9. Appendix

9.1. Minutes of meetings

The below table displays the meetings that took place during the month. It is a summary of what actions took place in the meeting. In addition, the table is divided corresponding to the sprints that took place.

Date	Duration	Main Issue	Agenda	Attendees
03/03/21	30min	Introduction	- Read through coursework - Noted down requirements which was to be prioritized and presented in next meeting	SM, MK NK, PC
04/03/21	50min	Requirements	- The prioritized requirements were discussed - Noted down high requirements for product backlog - Decided on who makes the product and sprint backlog	SM, MK NK, PC
05/03/21	30min	Use Case and Functionality	- users were identified - roles for each user were discussed - use case diagram was created - assigning tasks to each person in group for first sprint - and setting due dates for the tasks	SM MK
06/03/21	30min	Login system	- tested the login system for errors - debated on changing security algorithm	SM, MK NK, PC
09/03/21	30min	Review tasks and discussion	- completed tasks were discussed	SM, MK NK, PC

			- remaining tasks left for sprint were pushed to new sprint	
			- assigned tasks for new sprint	
12/03/21	30min	Functionality	- Tested student upload for document and image type error	SM, MK
			- touch on how comment will be coded	
14/03/21	30min	New sprint discussion	- assigned tasks for new sprint	SM, MK NK, PC
			- discussion on how to improve group work and be punctual	
18/03/21	30min	New sprint discussion	- Completed functions reviewed	SM, MK
			- assign new tasks	
23/03/21	30min	Documentation	- allocated phases to document on to each member	SM, MK NK, PC
25/03/21	30min	Documentation	- evaluated introduction, methodology, analysis phase	SM, MK
28/03/21	30min	Reports	- discussed on how reports for contributions will be implemented	SM, MK, NK
30/03/21	30min	Reports	- tested reports by testing for each faculty and year	SM, MK, NK
01/04/21	30min	Testing	- tested the website checking for errors and buys	SM, MK NK, PC
02/04/21	30min	Check documentation	- review of documentation	SM, MK
			- checked for missing sections	
03/04/21	30min	Screencast	- script writing	SM, MK NK
			- recording in sessions	
			- video editing	

Table 9: Meeting minutes

9.2. User Guide

Group repository:

https://drive.google.com/drive/folders/1esDF8WOVeLq-IaDrEu_DjcRYfE6OPXYJ?usp=sharing

Screencast URL: https://www.youtube.com/watch?v=rE3TrF4euZ8

Website: URL: localhost/campusroots/index.php

For setting up system:

- internet connection is needed

- extract the file campusroots

- copy the file into directory to either wamp, xampp or lamp according to what server is being used in your system

- run the server (wamp, xampp or lamp)

- open and create database magazinedb

- then import the database inside the db folder named magazinedb.sql.

- after database creation the website can be run

- URL for landing page localhost/campusroots/

- all users details that can be used are present in the table below:

User's Email and Password

Admin		Email: nami@gmail.com
		Password: admin0021
Marketing Coordinator	FOIT:	Email: starrybcampusrootit@gmail.com
		Password: @campusroot2001
	FOL:	Email: liamncampusrootl@gmail.com
		Password: @campusroot2001
	FOS:	Email: jeffacampusroots@gmail.com
		Password: @campusroot2001
	FOBA:	Email: sonamcampusrootba@gmail.com
		Password: @campusroot2001

Student	FOIT:	Email: tsukki@gmail.com
		Password: tsukki01
	FOL:	Email: vibha@gmail.com
		Password: vibha03
	FOS:	Email: jmonet@gmail.com
		Password: jmonet37
	FOBA:	Email: lanad@gmail.com
		Password: lanad04
Manager		Email: janesmith@gmail.com
		Password: mmcampusroot

