

Enterprise Web Software Development

Course: COMP1640

Submitted to: Dr Ray Stoneham

Course Leader

University of Greenwich, UK

Submitted by: A.S.M. Minhazul Hoque

ID: 000989813

Submit Date: 13/04/2017

Team Name: Team Zartan

Team Members

S/N	Member's name	Role
1	Mohiminul hossain	programmer
2	Omaer Ahmed	Database designer
3	Kazi Al Amin	UI designer and System analyst
4	A.S.M.Minhazul hoque	Tester, Scrum master

- Application Live
- Screencast URL
- Repository URL

Credentials:

S/N	Role	Username/ email	Password
1			
2			
3			
4			
5			
6			
7			
8			
9			

Table of content

Contents

Introduction.....	4
Product and process evaluation.....	4-5
Group assessment.....	6
Self-evaluation.....	6-7
Appendix-A.....	7-40
Appendix-B.....	40-47
Bibliography.....	48

Introduction

Team Zartan consist total of four members. A system analyst who is also playing the role of UI designer, a database developer, a coder and I took the role of the tester and scrum master in our team. Together we have successfully developed a working web system that performs convenient and automated EC management activities. Following report has been produced specifically from the perspective of a system tester and scrum master's point of view. Necessary contents, evaluations and test information has been compiled and demonstrated in this piece of report.

Product and process evaluation

Product evaluation

Extenuating circumstances (EC) is often cause problems for students as well as university authorities. Management of this issue is vital for any educational institute as lack of proper EC management may delay student's course period which cause misunderstanding between students and their university.

Extenuating circumstances in academic environment refers to any condition that may deprive a student to attend his/her normal academic activities such as attending exam, or submitting assignment. The circumstances can be claimed to the academic authority within recommended time for extensions but that depends upon the severity of the circumstances.

The system that has been built is all about EC management. According to the requirements provided, the system should have adequate database, designated panels functionality and security measures to execute convenient EC activities between students and authority staffs of a university.

Process evaluation

The system that has been tested is a role based secured web system which has sufficient functionalities to provide and process any EC claims. This system follows a hierarchy of management system, that is, it has an admin panel where an admin has sufficient flexibilities to manage academic claims for each session. Admin also has the authority to manage coordinators.

Next in the hierarchy comes the manager. According to the requirements, manager has the access to overview the overall EC processes and manager has his/her own panel where scheduled reports can be produced based on the overviewed EC activities. Those features are thoroughly and individually tested.

EC coordinator's panel has been tested in detail. EC coordinators have individual panel separated by sessions for performing their required activities. It has been considered that each faculty would have their own EC coordinator. EC coordinator has the flexibility to accept or reject designated faculty student's claims. A pending option has also been given in case a coordinator requires time for decisions.

Fourteen days of countdown time has been set as a reminder for the coordinator to process any claim. After that a particular unprocessed claim goes obsolete. The system is capable of providing those facilities. In addition with those, system development also ensured any accepted or rejected claims to be notified to its respected claimer. Each of those mentioned functionalities are tested and evaluated.

The EC management system is nothing without claimers. Here student claim the EC. So a designated student panel has been developed, sessions are again implemented so that contents of one student cannot be available to other student. Here in the system, a particular student can choose their subjects and modules. Respected assessments are automatically appears accordingly. The student then can see his/her EC closure period that has been fixed by the administrator beforehand. Information is displayed on last ` dates of EC claim for respected assessment. A closure date for evidence submission is also displayed. Student can send claim request to the EC coordinator, keeping in mind the claim closure period. Claim accepted or rejected are notified back to the student's panel once processed by the faculty EC coordinator.

As for authentication testing point of view the system has been equipped with necessary processes such as forget password recovery facility which has been tested separately, responsiveness, cross browser compatibilities, various types of authentication options and different security measures are used to keep the site safer. Each of those processes has been tested and described thoroughly later on in this report.

Every stage of system development ensured proper usage of agile scrum methodology. A complete user story has been made followed by product backlog and sprint back log. Minutes of meetings and day to day stand up meetings or at least regular online conferences on development progress are conducted to facilitate daily progress. A final burn down chart has been produced according to the pace of agile methodology. Although I took the responsibility to test the system but I also voluntarily played the role of scrum master in our team, therefore it was my responsibility to ensure all the members in this team follow this methodology in every development procedures. Detail agile processes have been provided in the appendix part of the report.

Group assessment

S/N	Name	Role	Comment	Performance rating
1	Omaer Ahmed	Database developer	System developer gave an overall good afford in his designated task, Although minor irregularities were observed during group meeting but that didn't impose any negative impact on the overall progress of the system	7/10
2	Kazi Al Amin	System analyst, UI designer	Showed great afford and enthusiasm at the beginning but ended up with some causalities, but overall he gave a fair assistance to the developer for gathering requirement data	8/10
3	Mohiminul hossain	Developer	He tried his very best to fill up the deficiencies made by system analyst and DB developer, at the same time competed his development role efficiently.	9/10

Self-evaluation

I played the role as a tester and scrum master in this assignment. I will not say I played my role perfectly. I had some deficiencies and laggings too but throughout this group work I have succeeded to establish an active and friendly environment in our team. As far as I have learned in my previous academic years, an agile group cannot be established until each member in the group thinks agile. Therefore Instead of blaming culture I have suggested my team members to be more flexible and dynamic so that issues and deficiencies can be handled together. On that state of team morality, even being a tester of the team, besides working on my testing role, I have performed several UI design and fixed numbers of bugs of our system. I have

also tried my level best to assist members of our team during any system related distresses.

Appendix-A

Reason for testing

Various testing has been conducted right after every major module developments in this system. The reason for such processes is to ensure that the end product meet the given requirements. Proper testing also assures final product quality.

Audience's role

1. End users: Numbers of stakeholders can be in this group. These people can be among the end users who will be involved throughout the system development procedure and will suggest revision and modifications until system requirements fulfill their given specifications.
2. Developer team: A team of four to seven people who takes the credit of developing the system. Such team may include database developer, GUI developers, coders, system analyst, testers and scrum master.
3. Tester: Tester comes from the developer team who has the specific responsibility to test the newly introduced system thoroughly to insure proper functionality and assure product quality.

Principle focus of testing

The testing will be conducted on the basis of few principles that reflect business ethics and norms. Below are the principles:

- The primary focus is to meet given requirements. Test is performed to ensure prioritized requirements are focused and properly fulfilled.
- Numbers of testing have been conducted on implemented requirements to assure overall product quality. This ensures convenience and reliability during usage.
- One last principle focus of testing is to ensure stability and keeping up to the mark standard of the system. In addition with all those, system consistency also plays a major role in this criterion.

Key test assumptions

Below are some test assumptions that should to be available during test attempt:

- Valid data must be retrieved from stakeholders in order to carry out all the relevant testing processes.

- All the system testing procedures will be solely monitored by the tester until the tester chose any compatible personal to carry out the task as a proxy.
- System related stakeholder supposed to be available during performance testing of the system, as stakeholders will be responsible to carry out this particular test.
- Tester will be giving all test related suggestions, helps and queries during system development and during testing processes.

List of tests attempted

1. Responsive testing
2. Compatibility testing
3. Cross browser testing
4. Security testing
5. Usability testing
6. Database testing
7. Functionality testing
8. Performance testing
9. Integration testing
10. Validation testing
11. Unit testing
12. Sprint backlog testing

Test Environment

- Device: Laptop (Dell Inspiron 3443)
- Operating system: Windows 10 Home (64 bit)
- Browser: Google chrome (version 57.0.2987.133), Microsoft Edge 38.14393.0.0, Mozilla Firefox 52.0.2, Safari 5.1.4.

1. Responsive testing

Similar to the compatibility testing but instead of running the system on different operative platform, this time the system is tested on various computing devices such as desktop pc, laptop of various screen sizes and resolutions, tablet, phablets of all sizes or other small screened smart mobile devices. In order to pass this test the system must run efficiently in all devices and must automatically adapt to different screen dimensions and resolutions without unnecessary horizontal scrolls.

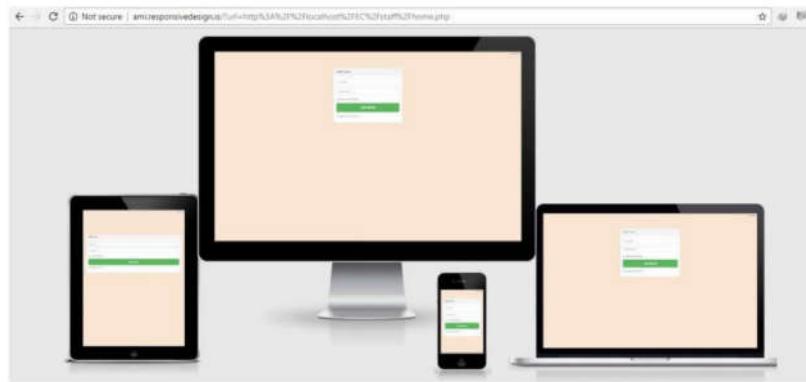


Figure 1 Online responsive testing tool is used to test web responsiveness
(am i responsive design, n.d.)

2. Compatibility testing

This is one of the basic testing procedures that determine how far the system is compatible on different machine OS platform. For executing this test process the system that has been subjected for this testing are initiated on different operating platform such as windows or mac.

3. Cross browser testing

Sometime some system shows various functionality or UI issues when run on different browsers. This test is dedicated to those issues. Here same system was run on various browsers such as Internet explorer, Mozilla Firefox, google chrome, safari and opera browsers.

In my web-testing career I have experienced this as the most influencing part on web site testing. These are some of the most popular web browsers widely used. Therefore testing was executed on each of those browsers for determining systems cross browsing compatibility

Welcome Minhaz

Student- Computer Science and Engineering Department

Enter claim Description

Select Module Name

Select Assessment Type

Select EC Type

Describe Your Claim
200 character remaining
Not more than 200 character...

Claim

Please select your module name and assessment type to see their date details

Activate Windows
Go to Settings to activate Windows.

Figure 2 Google Chrome

Welcome Minhaz

Student- Computer Science and Engineering Department

Enter claim Description

Select Module Name

Select Assessment Type

Select EC Type

Describe Your Claim
200 character remaining
Not more than 200 character...

Claim

Please select your module name and assessment type to see their date details

Activate Windows
Go to Settings to activate Windows.

