($id);
152     $user->setPassword($password);
153     $_SESSION["id"] = $id;
154     $_SESSION["password"] = $password;
155     $_SESSION["First"] = "First";
156     switch ($user->getPermissionType($user->id, $user->password)) {
157         case "admin":
158             //echo " after permission done";
159             header("Location: ../../AdminController/index");
160             break; //break is needed.Otherwise it will break
161         case "user":
162             header("Location: ../../UserController/index");
163             break;
164         case "critic":
165             header("Location: ../../CriticController/index");
166             break;
167         case "contentCreator":
168             header("Location: ../../ContentCreatorController/index");
169             break;
170     }
171 }
172
173 public function logout()
174 {
175     session_unset();
176     session_destroy();
177
178     header("Location: ../index");
179
180     //$this->view('LoginView');
```

# Content Rating Site

```
    }

    public function loadSignUp()
    {
        $this->view('SignUpView');
    }
}
```

## AdminController.php

```
1  <?php
2  require_once CONTROLLERS . 'CriticController.php';
3  class AdminController extends Controller
4  {
5      public function index()
6      {
7
8          $data = [
9              "Id" => $_SESSION['id'],
10             "Password" => $_SESSION['password'],
11             "totalCritic" => $this->getTotalNumberOfCritic(),
12             "totalUser" => $this->getTotalNumberOfUser(),
13             "totalContentCreator" => $this->getTotalNumberOfContentCreator()
14         ];
15         $this->view("Admin/AdminView", $data);
16     }
17
18     public function getTotalNumberOfCritic() // brings total number of Critic on
19 database
20     {
21         $critic = $this->model('CriticModel');
22         return $critic->getTotalNumberOfCritic();
23     }
24     public function getTotalNumberOfUser() //brings total number of normal user on
25 database
26     {
27         $user = $this->model('UserModel');
28         return $user->getTotalNumberOfUser();
29     }
30     public function getTotalNumberOfContentCreator() //brings total number of
31 Content Creator on database
32     {
33         $cc = $this->model('ContentCreatorModel');
34         return $cc->getTotalNumberOfContentCreator();
35     }
36
37     public function getAllUsers() //Get all the Users with their info
38     {
39         $users = $this->model('UserModel')->getAllUsers();
40         echo json_encode(["data" => $users]);
41     }
42
43     public function getAllContentCreator() //Get all the Content creator with
44 their info
45     {
46         $users = $this->model('ContentCreatorModel')->getAllContentCreator();
47         echo json_encode(["data" => $users]);
48     }
49 }
```



# Content Rating Site

```

48     }
49
50     public function getAllCritic() //Get all the Critic with their info
51     {
52         $users = $this->model('CriticModel')->getAllCritic();
53         echo json_encode(["data" => $users]);
54     }
55
56     public function getAllPendingCritic()
57     {
58         $users = $this->model('CriticModel')->getAllPendingCritic();
59         echo json_encode(["data" => $users]);
60     }
61
62     public function TestingButton()
63     {
64         echo "Hoise toh hasidul eto dum ken";
65     }
66
67     public function getAllContent()
68     {
69         $content = $this->model('ContentModel')->getAllContent();
70         echo json_encode(["data" => $content]);
71     }
72     public function deleteContentFromContentTable($contentId)
73     {
74         $this->model('ContentModel')->deleteContent($contentId);
75     }
76     public function deleteContentFormUserTable($userId)
77     {
78         $this->model('UserModel')->deleteUser($userId);
79         // $this->model('LoginMode')->deleteUser($userId); //baki ase
80     }
81     public function deleteCriticFromCriticTable($userId)
82     {
83         $this->model('CriticModel')->deleteCritic($userId);
84     }
85     public function deleteContentCreatorFormContentCreatorTable($userId)
86     {
87         $this->model('ContentCreatorModel')->deleteContentCreator($userId);
88     }
89 }

```

## C02 and C04 Evaluation: Project Report Evaluation

Completeness (5)	Validation (5)	Feature Implementation (5)	Concept Understanding (5)	Promptness (5)	Total (25)