Table 10: User credentials

9.3. Final Detailed Class Diagram

During the sprints the class diagram was updated and the diagram below is the final version:
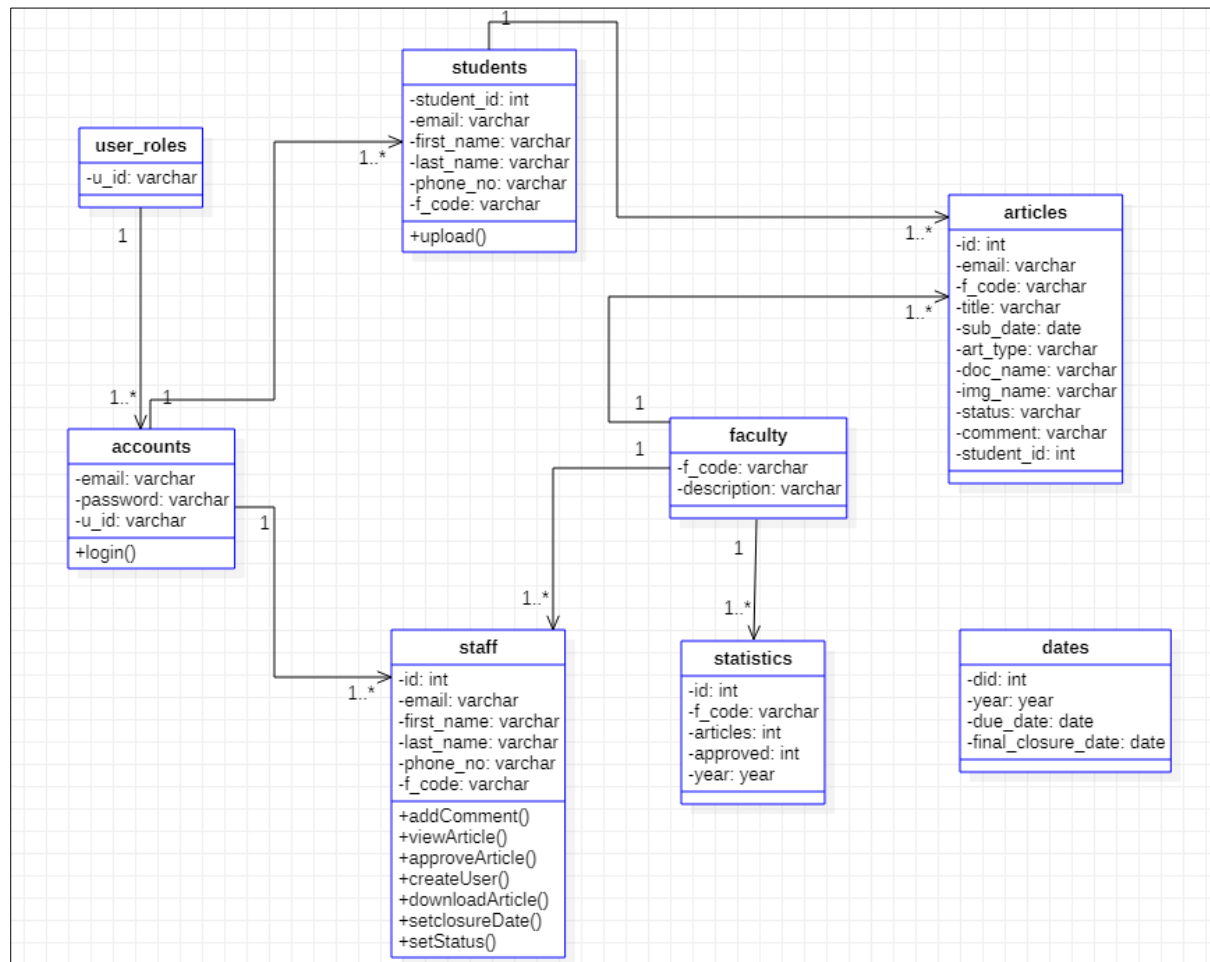
Figure 29: Final detailed class diagram

No.Page: 51

9.4. Sequence Diagrams

Guest Sequence diagram:
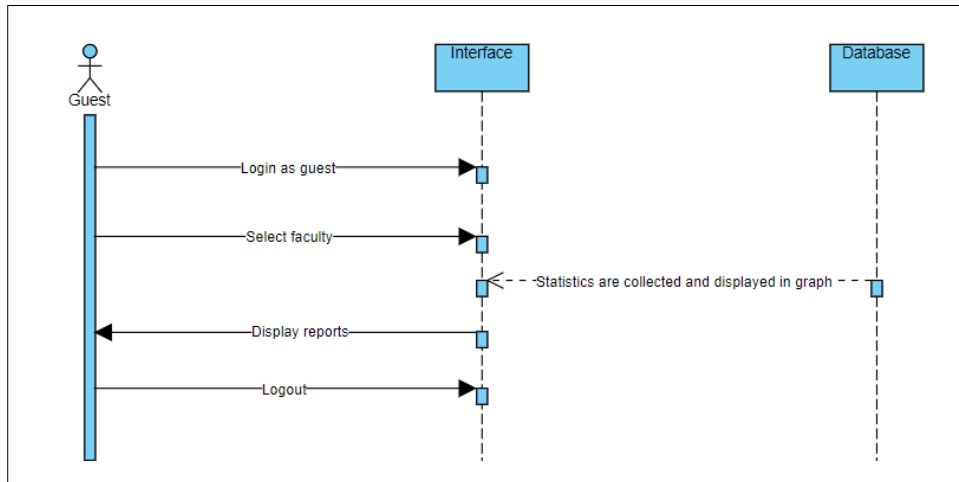
Figure 30: Guest sequence diagram