Figure 3 Mozilla Firefox

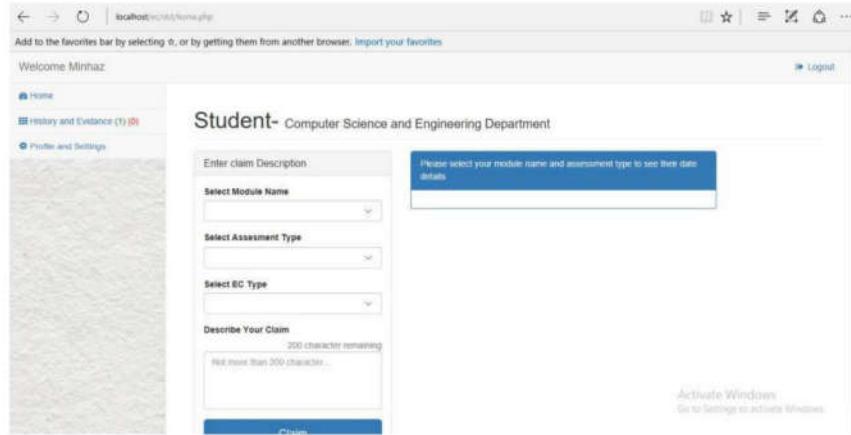


Figure 4 Internet Explorer

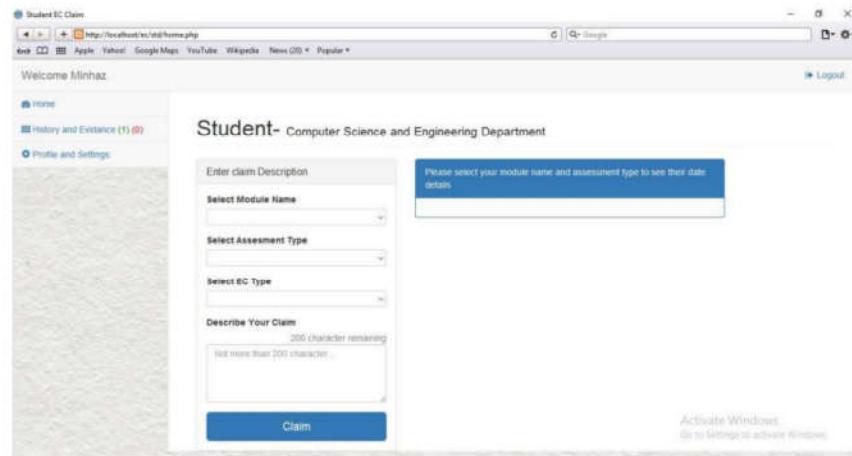


Figure 5 Safari

4. Security testing

A system is of no value if it doesn't have sufficient authentication and privacy. Various security measures can be given in a system depends upon importance of the content data. These security processes saves the system from foreign vulnerabilities and various harmful cyber-attacks.

The system that has been build has some useful security processes coded for a reliable and safe system usage. Below are the security measures that have been tested:

- Once authorized login occurs, users can get the access of page contents. If that user copy the URL of the home page and try to use that again after

logout, the user will not be able to do so. Internal page URL cannot be directly used onto the browser to access any content without login validation.

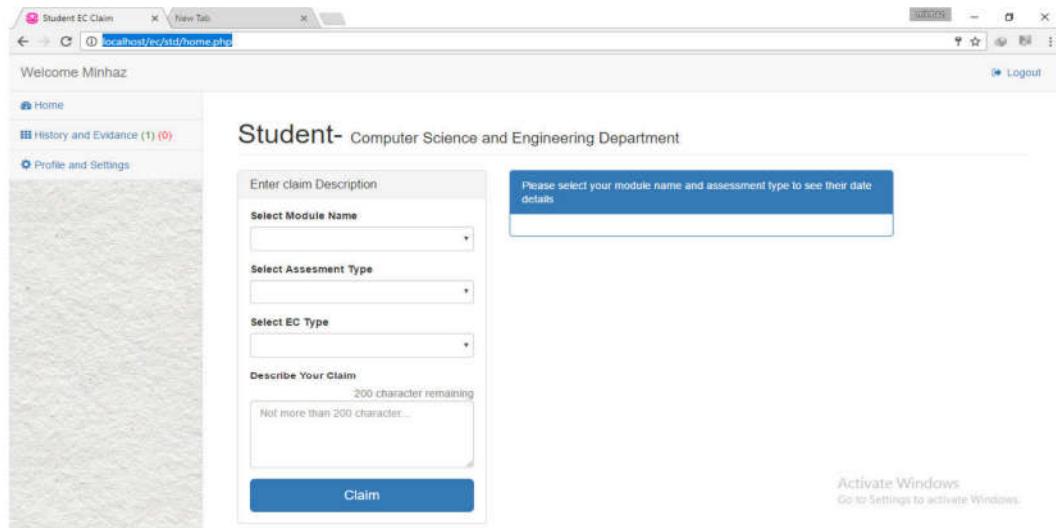


Figure 6 Home page appears after a successful login, link copied from the home page

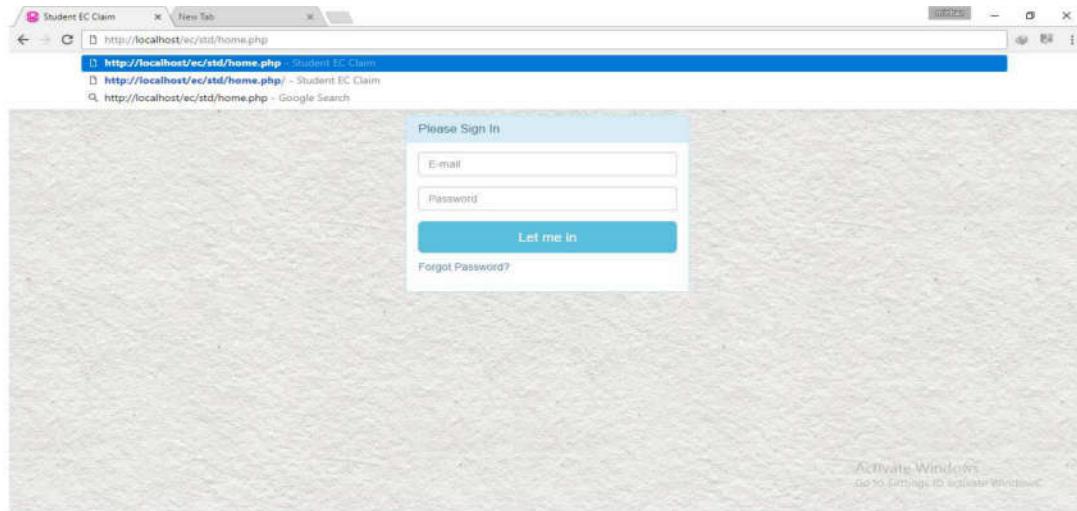


Figure 7 after logout, copied URL has been used on the search bar to get direct access

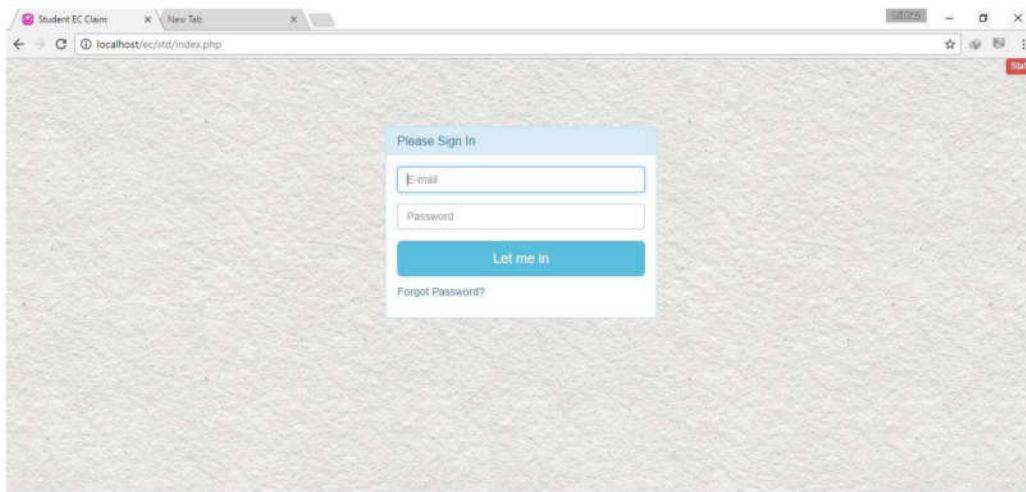


Figure 8 The request has been failed and the system automatically returns back to the same page

- Login authentication and validation has been tested thoroughly; in case of empty fields, invalid or wrong data warning notification will appear.

Please Sign In

E-mail
shaon.minhaz@gmail.com

.....

Let me in

Forgot Password?

Please Sign In

E-mail

Password

Let me in

Forgot Password?

Figure 9 if email field is empty it shows warning notification upon clicking 'Let me in' button

Please Sign In

shaon.minhaz@gmail.com

Password

Let me in

Forgot Password?

Please Sign In

shaon.minhaz@gmail.com

Password

Let me in

Forgot Password?

Figure 10 if password is empty, upon clicking the 'Let me in' button warning notification pops up.



Figure 11 if wrong password or wrong email is provided warning massage pops up.