Marketing Manager Sequence Diagram:
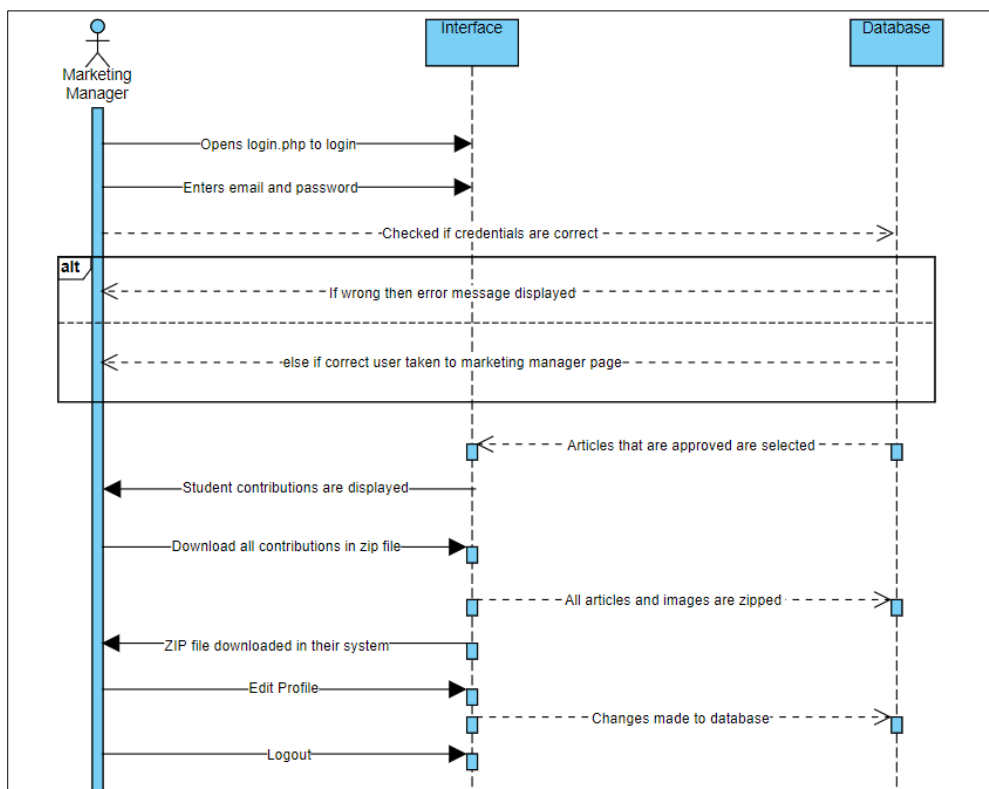
Figure 31: Manager sequence diagram

Admin Sequence Diagram:

No.Page: 52
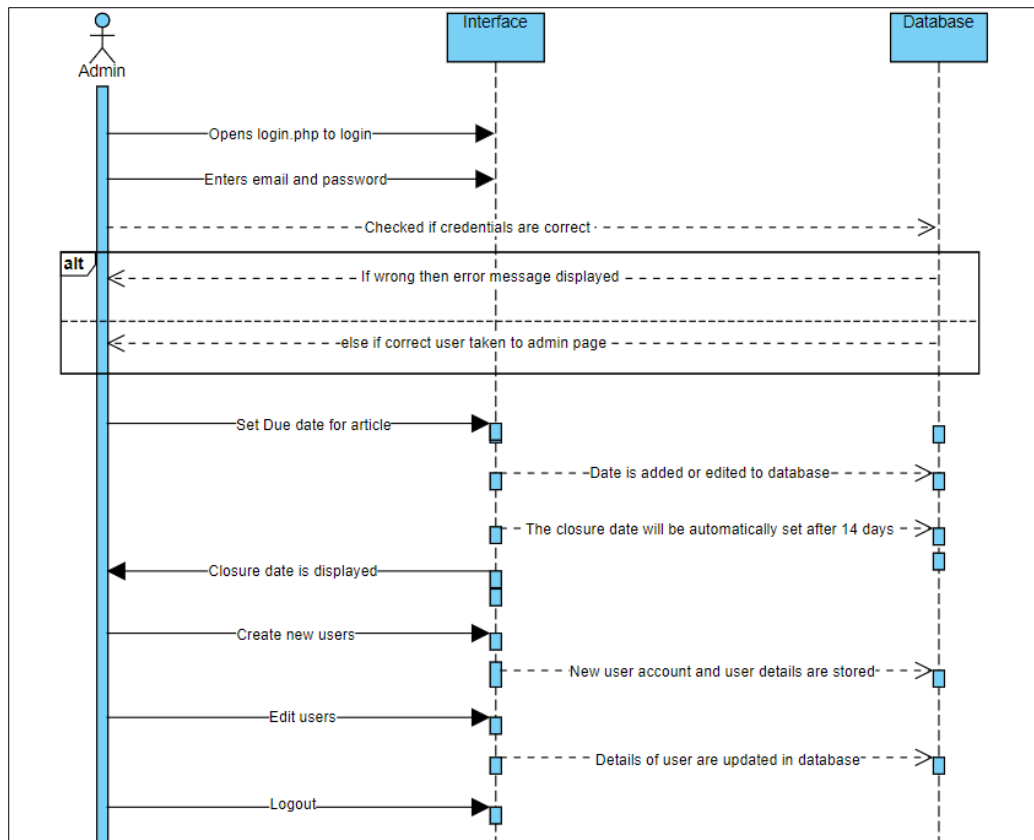
Figure 32: Admin sequence diagram
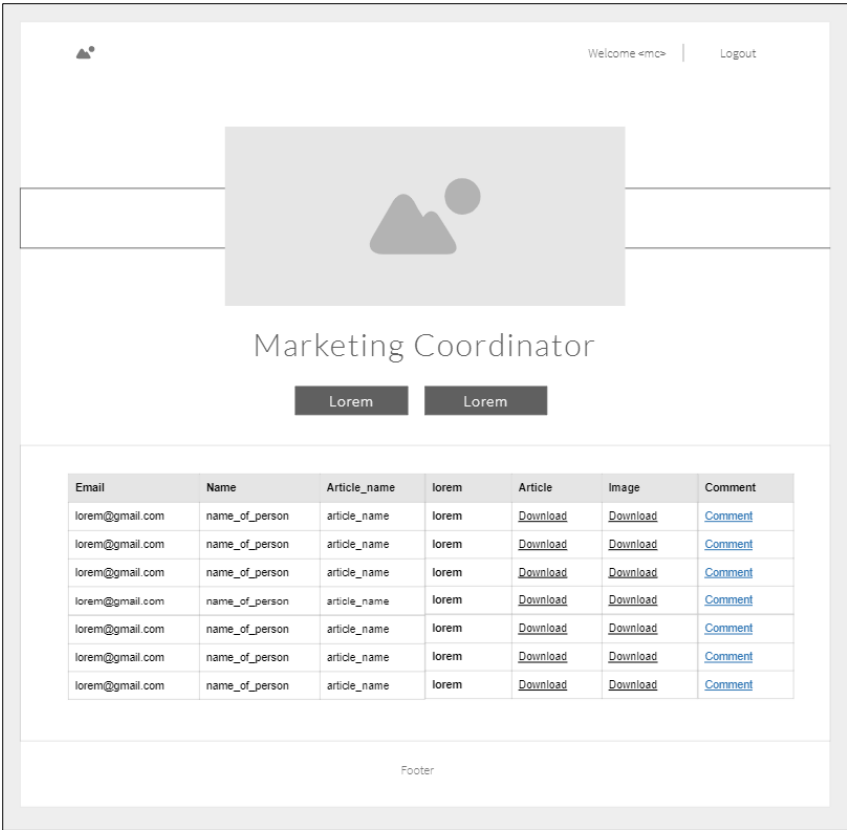
9.5. Wireframes
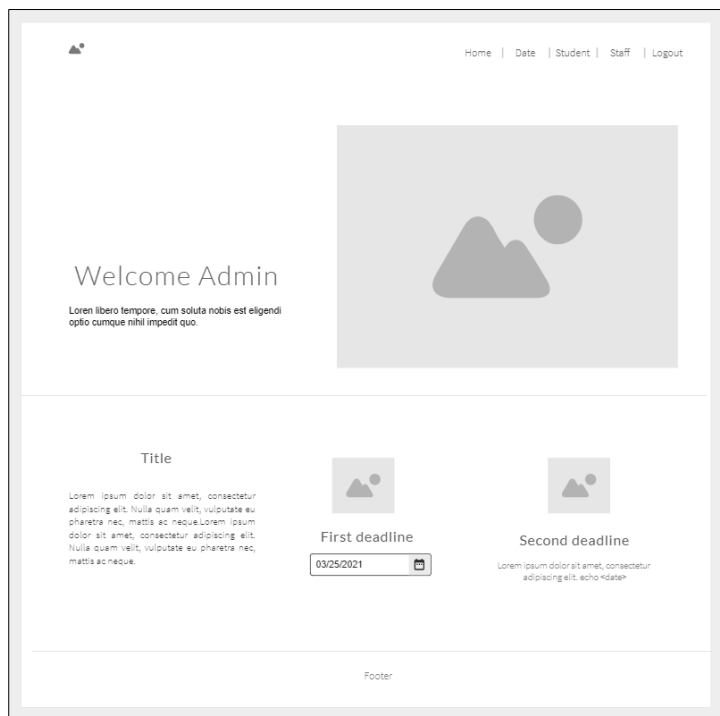
Figure 33: Admin page wireframe