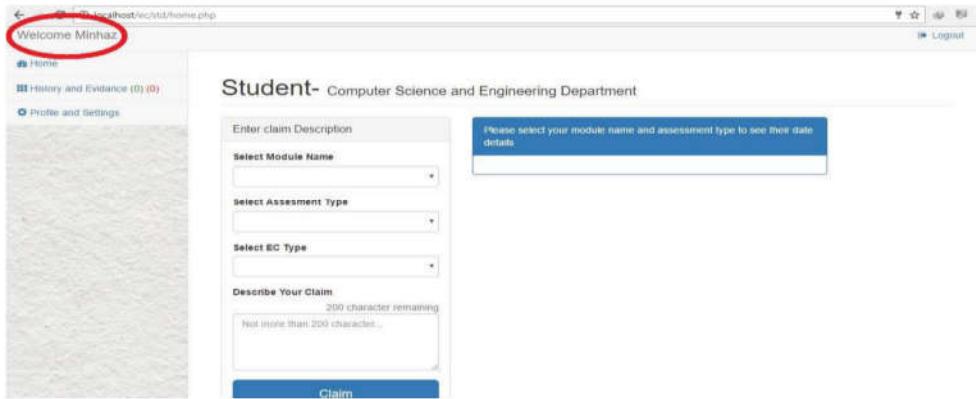
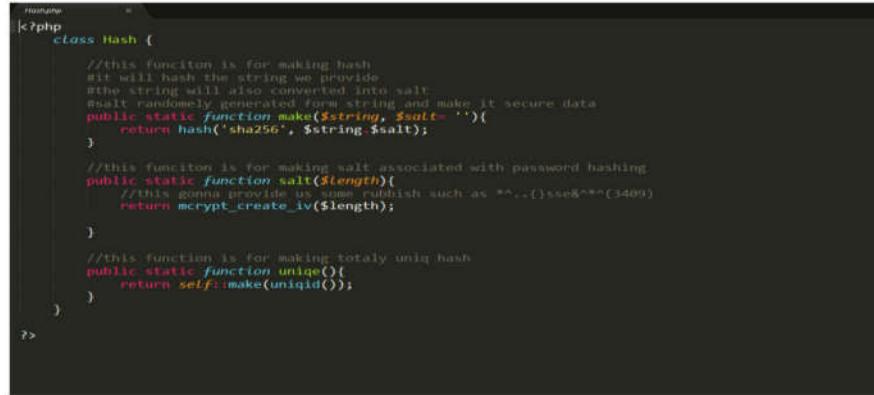


Figure 12 valid email and password allow a user to get access to the home page

- Hash and salt technology is used to encrypt the password. This is commonly used to make the password unusable for the unauthorized intruders who can misuse and may cause confidentiality issues in the database. Here **SHA 256** hashing is used for password security.



```
Hash.php
<?php
class Hash {
    //this function is for making hash
    //it will hash the string we provide
    //the string will also converted into salt
    //salt randomly generated form string and make it secure data
    public static function make($string, $salt= ''){
        return hash('sha256', $string . $salt);
    }

    //this function is for making salt associated with password hashing
    public static function salt($length){
        //this gonna provide us some rubbish such as ^^...sse&^^3409
        return mcrypt_create_iv($length);
    }

    //this function is for making totally uniq hash
    public static function unique(){
        return self::make(uniqid());
    }
}

?>
```

Figure 13 Code of hash function



```
$salt = Hash::salt(32);

$user = DB::getInstance()->insert('ec_coo',array(
    'coo_name' => Input::get('name'),
    'coo_email' => Input::get('email'),
    'coo_password' => Hash::make(Input::get('password'), $salt),
    'salt' => $salt,
    'fac_name' => Input::get('faculty'),
    'group' => '4'
));
```

Figure 14 this is how SALT security has been inserted in hashing password

- Sql injection protection has been tested. PDO has been used to save the system database from sql injection. This is also used to improve query results and make the database driver modification easier and more convenient.

```
private function __construct(){
    try{
        $this->_pdo = new PDO('mysql:host=' . Config::get('mysql/host') . ';dbname=' .
            Config::get('mysql/db'),
            Config::get('mysql/username'),
            Config::get('mysql/password'));
        //echo 'connected';
    }catch(PDOexception $e){
        die($e->getMessage());
    }
}

public static function getInstance(){
    if(!isset(self::$_instance)){
        self::$_instance = new DB();
    }
    return self::$_instance;
}

public function query($sql, $params = array()){
    $this->_error = false;
    if($this->_query = $this->_pdo->prepare($sql)){
        $x=1;
        if(count($params)){
            foreach($params as $param){
                $this->_query->bindValue($x, $param);
                $x++;
            }
        }
        if($this->_query->execute()){
            $this->_results = $this->_query->fetchAll(PDO::FETCH_OBJ);
            $this->_count = $this->_query->rowCount();
        }else{
            $this->_error = true;
        }
    }
    return $this;
}
```

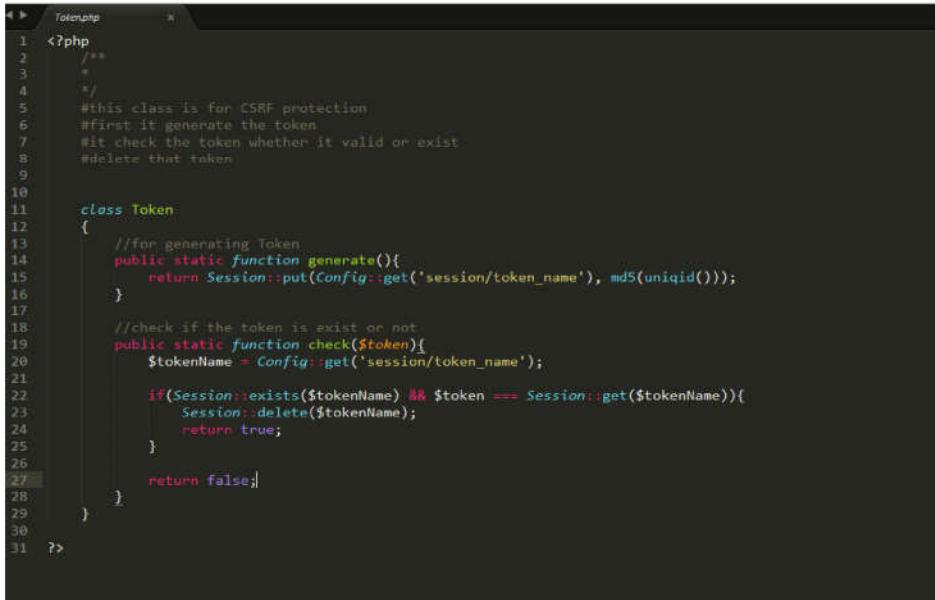
Figure 15 Code for SQL Injection protection process

- Text fields are protected from unauthorized scripting symbols. This will prevent any sort of illegal or unauthorized injections to the input field of the system.

```
function clean(e){  
    var textfield = document.getElementById(e);  
  
    var regex = /[^\w\s]/gi;  
  
    textfield.value = textfield.value.replace(regex, "");  
}
```

Figure 16 certain injecting characters such as ";" - # '=' will not work on input fields of the system.

- Cross Site Request Forgery or CSRF is a forgery attack where malicious users exploit the system and inject harmful or unauthorized commands. Appropriate protection has been implemented in the system against this threat. The code on CSRF protection process has been separately snipped for testing purpose.



```

1 <?php
2 /**
3 *
4 */
5 #this class is for CSRF protection
6 #first it generate the token
7 #it check the token whether it valid or exist
8 #delete that token
9
10
11 class Token
12 {
13     //for generating Token
14     public static function generate(){
15         return Session::put(Config::get('session/token_name'), md5(uniqid()));
16     }
17
18     //check if the token is exist or not
19     public static function check($token){
20         $tokenName = Config::get('session/token_name');
21
22         if(Session::exists($tokenName) && $token === Session::get($tokenName)){
23             Session::delete($tokenName);
24             return true;
25         }
26
27         return false;
28     }
29 }
30
31 ?>

```

Figure 17 Code for CSRF attack protection process



```

<form role="form" action="" method="post">
    <fieldset>
        <div class="form-group">
            <input class="form-control" placeholder="E-mail" name="email" type="email" value=<?php echo escape(Input::get('email'))?>" autofocus>
        </div>
        <div class="form-group">
            <input class="form-control" placeholder="Password" name="password" type="password" value="">
        </div>
        <div class="checkbox">
            </div>
            <!-- Change this to a button or input when using this as a form -->
            <input type="hidden" name="token" value=<?php echo Token::generate()?>">
            <input type = "submit" value="Let me in" class="btn btn-lg btn-success btn-block">
        <div class="checkbox">
            <a href="forgot_password.php">Forgot Password?</a>
        </div>
    </fieldset>
</form>

```

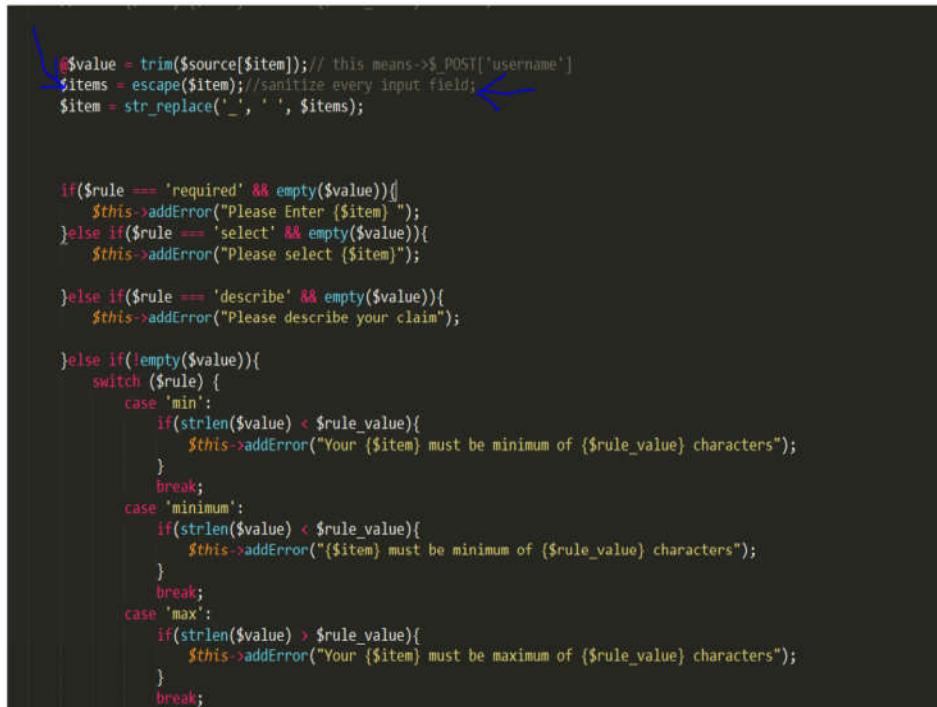
Figure 18 this is how CSRF protection is applied in salt

- XSS or cross site scripting protection has been applied in the system. XSS protection is used to prevent any scripting code injection in any part of system forms.



```
<?php
function escape($string){
    return htmlentities($string, ENT_QUOTES, 'UTF-8');
}
?
?>
```