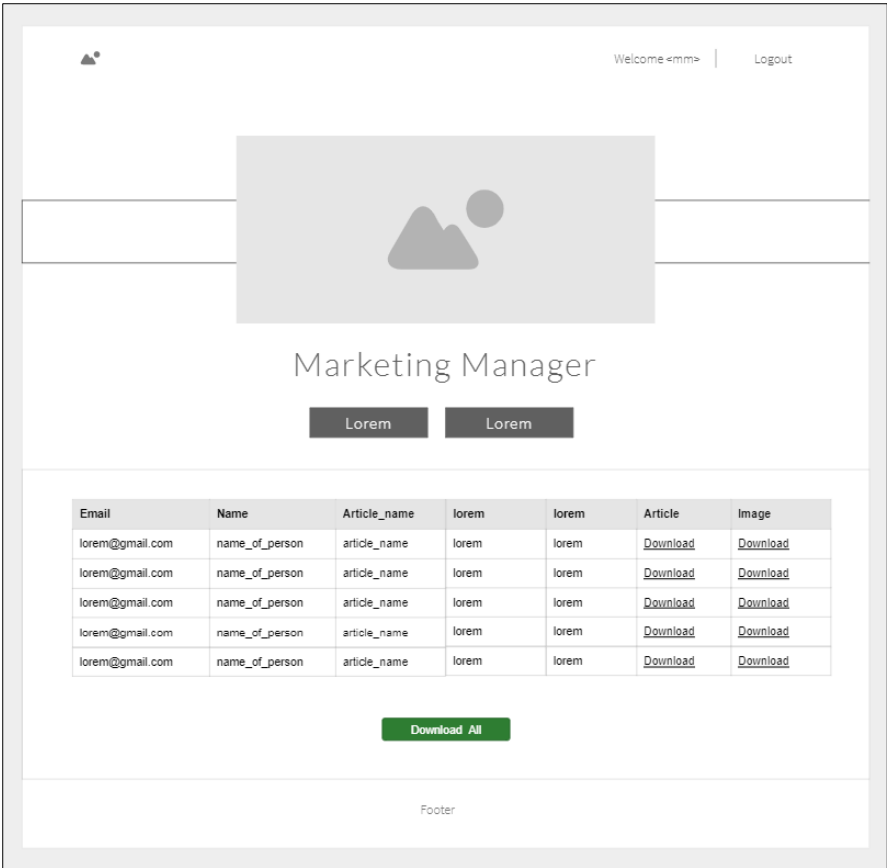*Figure 34: Coordinator page wireframe*

Figure 35: Student article page wireframe
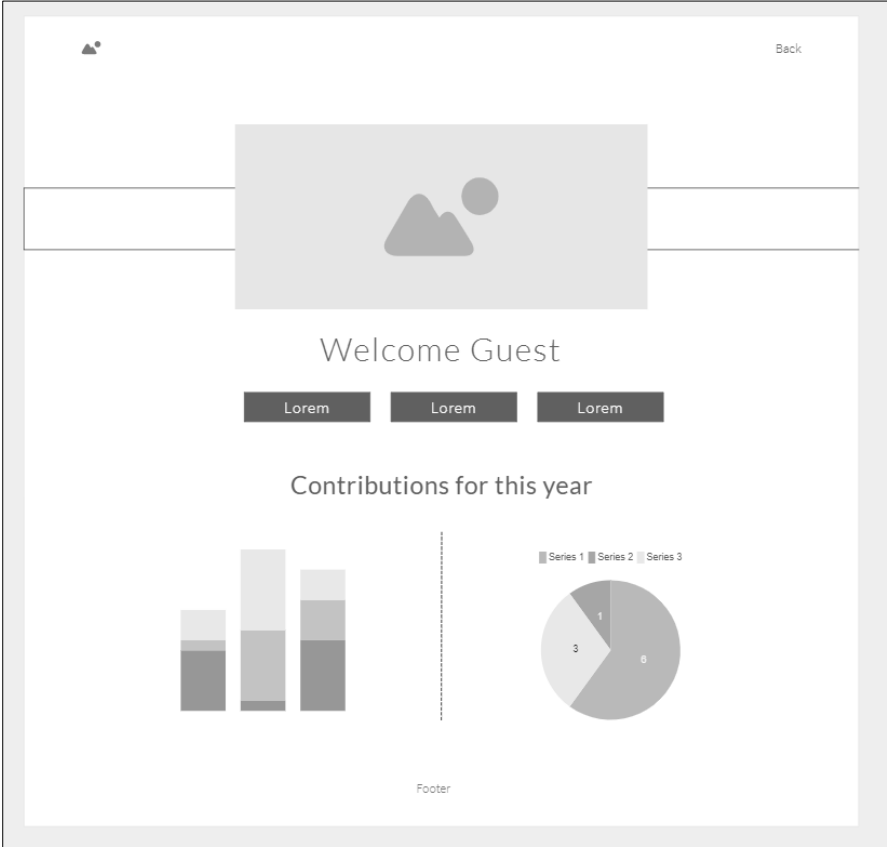
No.Page: 54

Figure 36: Marketing Manager wireframe
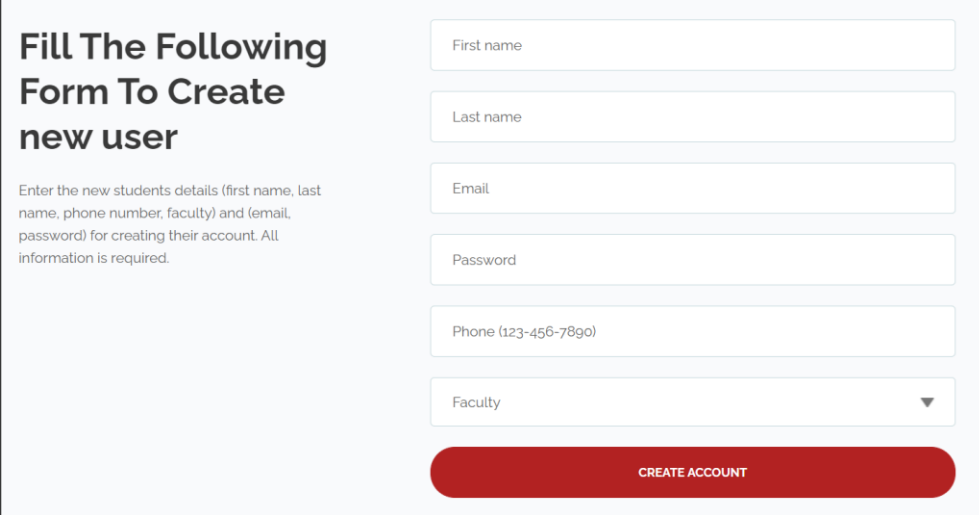
Figure 37: Guest page wireframe

9.6. Test logs Result Evidence

Test 1: Admin can add users

The form below is used for adding new students, if the information is not added than error displays below the text box else the account is created.
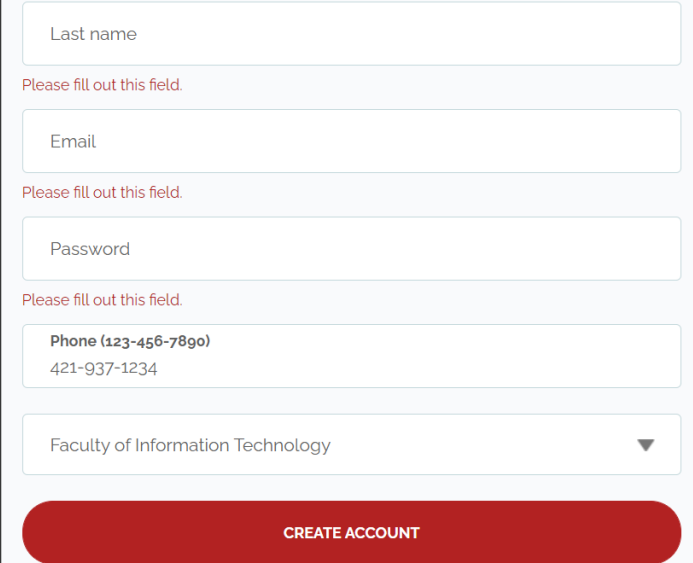
Figure 38: user creation

Figure 39: error message if user does not enter details

Figure 40: if no errors success message displays

Figure 41: evidence in database

The form for creating staff is same except the staff roles drop box and form is used for adding new staff, if the information is not added than error displays below the text box else the account is created.