Figure 19 Code of XSS protection process



```
$value = trim($source[$item]); // this means->$_POST['username']
$item = escape($item); //sanitize every input field;
$item = str_replace('_', ' ', $item);

if($rule === 'required' && empty($value)){
    $this->addError("Please Enter {$item} ");
} else if($rule === 'select' && empty($value)){
    $this->addError("Please select {$item} ");
}

else if($rule === 'describe' && empty($value)){
    $this->addError("Please describe your claim");
}

else if(!empty($value)){
    switch ($rule) {
        case 'min':
            if(strlen($value) < $rule_value){
                $this->addError("Your {$item} must be minimum of {$rule_value} characters");
            }
            break;
        case 'minimum':
            if(strlen($value) < $rule_value){
                $this->addError("{$item} must be minimum of {$rule_value} characters");
            }
            break;
        case 'max':
            if(strlen($value) > $rule_value){
                $this->addError("Your {$item} must be maximum of {$rule_value} characters");
            }
            break;
    }
}
```

Figure 20 this is how XSS protection process has been applied in the system

- Testing has been conducted on URL filtering process. The process is applied on the system so that prevention can be ensured if certain symbols and characters such as '@. %#', etcetera is used in the system URL to gain illegal access.

```

<form role="form" method="post" action="">
<fieldset>
<?php
$id = Input::get('id');
$filterid = filter_var($id, FILTER_SANITIZE_NUMBER_INT); ←

$data = DB::getInstance() ->query("SELECT
c.co_name,
c.fac_name,
d.dep_name,
m.mod_title,
cl.claim_no,
cl.claim_details,
cl.ec_type,
cl.module_title,
cl.assessment,
cl.claim_date,
cl.claim_feedback,
DATEDIFF(cl.last_date,CURDATE()) as remaining_day,
cl.evidence_details,
cl.std_id
FROM
ec_coo AS c
INNER JOIN
department AS d ON c.fac_name = d.fac_name
INNER JOIN
module AS m ON m.dep_name = d.dep_name
INNER JOIN
claim AS cl ON cl.module_title = m.mod_title
WHERE
cl.claim_no = ? and c.fac_name = ?",
array("$filterid","$fac_name"));

```

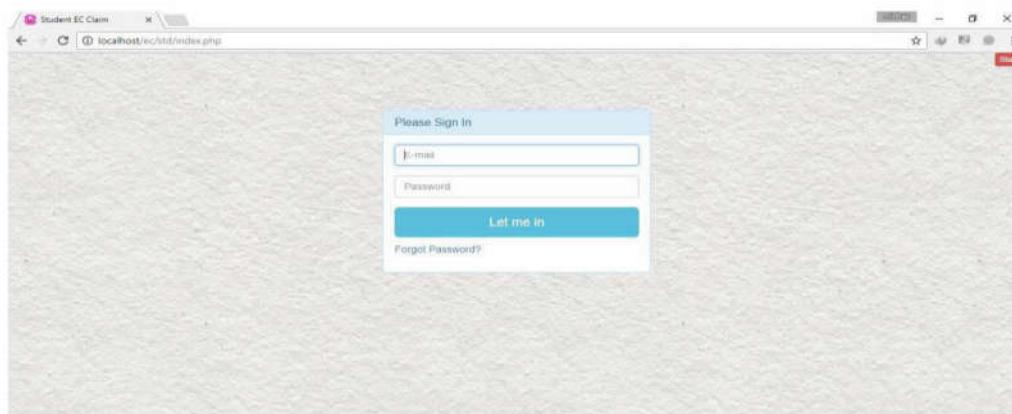
Figure 21 this is how URL filtration process has been implemented in the system

5. Usability testing

Usability testing has been conducted to check convenient usability, easy to use and flexibilities in the system. The test is conducted on the basis of better UI, comfortable usability, consistency, easier navigation and control options of the system.

Below are the screenshots that has been used to support usability testing of this system:

1. Simplicity in UI design for student and staff login page. The colors and textured background used is eye friendly and doesn't create issues on regular user activities.



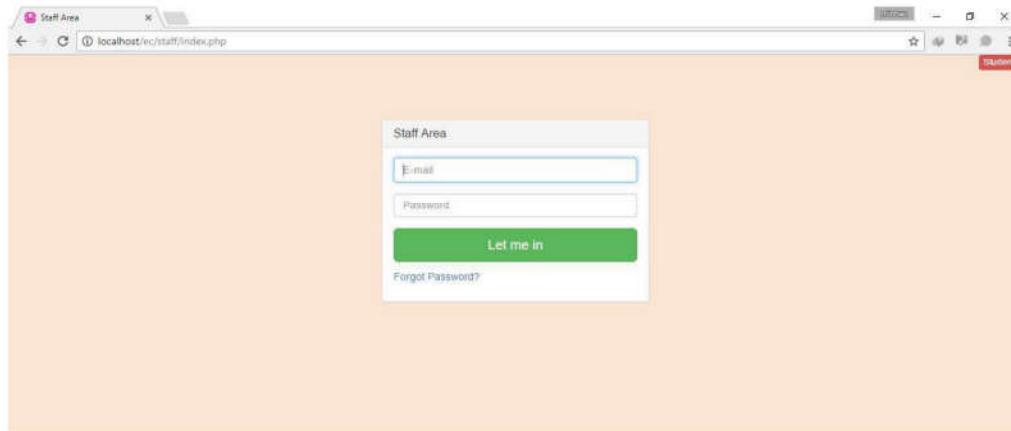


Figure 22 students and staff login page

2. Presence of appropriate notifications, help or instruction massages and sufficient statements and headings for convenient usability.

The screenshot shows the "Student EC Claim" dashboard. A sidebar on the left lists "Process Update notification" and "Page navigation options". The main area has a "Welcome and ID statements" header. It includes a "Student- Computer Science and Engineering Department" section, a "Please select your module name and assessment type to see their date details" message, and an "Instruction statements" box. A "Claim" button is at the bottom. Red boxes highlight the "Welcome and ID statements" header, the "Student" section, the "Please select your module name and assessment type" message, and the "Instruction statements" box. Arrows point from the sidebar's "Page navigation options" to the "History and Evidence" and "Profile and Settings" links in the main menu.

Figure 23 Different notifications, navigations and instructions are highlighted in a dashboard.

3. Proper search option and pagination functionalities are available in pages that have large number of data.

Staff Area

Welcome EC Manager

Logout

All Assessment Details

Assessment Details

Search Data...

Serial No Assignment Type Due Date Last Evidence Submission Date Department Name Module Name Semester Name

Serial No	Assignment Type	Due Date	Last Evidence Submission Date	Department Name	Module Name	Semester Name
1	assignment	2018-04-30	2018-04-30	Computer Science and Engineering	Introduce to Python	3
2	assignment	2018-04-30	2018-05-10	Computer Science and Engineering	Web design	3
3	assignment	2017-08-31	2017-09-09	English	DELS	2
4	exam	2017-09-30	2017-10-10	English	Advance literature	2
5	assignment	2017-09-30	2017-09-28	Computer Science and Engineering	Introduce to Network	2
6	exam	2017-09-27	2017-10-07	Computer Science and Engineering	Algorithm	2
7	exam	2017-04-27	2017-05-07	English	English for Academic purpose	1
8	exam	2017-04-30	2017-05-10	English	English grammar	1
9	exam	2017-04-25	2017-05-05	Computer Science and Engineering	Data structure	1
10	assignment	2017-04-30	2017-04-30	Computer Science and Engineering	C programming	1

Activate Windows
Go to Settings to activate Windows.

1 ← Proper pagination

Figure 24 pagination and search option for retrieving data from bunch of information

6. Database testing

This testing determines database efficiency. Through this testing DB securities and integrities are also determined. CRUD performances in the system are dependent on the database efficiency. So by determining system CRUD, database efficiency can be determined.

Encrypted passwords

Hash and Salt implementations

	student_id	std_name	std_email	std_dob	std_gender	std_institute	std_password
1	NULL	rahulshah9@gmail.com	NULL	male	MALE	MALE	argY1+e...8z++H 2tAVQGzv+oed
2	NULL	rahulshah9@gmail.com	NULL	male	MALE	MALE	argY1+e...8z++H 2tAVQGzv+oed
3	NULL	rahulshah9@gmail.com	NULL	male	MALE	MALE	argY1+e...8z++H 2tAVQGzv+oed
4	NULL	Shivam mishra@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
5	NULL	apoorwika@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
6	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
7	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
8	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
9	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
10	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
11	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
12	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
13	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
14	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
15	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
16	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
17	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
18	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
19	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
20	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
21	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
22	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
23	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
24	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed
25	NULL	rahulshah9@gmail.com	NULL	NULL	NULL	NULL	argY1+e...8z++H 2tAVQGzv+oed

Figure 25 having the data encrypted meets a major requirement in an efficient database

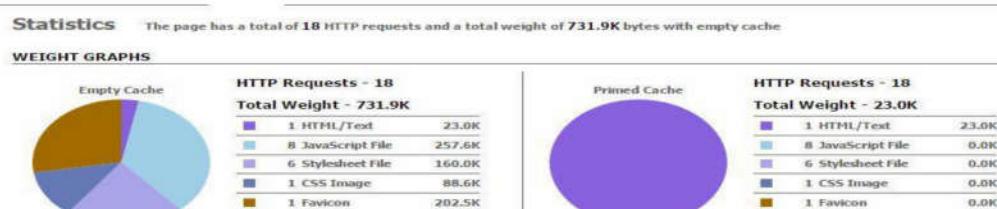
7. Functional testing

This test process checks all the internal and outgoing links of the system. The input methods and consistency of user forms are also checked here. Other than that broken links and any existence of isolated or unused pages or links and cookies are also checked here.

Following are some screenshots from the important functional activities of the system:

8. Performance testing

Performance testing shows how system performs under various conditions such as during web load or certain conditions that needs system's highest performing abilities. Multiple users had been logged at the same time under same server to check the load testing of the system. Online performance testing tool has also been used to make a further evaluation of the system.



Copyright © 2017 Yahoo! Inc. All rights reserved.

Figure 26 Online web performance testing tool has been used to test this criteria

9. Integration testing

In this test process multiple units of system modules are tested as a cluster to sort out system functions as a whole. More than one module can be responsible for certain functions of a system. Integrity testing performs on those functions.

The screenshot shows a web-based application interface for a student to submit an Evidence Claim (EC). The top navigation bar includes links for Home, History and Evidence (0), Profile and Settings, and Logout. The main section is titled "Student- Computer Science and Engineering Department". It contains several input fields: "Enter claim Description" (with placeholder text "Please select your module name and assessment type to see their details"), "Select Module Name" (set to "C programming"), "Select Assessment Type" (set to "assignment"), "Select EC Type" (set to "Finance"), and "Describe Your Claim" (containing the text "financial crisis"). There is also a "Claim" button at the bottom. A message "Activate Windows Go to Settings to activate Windows." is visible on the right side.

Figure 27 A student EC claim has been made

The screenshot shows a "Claim and Evidence Details" page. The top navigation bar is identical to Figure 27. The main content area is titled "Your claim and evidence details" and displays a success message "Claim submitted successfully , please upload evidence!". Below this is a search bar labeled "Search Data...". A table lists three claims, with the third one highlighted by a red box. A red arrow points from the text "EC claim has been recorded" in the previous figure to this row. The table columns are: Serial no, Claim Module, Module Assessment Type, EC Type, Your Claim Details, Your Evidence Status, Your claim feedback, Your claim date, and Upload Evidence. The third row (highlighted) contains: 3, C programming, assignment, accident, "Had a serious accident recently, currently in hospital.", available, pending, 2017-04-11, Upload Evidence. The other two rows are: 1, Data structure, exam, other, "Going outside the country for professional reasons", not available, processing, 2017-04-11, Upload Evidence; and 2, C programming, assignment, finance, financial crisis, not available, pending, 2017-04-11, Upload Evidence. A message "Activate Windows Go to Settings to activate Windows." is at the bottom right.

Serial no	Claim Module	Module Assessment Type	EC Type	Your Claim Details	Your Evidence Status	Your claim feedback	Your claim date	Upload Evidence
1	Data structure	exam	other	Going outside the country for professional reasons	not available	processing	2017-04-11	Upload Evidence
2	C programming	assignment	finance	financial crisis	not available	pending	2017-04-11	Upload Evidence
3	C programming	assignment	accident	Had a serious accident recently, currently in hospital.	available	pending	2017-04-11	Upload Evidence

Figure 28 EC claim list for this student. This claim has also been sent to the coordinator's dashboard

Welcome Minhz

[Logout](#)

[Home](#)

[History and Evidence \(0\)](#)

[Profile and Settings](#)

Upload Evidence

Your claim Details

Claim Module: C programming

Assessment type: assignment

EC Type: finance

Your claim description: financial crisis

You can Upload evidence only Once

Upload Evidence in Image or PDF

Your file has been uploaded successfully
Choose File: No file chosen

Upload

Your Uploaded Evidence

ABC Corporation
Comparative Balance Sheet
December 31, 1991, 1990, 1989

Assets	1991	1990	1989
Cash and cash equivalents	\$1,000	\$1,000	\$1,000
Accounts receivable	\$2,000	\$2,000	\$2,000
Inventories	\$3,000	\$3,000	\$3,000
Prepaid expenses	\$400	\$400	\$400
Total current assets	\$6,400	\$6,400	\$6,400
Property, plant and equipment	\$10,000	\$10,000	\$10,000
Less accumulated depreciation	(\$4,000)	(-\$4,000)	(-\$4,000)
Total property, plant and equipment	\$6,000	\$6,000	\$6,000
Less current maturities of long-term debt	\$1,000	\$1,000	\$1,000
Net property, plant and equipment	\$5,000	\$5,000	\$5,000
Less allowance for doubtful accounts	(-\$100)	(-\$100)	(-\$100)
Less inventories held for sale	(-\$200)	(-\$200)	(-\$200)
Less other assets	(-\$100)	(-\$100)	(-\$100)
Total assets	\$11,300	\$11,300	\$11,300
Liabilities			
Accounts payable	\$2,000	\$2,000	\$2,000
Accrued liabilities	\$1,000	\$1,000	\$1,000
Current maturities of long-term debt	\$1,000	\$1,000	\$1,000
Less current maturities of long-term debt	\$1,000	\$1,000	\$1,000
Less other liabilities	(-\$100)	(-\$100)	(-\$100)
Total liabilities	\$3,100	\$3,100	\$3,100
Stockholders' equity	\$8,200	\$8,200	\$8,200
Less treasury stock	(-\$100)	(-\$100)	(-\$100)
Retained earnings	\$8,300	\$8,300	\$8,300
Total stockholders' equity	\$8,300	\$8,300	\$8,300
Total assets = Total liabilities + Stockholders' equity	\$11,300	\$11,300	\$11,300

Activate Windows

Figure 29 evidence has been uploaded

Welcome coordinator 1

Pending (1)

Profile and Setting

Co-ordinator- Faculty of science

Search Data ...

On clicking view, Coordinator will be on a new page that has detail of this new claim and decision options for those claims.

Serial no	Claim Module	Student ID	Module Assessment Type	EC Type	Claim Details	Evidence Status	Claim feedback	Claim date	View claim
1	C programming	9	assignment	finance	financial crisis	available	pending	2017-04-11	View
2	Data structure	9	exam	other	Going outside the country for professional reasons	not available	processing	2017-04-11	View
3	C programming	7	assignment	family	facing family crisis	not available	pending	2017-04-11	View
4	C programming	9	assignment	accident	Had a serious accident recently. currently in hospital.	available	pending	2017-04-11	View

Figure 30 In coordinator's dashboard the new claim entry can be seen at the top of the list

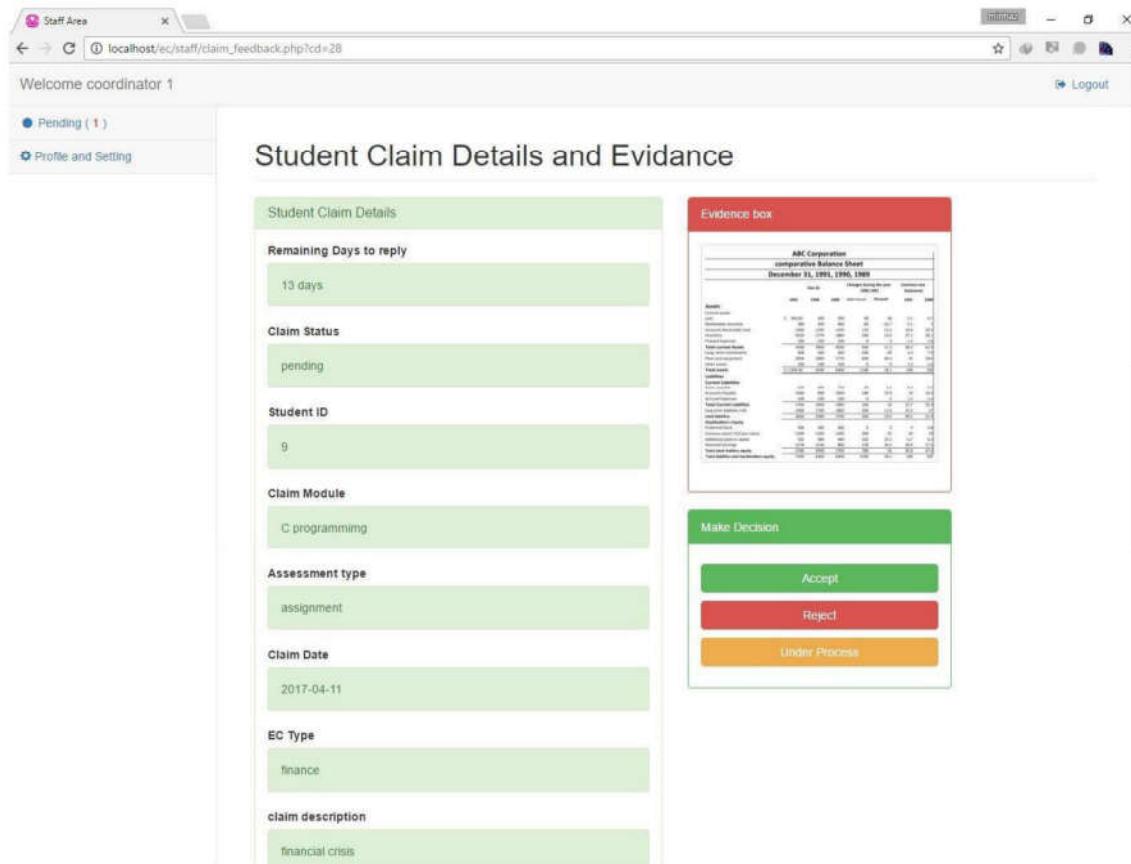


Figure 31 Detail of the claim along with evidence appears for the coordinator to make decisions

10. Validation testing

Validation testing ensures that the system meets required specifications and fulfills functional purposes. The testing is conducted to keep the clients actual requirements in mind and validates functionalities according to it.

Below are some screenshots that validates different required functionalities of the system:

Welcome coordinator 1

[Logout](#)

Pending (1)

Profile and Setting

Profile and Settings

Your profile Details

Info	Details
First Name	coordinator 1
Email	coordinator1@gmail.com

Change Your Email

This email is not available. Choose another

[Update](#)

Change Your password

Current Password	
New Password	
Confirm Password	
Change	

Providing same email address in update email field will give you validation warning message.