Figure 42: staff account creation

Figure 43: if no errors success message displays

Figure 44: evidence in database

Test 2: User Login

The users type in email and password, if incorrect error shows or redirects them to respective user role pages. Unlike all other users' guests don't have credentials but can choose faculty.
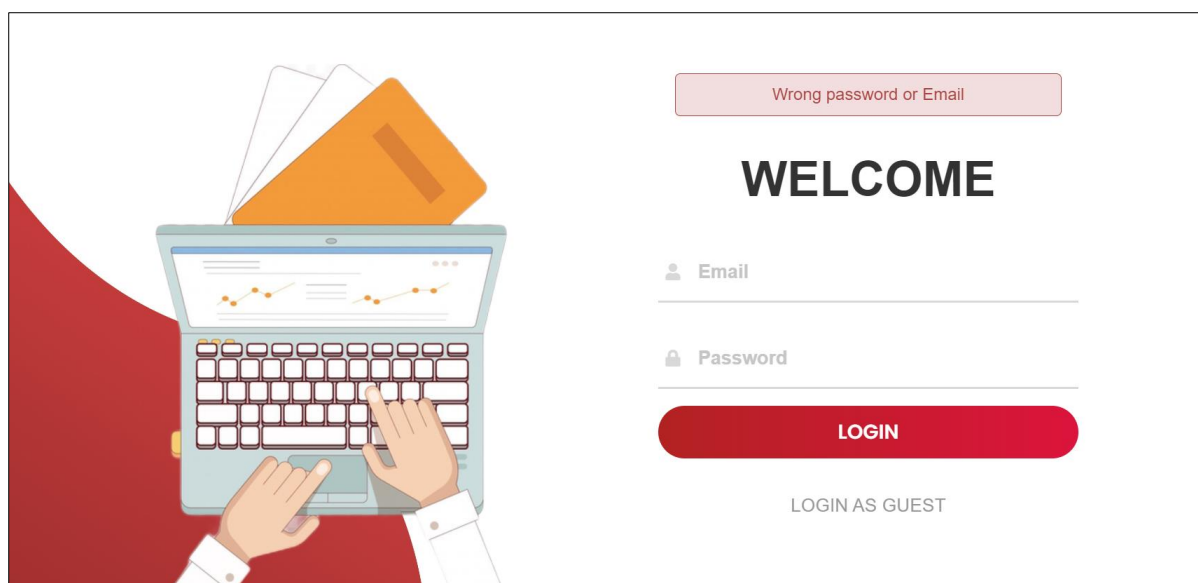
Figure 45: Error message if students do not enter credentials correctly
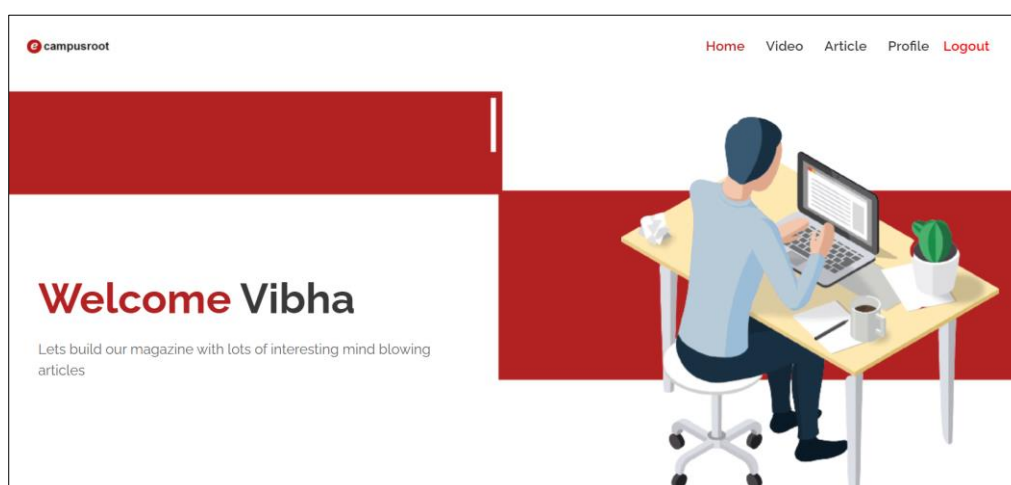