Figure 32

Welcome coordinator 1

[Logout](#)

Pending (1)

Profile and Setting

Profile and Settings

Your profile Details

Info	Details
First Name	coordinator 1
Email	coordinator1@gmail.com

Change Your Email

[Update](#)

Change Your password

This new-password doesn't match with each other	
Current Password	
New Password	
Confirm Password	
Change	

Validation message appears if new password chosen and confirm password doesn't match with each other

Figure 33

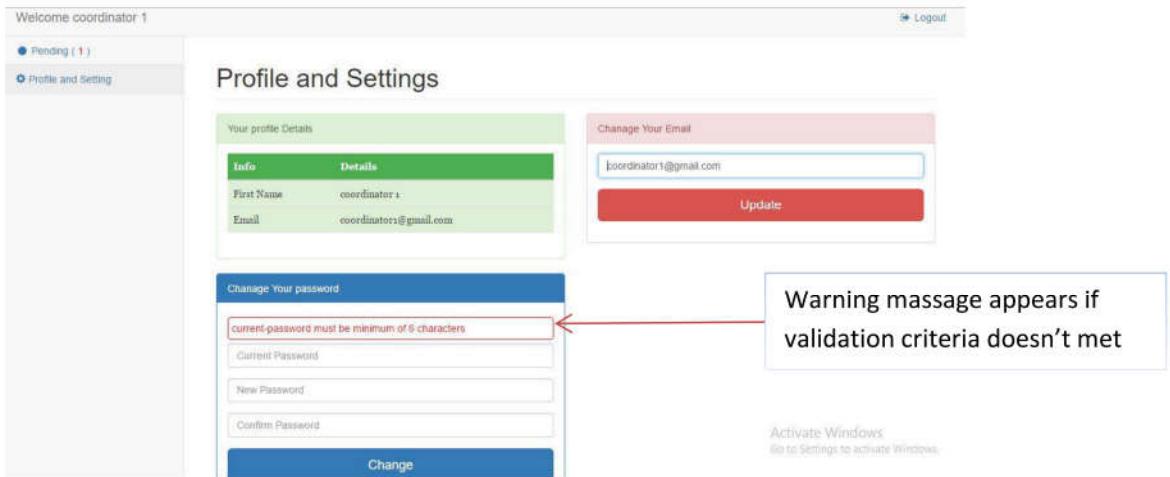


Figure 34 Necessary warning massages appears if invalid inputs are provided

1. Unit testing

Sessions

```
<?php
//this page must include all over the page
ob_start();
session_start();

$GLOBALS['config'] = array(
    'mysql' => array(
        'host' => '127.0.0.1',
        'username' => 'root',
        'password' => '',
        'db' => 'ec'
    ),

    'remember' => array(
        'cookie_name' => 'hash',
        'cookie_expiry' => 60800
    ),

    'session' => array(
        'session_name_1' => 'user',
        'session_name_2' => 'staff',
        'token_name' => 'token'
    )
);

spl_autoload_register(function($cClass){
    require_once dirname(__DIR__).'/classes/'.$cClass.'.php';
});

require_once dirname(__DIR__).'/function/sanitize.php';
```

This will guide students to the student panel

This will navigate user to the staff panel

Figure 35 Session code will guide different users to separate panel

```

class Staff {
    private $_db,
        $_data, // for storing data
        $_session_name_1,
        $_isLoggedIn; //for checking that user logged In or Not

    public function __construct($user= null){
        $this->_db = DB::getInstance();

        $this->_session_name_2 = Config::get('session/session_name_2');

        if(!$user){
            if(Session::exists($this->_session_name_2)){
                $user = Session::get($this->_session_name_2);

                if($this->staff_find($user)){
                    $this->_isLoggedIn = true;
                }else {
                    //process logout
                }
            }else {
                $this->staff_find($user);
            }
        }
    }
}

```

Figure 36 staff session has been applied here

```

class User {
    private $_db,
        $_data, // for storing data
        $_session_name_1,
        $_isLoggedIn; //for checking that user logged In or Not

    public function __construct($user = null){
        $this->_db = DB::getInstance();

        //get session form init class
        $this->_session_name_1 = Config::get('session/session_name_1');

        //for student
        if(!$user){
            if(Session::exists($this->_session_name_1)){
                $user = Session::get($this->_session_name_1);

                if($this->find($user)){
                    $this->_isLoggedIn = true;
                }else {
                    //process logout
                }
            }else {
                $this->find($user);
            }
        }
    }
}

```

Figure 37 student session has been applied here

Login/Logout

```

    //for student
    if(!$user){
        if(Session::exists($this->_session_name_1)){
            $user = Session::get($this->_session_name_1);

            if($this->find($user)){
                $this->_isloggedin = true;
            }else {
                //process logout
            }
        }else {
            $this->find($user);
        }
    }

    //function for create data in database table user
    public function create($fields =[]){
        if($this->_db->insert('student', $fields)){
            throw new Exception('there was a problem to insert data');
        }
    }

    //for student find and login
    private function find($user = null){
        if($user){
            $field = (is_numeric($user)) ? 'std_id' : 'std_email';
            $data = $this->_db->get('student',array($field,'=',$user));

            if($data->count()){
                $this->_data = $data->first();
                return true;
            }
        }
    }

    public function login($email= null, $password = null){
        $user = $this->find($email);

        if($user){
            if($this->data()->std_password === Hash::make($password, $this->data()->salt)){
                Session::put($this->_session_name_1, $this->data()->std_id);
                return true;
            }
        }
    }
}

```

Figure 38 code for system login validation

```

<?php
require_once dirname(__DIR__).'/core/init.php';

/*$user = new User();
//$/user->logout(Config::get('session/session_name_3'));

if($user->logout(config::get('session/session_name_1'))){
    Redirect::to('index.php');
}else if($user->logout(config::get('session/session_name_2'))){
    Redirect::to('index.php');
}else if($user->logout(config::get('session/session_name_4'))){
    Redirect::to('index.php');
}else if($user->logout(config::get('session/session_name_3'))){
    Redirect::to('index.php');
}*/ 

Session_destroy();

Redirect::to('index.php');
?>

```

Figure 39 system logout codes

Forget password and password recovery

```
<?php
class Mail {

    //this function is for sending mail
    public static function send_mail($email, $subject, $message){
        require_once ('mails/mail/class.phpmailer.php');
        require_once ('mails/mail/class.smtp.php');

        $mail = new PHPMailer();
        $mail->IsSMTP();
        $mail->SMTPDebug = false;
        $mail->SMTPAuth = true;
        $mail->SMTPSecure = "ssl";
        $mail->Host = 'smtp.gmail.com';
        $mail->Port = 465;
        // $mail->SMTPOAuth = true;
        $mail->AddAddress($email);
        $mail->Username = "mhsporsho@gmail.com";
        $mail->Password = "thisismygmailaccount"; //should be change
        $mail->SetFrom('mhsporsho@gmail.com','EC_Admin');
        $mail->AddReplyTo('mhsporsho@gmail.com','EC_Admin');
        $mail->Subject = $subject;
        $mail->MsgHTML($message);

        if($mail->Send()){
            echo "<div id='zartan_success_message'>";
            echo "A mail has sent to your email, Please check your email for further process";
            echo "</div>";
        }else {
        }
    }
}
```

Figure 40 password recovery processing code

```
if(Token::check(Input::get('token'))){
    if(Input::exists()){
        $validate = new validate();
        $validation = $validate->check($_POST, array(
            'code' => array('required' => true),
            'new-password' => array(
                'required' => true,
                'minimum' => 6
            ),
            'confirm-password' => array(
                'required' => true,
                'matches' => 'new-password'
            ),
        ));
    }
    if($validation->passed()){
        $code = Input::get('code');
        $salt = Hash::salt(32);
        // $password = Hash::make(Input::get('new-password'),$salt);
        $password = Hash::make(Input::get('new-password'),$salt);

        $data = DB::getInstance()->query("UPDATE ec_coo SET coo_password = '$password', salt= '$salt' WHERE code = ? ",array($code));
        if($data){
            Session::delete('password_code');
            Session::flash('forgot_password','Password has been updated successfully, login now');
            Redirect::to('index.php');
        }else {
            echo "not updated";
        }
    }else {
        echo "<div id='zartan_login_error_message'>";
        foreach ($validation->errors() as $error) {
            echo $error . '<br>';
        }
        echo "</div>";
    }
}
>
```

Figure 41 password recovery success code

Student claim

```

$user = DB::getInstance()->insert('claim',array(
    'module_title' => Input::get('module_name'),
    'assessment'   => Input::get('assessment_type'),
    'ec_type'      => Input::get('EC_type'),
    'claim_details' => Input::get('claim'),
    'claim_date'   => date('Y-m-d'),
    'last_date'    => date('Y-m-d', strtotime("+14 days")),
    'std_id'        => $user->data()->std_id,
));

if($user){

    Session::put('claim_success','Claim submitted successfully , please upload evidence!');
    Redirect::to('evidence.php');

} else {
    echo "<div class='row'>";
    echo "<div class='col-lg-8 col-lg-offset-2'>";
    echo "<div id='zartan_error_message'>";
    echo "sorry,problem to claiming";
    echo "</div>";
    echo "</div>";
    echo "</div>";
}

} else {
    echo "<div class='row'>";
    echo "<div class='col-lg-9 col-lg-offset-1'>";
    echo "<div id='zartan_error_message'>";
    foreach ($validation->errors() as $error) {
        echo $error.'<br>';
    }
    echo "</div>";
    echo "</div>";
    echo "</div>";
}

```

Figure 42 Student claim table and claim submission option

```

$user = new User();
if($user->isLoggedIn()){
    $std_id = $user->data()->std_id;

    $data = DB::getInstance()->query("
        SELECT
            count(cl.claim_feedback) as decision,
            s.std_id
        FROM
            student as s
        INNER join
            claim as cl
        ON
            s.std_id = cl.std_id
        where cl.std_id = $std_id and cl.claim_feedback = 'accepted'
    ");

    foreach ($data->results() as $data) {
        echo $data->decision;
    }

}

```

Figure 43 Claim accepted notification in student panel

```

$user = new User();
if($user->isLoggedIn()){
    $std_id = $user->data()->std_id;
    $data = DB::getInstance()->query("
        SELECT
            count(cl.claim_feedback) as decision,
            s.std_id
        FROM
            student as s
        INNER JOIN
            claim as cl
        ON
            s.std_id = cl.std_id
        where cl.std_id = '$std_id' and cl.claim_feedback = 'rejected'
    ");
    foreach ($data->results() as $data) {
        echo $data->decision;
    }
}

```

Figure 44 Claim rejected notification in student panel

Coordinator's action on Student claim

```

<?php
if($user->isLoggedIn()){
    $user = new Staff();
    if($user->hasPermission('coordinator')){
        $fac_name = $user->data()->fac_name;
        $data = DB::getInstance()->query("SELECT
            count(cl.claim_feedback) as pending,
            cl.claim_feedback
        FROM
            ec_coo AS c
        INNER JOIN
            department AS d ON c.fac_name = d.fac_name
        INNER JOIN
            module AS m ON m.dep_name = d.dep_name
        INNER JOIN
            claim AS cl ON cl.module_title = m.mod_title
        where d.fac_name = '$fac_name' and cl.claim_feedback = 'not
            accepted' or d.fac_name = '$fac_name' and cl.claim_feedback =
            'processing'
    ");
    foreach ($data->results() as $data) {
        echo "<span style='color:red'>$data->pending.</span>";
    }
}
?> </a>

```

Figure 45 Notification appears on coordinators panel for any new claim

```

$data = DB::getInstance() ->query("SELECT c.co_name, c.fac_name, d.dep_name, m.mod_title,
cl.claim_no, cl.claim_details, cl.ec_type, cl.module_title,
cl.assessment, cl.claim_date, cl.evidence_details, cl.claim_feedback, cl.std_id
from ec_coo as c
INNER join department as d
on c.fac_name = d.fac_name
INNER join module as m
on m.dep_name = d.dep_name
inner join claim as cl
on cl.module_title = m.mod_title
where c.fac_name = ? order by cl.claim_no desc limit $start, $perPage",array("$fac_name"));

if(!$data->count()){
    echo "<span style='color:red;'>There is no claim in this Faculty</span>";
} else {
?>
<thead>
|View</a>

|  |

```

Figure 46 Claim list on coordinator's panel

```

<?php
// $claim_no = Input::get('cd');

$data = DB::getInstance()->query("SELECT
    c.coo_name,
    c.fac_name,
    d.dep_name,
    m.mod_title,
    cl.claim_no,
    cl.file_name,
    cl.file_path,
    cl.std_id
  FROM
    ec_coo AS c
  INNER JOIN
    department AS d ON c.fac_name = d.fac_name
  INNER JOIN
    module AS m ON m.dep_name = d.dep_name
  INNER JOIN
    claim AS cl ON cl.module_title = m.mod_title
  WHERE
    cl.claim_no = ? AND c.fac_name = ?",
    array("$filterid", "$fac_name"));

foreach ($data->results() as $data) {
    $a= $data->file_path;
    $b = explode('/', $a);

    if(in_array('pdf', $b)){
        echo "<div class='alert alert-danger'>";
        echo "<img src='../images/Graphicloads-Filetype-Pdf.ico' width=25 height=25>" . ' ' . $data->file_name;
        echo "</div>";
    }
    <a href="download.php?cd=<?php echo $data->claim_no?>"><button class="btn btn-warning">Download</button></a>
    <?php
}

```

```

foreach ($data->results() as $data) {
    $a= $data->file_path;
    $b = explode('/', $a);

    if(in_array('pdf', $b)){
        echo "<div class='alert alert-danger'>";
        echo "<img src='../images/Graphicloads-Filetype-Pdf.ico' width=25 height=25>" . ' ' . $data->file_name;
        echo "</div>";
    }
    <a href="download.php?cd=<?php echo $data->claim_no?>"><button class="btn btn-warning">Download</button></a>
    <?php
} else {
    >
    <div class="thumbnail">
    <?php
    echo "<img src='../std/$data->file_path' height= 300 width= 280 class='img-rounded' alt='No Evidence'>";
    >
    </div>
    <?php
}
}

    >
<?php
}
?>

```

Figure 47 Coordinator's decision options on student claims

Administrative tasks

```
$user = DB::getInstance()->insert('assessment',array(
    'dep_name' => Input::get('department_name'),
    'assess_type' => Input::get('assessment_type'),
    'assess_due_date' => Input::get('due_date'),
    'assess_final_date' => Input::get('final_date'),
    'mod_title' => Input::get('module_name'),
    'semister_id' => Input::get('semister_name'),

));
if($user){
    echo "<div class='row'>";
    echo "<div class='col-lg-8 col-lg-offset-2'>";
    echo "<div id='zartan_success_message'>";
    echo "Data insert successfully";
    echo "</div>";
    echo "</div>";
    echo "</div>";

} else {
    echo "<div class='row'>";
    echo "<div class='col-lg-8 col-lg-offset-2'>";
    echo "<div id='zartan_error_message'>";
    echo "There is problem to insert data";
    echo "</div>";
    echo "</div>";
    echo "</div>";
}
else {
    echo "<div class='row'>";
    echo "<div class='col-lg-9 col-lg-offset-1'>";
    echo "<div id='zartan_assessment_error_messages'>";
    foreach ($validation->errors() as $error) {
        echo $error.'<br>';
    }
}
```

Figure 48 Admin can fix EC claim deadline for individual faculties

```
//insert data
DB::getInstance()->query("ALTER TABLE semister AUTO_INCREMENT = 1");

$user = DB::getInstance()->insert('semister',array(
    'semister_name' => Input::get('semister_name'),
    'closure_date' => Input::get('closure_date')
));

if($user){
    //Session::delete('semister_delete');
    Session::delete('semister_update');
    echo "<div class='row'>";
    echo "<div class='col-lg-8 col-lg-offset-2'>";
    echo "<div id='zartan_success_message'>";
    echo "Data insert successfully";
    echo "</div>";
    echo "</div>";
    echo "</div>";

} else {
    echo "<div class='row'>";
    echo "<div class='col-lg-7 col-lg-offset-3'>";
    echo "<div id='zartan_error_message'>";
    echo "This data already exists, Please Update this data";
    echo "</div>";
    echo "</div>";
    echo "</div>";
}
else {
    echo "<div class='row'>";
    echo "<div class='col-lg-8 col-lg-offset-2'>";
    echo "<div id='zartan_error_message'>";
    foreach ($validation->errors() as $error) {
        echo $error.'<br>';
    }
    echo "</div>";
    echo "</div>";
    echo "</div>";
}
```

Figure 49 admin can fix EC related evidence submission deadline for individual faculties

```

$salt = Hash::salt(32);

$user = DB::getInstance()->insert('ec_coo', array(
    'coo_name' => Input::get('name'),
    'coo_email' => Input::get('email'),
    'coo_password' => Hash::make(Input::get('password'), $salt),
    'salt' => $salt,
    'fac_name' => Input::get('faculty'),
    'group' => '4'
));

if($user){

    echo "<div class='row'>";
    echo "<div class='col-lg-8 col-lg-offset-2'>";
    echo "<div id='zartan_success_message'>";
    echo "Coordinator added successfully";
    echo "</div>";
    echo "</div>";
    echo "</div>";

} else {
    echo "<div class='row'>";
    echo "<div class='col-lg-7 col-lg-offset-2'>";
    echo "<div id='zartan_error_message'>";
    echo "This data already exists, Please Update this data";
    echo "</div>";
    echo "</div>";
    echo "</div>";
}

else {
    echo "<div class='row'>";
    echo "<div class='col-lg-9 col-lg-offset-1'>";
    echo "<div id='zartan_error_message'>";
    foreach ($validation->errors() as $error) {
        echo $error.'<br>';
    }
    echo "</div>".
```

Figure 50 Admin can add delete or even modify coordinators information

EC manager's activities

```
$data = DB::getInstance()->query("SELECT
    c.coo_name,
    c.fac_name,
    d.dep_name,
    m.mod_title,
    count(cl.claim_no) as total_claim,
    cl.claim_date,
    count(DISTINCT cl.std_id) as total_student
FROM
    ec_coo AS c
INNER JOIN
    department AS d ON c.fac_name = d.fac_name
INNER JOIN
    module AS m ON m.dep_name = d.dep_name
INNER JOIN
    claim AS cl ON cl.module_title = m.mod_title
where   c.fac_name = ? and Year(cl.claim_date) = ? ",array("$faculty","$year"));

foreach ($data->results() as $data) {

    //Session::delete('semister_delete');
    ?>
    <div class="panel panel-success">
        <div class="panel-heading">
            <div class="row">
                <div class="col-xs-12 text-left">
                    <div class="huge"><span style="font-size: 15px;">In <?php echo "<h4
style='display:inline-block;'>" $year "</h4>"?> total students who already claim in <?php
echo $faculty;?> Faculty</span> - <?php echo $data->total_student?></div>
                </div>
            </div>
        </div>
    </div>
```

Figure 51 EC panel can produce report on total number of students claimed EC

```
$data = DB::getInstance()->query("SELECT
    c.coo_name,
    c.fac_name,
    d.dep_name,
    m.mod_title,
    count(cl.claim_no) as total_claim,
    cl.claim_date
FROM
    ec_coo AS c
INNER JOIN
    department AS d ON c.fac_name = d.fac_name
INNER JOIN
    module AS m ON m.dep_name = d.dep_name
INNER JOIN
    claim AS cl ON cl.module_title = m.mod_title
where   c.fac_name = ? and Year(cl.claim_date) = ? ",array("$faculty","$year"));

foreach ($data->results() as $data) {

    //Session::delete('semister_delete');
    ?>
    <div class="panel panel-warning">
        <div class="panel-heading">
            <div class="row">
                <div class="col-xs-12 text-left">
                    <div class="huge"><span style="font-size: 20px;">In <?php echo "<h4
style='display:inline-block;'>" $year "</h4>"?> total Claim in <?php echo $faculty;?>
Faculty</span> - <?php echo $data->total_claim?></div>
                </div>
            </div>
        </div>
    </div>
```

Figure 52 EC manager can see report of total number of EC claimed in each faculty

```

$data = DB::getInstance()->query("SELECT
    COUNT(cl.claim_no) AS total_claim, count(distinct cl.std_id) as total_student,
    (
        SELECT
        COUNT(a.assess_type)
        FROM
        assessment AS a
        LEFT JOIN
        department AS dp ON dp.dep_name = a.dep_name
        LEFT JOIN
        claim AS clm ON a.mod_title = clm.module_title
        WHERE
        dp.fac_name = '$faculty'
    ) AS total_assessment
FROM
    claim AS cl
INNER JOIN
    module AS m ON cl.module_title = m.mod_title
INNER JOIN
    department AS d ON d.dep_name = m.dep_name
WHERE
    d.fac_name = '$faculty' AND YEAR(cl.claim_date) = '$year'");

foreach ($data->results() as $data) {

    //Session::delete('semister_delete');
    ?>

    <div class="panel panel-warning">
        <div class="panel-heading">
            <div class="row">

                <div class="col-xs-12 text-left">
                    <?php
                    echo "Total assessment- ".$data->total_assessment."<br>";
                    echo "Total claim- ".$data->total_claim."<br>";
                    echo "Total student- ".$data->total_student;

                    ?>
                    <div class="huge"><span style="font-size: 18px;">In <?php echo "<h4 style='display:inline-block;'>$year </h4>">
                    >> total Percentage of Claim in <?php echo $faculty;?> Faculty</span> - <?php

                    $total_assessment = $data->total_assessment;
                    $total_claim = $data->total_claim;
                    $total_student = $data->total_student;

                    if($total_assessment == 0){
                        echo "0 %";
                    }else {
                        $claim_percentage = $total_claim/$total_assessment*100;
                        echo $total = number_format($claim_percentage,2)."%";
                    }
                </div>
            </div>
        </div>
    </div>
}

```

Figure 53 this codes produced EC report in a form of percentage for Manager

2. Sprint backlog testing

This particular testing has been performed to ensure the system development followed all the agile norms and processes. Each development followed sprint backlog. It has been tested that the system development went parallel with the minute of meetings, sprint backlog.

(softwaretestinghelp, n.d.)

Test log

Test no.	Test name	Description	Test outcome	Result	Interpretation
1	Responsive	Tested on design responsiveness on various screen resolutions	Successfully overcome the test criteria	Passed	
2	Compatibility	Tested on system compatibility on various OS platforms	Test result came successful	Passed	
3	Cross browsing	Systems are played on various browsers	Successfully tested	Passed	
4	Security	Different security testing have been conducted	Various test has been tested successfully	Passed	
5	Usability	Usability testing has been successfully performed	Usability testing has been thoroughly tested	Passed	
6	Database	Testing is conducted aspects of database	Tested repeatedly with success	Passed	
7	Functionality	Functionalities of the system has been successfully tested	Successfully tested	Passed	
8	Performance	Online tools has been tried to create basic test scenario	Issues raised	Failed	Web load and stress test could be conducted if the system could be live.
9	Integration	System integration has been tested	Test completed successfully	Passed	
10	Validation	Proper validation testing has been	Test conducted successfully	Passed	

		conducted and demonstrated			
11	Unit	Most of the major system units have been tested individually	Most of the functional units are successfully tested	Passed	With the help of the developer, each unit is separately evaluated for testing.
12	Sprint testing	System development in parallel with the agile processes are tested	Successfully monitored and tested accordingly	Passed	

Appendix-B

Scrum processes

Sprint backlog

Sprint backlog (Team Zartan)

Serial no.	Product backlog	Unit task	Sprint duration	Status
1	4	Database table creation	2	Done
		UI design artifact	1	Done
		Coding	1	Done
		Testing	1	Done
2	5	Analyze admin panel	1	Done
		Data table link	1	Done
		Coding		
		Testing	1	Done
3	2	Admin panel design	1	Done
		DB table linking	1	Done
		Coding	1	Done
		Testing	1	Done
4	3	Design coordinator panel	1	Done
		Linking db table	1	Done
		Coding	1	Done
		Testing	1	Done
5	6	Coordinator ec functionalities	1	Done
		UI design		
		Coding	2	Done
		Testing	1	Done
6	7	Design Student panel	1	Done

		Db linking		
		coding	1	Done
		testing	1	Done
7	8	Student table management	1	Done
		Analyze design artifact	1	Done
		coding	1	Done
		testing	1	Done
8	9	manager panel design	1	Done
		Analyze panel functionalities		
		coding	1	Done
		testing	1	Done
9	10	Design manager report generator design	2	Done
		coding	1	Done
		testing	1	Done
10	1	Final bug fixing	1	Done
		Overall testing	1	Done
		Final group assessment	1	Done
		Final evaluation		

Minute of meeting

Minutes of meeting (Team Zartan)

Meeting no. 01			
Meeting title:	Individual Roles, Meeting times and Schedules, user story, methodology		
Meeting date:	15/02/2017	Meeting Time:	4:00pm
		Meeting location:	Daffodil International Academy
Objectives:	Fixing individual roles, daily meeting times, schedules and discuss user story and methodology that need to be used		
Participants:	A.S.M Minhazul hoque, Mohiminul Hossain, Kazi Al-Amin, Omaer ahmed		
Meeting outcomes:	Role of each members and regular meeting schedules will be fixed. Understand user story and methodology confirmed.		

Meeting no. 02			
Meeting title:	In depth data analysis, Product backlog		
Meeting date:	20/02/2017	Meeting Time:	3:30pm
		Meeting location:	Daffodil International Academy
Objectives:	Discussion on task analysis in depth, brief discussion on producing product back log		
Participants:	A.S.M Minhazul hoque, Mohiminul Hossain, Kazi Al-Amin, Omaer ahmed		
Meeting outcomes:	In depth data analysis on the individual tasks. Rough product backlog has been announced		

Meeting no. 03			
Meeting title:	Completed Product backlog, Task priority list has been made		
Meeting date:	22/02/2017	Meeting Time:	4:00pm
		Meeting location:	Daffodil International Academy
Objectives:	Finalize Initial product backlog, task priority list.		
Participants:	A.S.M Minhazul hoque, Mohiminul Hossain, Kazi Al-Amin		
Meeting outcomes:	Finalizing initial product backlog, task priority list has been made.		

Meeting no. 04			
Meeting title:	Discussion on database, class diagrams and ERD		
Meeting date:	27/02/2017	Meeting Time:	3:30pm
		Meeting location:	Daffodil International Academy
Objectives:	Discussion of class diagram that has been produced and rough work on ERD diagram.		
Participants:	A.S.M Minhazul hoque, Mohiminul Hossain, Kazi Al-Amin		
Meeting outcomes:	Group review and finalization of system class diagram. Rough discussion on ERD diagram		

Meeting no. 05			
Meeting title:	Discussion on sprint backlog, produced class diagram and detail ERD		
Meeting date:	02/03/2017	Meeting Time:	3:30pm
		Meeting location:	Daffodil International Academy
Objectives:	Discussion of class diagram that has been produced and rough work on ERD diagram.		
Participants:	A.S.M Minhazul hoque, Mohiminul Hossain, Kazi Al-Amin		
Meeting outcomes:	Group review and finalization of system class diagram. Rough discussion on ERD diagram		

Meeting no. 06			
Meeting title: Completed sprint backlog, design artifact, finalizing database, coding			
Meeting date:	5/03/2017	Meeting Time:	4:00pm
		Meeting location:	Daffodil International Academy
Objectives:	Present sprint backlog, final database, start designing artifact for each model, start coding .		
Participants:	A.S.M Minhazul hoque, Mohiminul Hossain, Kazi Al-Amin, Omaer ahmed		
Meeting outcomes:	Sprint backlog for the group assignment, database finalized, design artifact has been started, coding has been started.		

Meeting no. 07			
Meeting title: Discussion on progress rate, overview agile processes			
Meeting date:	09/03/2017	Meeting Time:	4:00pm
		Meeting location:	Daffodil International Academy
Objectives:	Overall progress review, discussion whether the process is running in parallel with agile processes		
Participants:	A.S.M Minhazul hoque, Mohiminul Hossain, Kazi Al-Amin, Omaer ahmed		
Meeting outcomes:	Discussed on current progress, process monitored under agile framework.		

Meeting no. 08			
Meeting title: Code bug discussion and ways to overcome			
Meeting date:	13/03/2017	Meeting Time:	4:00pm
		Meeting location:	Daffodil International Academy
Objectives:	Discussion on problems raised during coding, further issues came up during developments		
Participants:	A.S.M Minhazul hoque, Mohiminul Hossain, Kazi Al-Amin, Omaer ahmed		
Meeting outcomes:	Bugs have been discussed; ways of solution has been discusses, further issues raised have been evaluated and sorted out.		

Meeting no. 09			
Meeting title:	Further bug issues		
Meeting date:	16/03/2017	Meeting Time:	4:00pm
		Meeting location:	Daffodil International Academy
Objectives:	Bug issue raised, group discussion on that problem, review solution		
Participants:	A.S.M Minhazul hoque, Mohiminul Hossain, Kazi Al-Amin, Omaer ahmed		
Meeting outcomes:	Solution to fix the bug has been found, testing have been started according to test logs and test plans		

Meeting no. 10			
Meeting title:	Refining on design, colors, further coding defects, testing		
Meeting date:	19/03/2017	Meeting Time:	4:00pm
		Meeting location:	Daffodil International Academy
Objectives:	Concentration on front end usage experiences, further coding deficiencies are identified and focused		
Participants:	A.S.M Minhazul hoque, Mohiminul Hossain, Kazi Al-Amin, Omaer ahmed		
Meeting outcomes:	Front end design refining, solving coding bugs, testings		

Meeting no. 11			
Meeting title:	Design interface, bug fixing		
Meeting date:	23/03/2017	Meeting Time:	3:00pm
		Meeting location:	Daffodil International Academy
Objectives:	Solve interface issues and bug fixing on functionalities		
Participants:	A.S.M Minhazul hoque, Mohiminul Hossain, Kazi Al-Amin, Omaer ahmed		
Meeting outcomes:	System UI improves, More coding issues arises		

Meeting no. 12			
Meeting title:	Discussion about Additional features and functionalities		
Meeting date:	27/03/2017	Meeting Time:	3:00pm
		Meeting location:	Daffodil International Academy
Objectives:	Adding additional components such as paginations, few instruction statements		
Participants:	A.S.M Minhazul hoque, Mohiminul Hossain, Kazi Al-Amin, Omaer ahmed		
Meeting outcomes:	Coder and system UI designer is working on this matter.		

Meeting no. 13			
Meeting title:	Discussion on overall progress		
Meeting date:	01/04/2017	Meeting Time:	5:00pm
		Meeting location:	Daffodil International Academy
Objectives:	Overall progress discussion, deficiencies, limitations, and achievements		
Participants:	A.S.M Minhazul hoque, Mohiminul Hossain, Kazi Al-Amin, Omaer ahmed		
Meeting outcomes:	Over all system evaluation, team limitation and achievements		

Meeting no. 14			
Meeting title:	Agile documentations		
Meeting date:	06/04/2017	Meeting Time:	3:00pm
		Meeting location:	Daffodil International Academy
Objectives:	Agile documents, discussion on whether product backlog, sprint backlog etcetera are going according to the actual system progress.		
Participants:	A.S.M Minhazul hoque, Mohiminul Hossain, Kazi Al-Amin, Omaer ahmed		
Meeting outcomes:	Group evaluation of agile process in parallel with the system developments.		

Meeting no. 15			
Meeting title: Final meeting, last minute optimizations of system			
Meeting date:	09/04/2017	Meeting Time:	3:00pm
		Meeting location:	Daffodil International Academy
Objectives:	Final improvements, fixation of minor issues, and UI on the basis of repeated testing.		
Participants:	A.S.M Minhazul hoque, Mohiminul Hossain, Kazi Al-Amin, Omaer ahmed		
Meeting outcomes:	Final touch on system codes and agile documentations. Last group meetings.		

Bibliography

am i responsive design, [Online], Available: <http://ami.responsivedesign.is/> [01 April 2017].

softwaretestinghelp, [Online], Available: <http://www.softwaretestinghelp.com/web-application-testing/> [14 march 2017].