Figure 46: Student login redirected page

Figure 47: Admin login redirected page

Figure 48: Coordinator login redirected page

Figure 49: Manager login redirected page
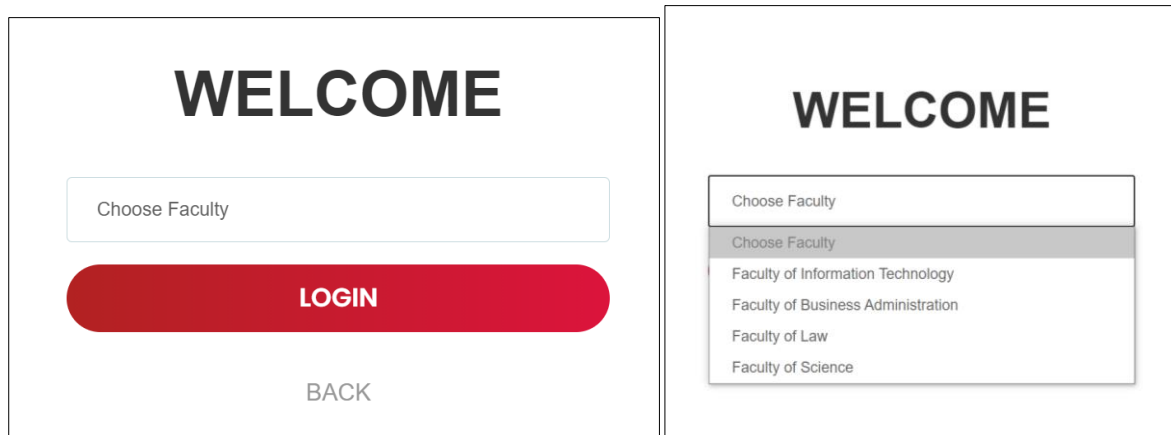
Figure 50:guest login redirected page
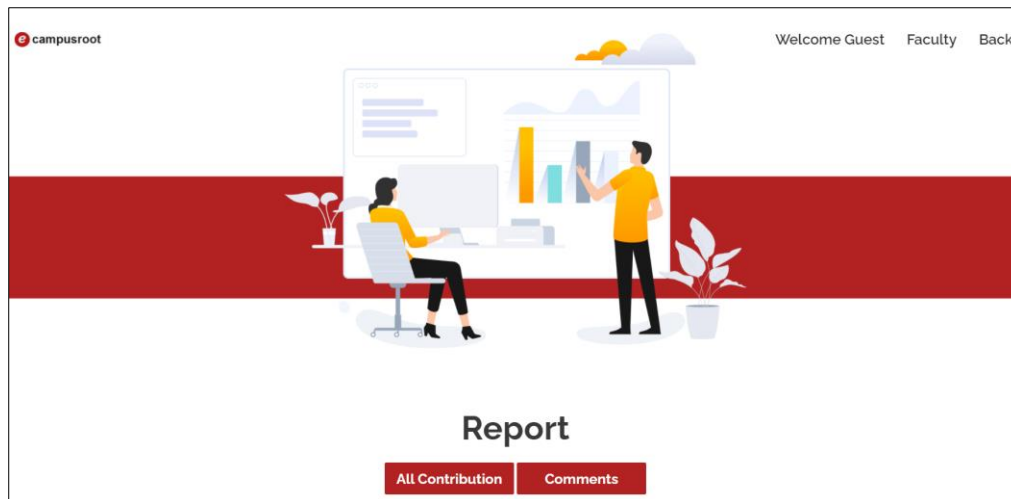
Figure 51: Guest dashboard after selecting faculty

Test 3 and 4: Student uploading contribution and have to accept terms and conditions

The student contribution submission form is as below and when needed information is not present error message, if document and image type are wrong error message and if document already exists error message, if check box for terms not checked error message else if everything is correct contribution is uploaded. And students can make more than one contribution.

Figure 52: Article submission form if no input

Figure 53: article page error if same document is submitted twice

Figure 54: if format is wrong for uploads

Figure 55: if uploads are successful message

Test 5: Coordinator receives email on new student upload

The coordinator of the corresponding student faculty will receive notification about student submission.

Figure 56: if contribution uploads successfully than email send to coordinator of that faculty

Test 5: Admin set closure date

If there is due date already set the admin can edit the date else a new date will be added

Figure 57: shows due date not set and the date selection allows for set a date

If due date is set for example 8th April than the system will automatically calculate 14 days plus for closure date which is second deadline.
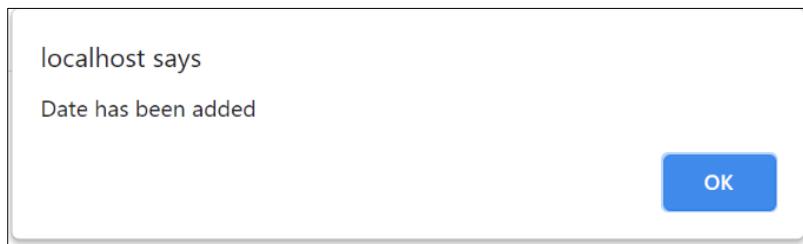
Figure 58: if date added is successful message displays
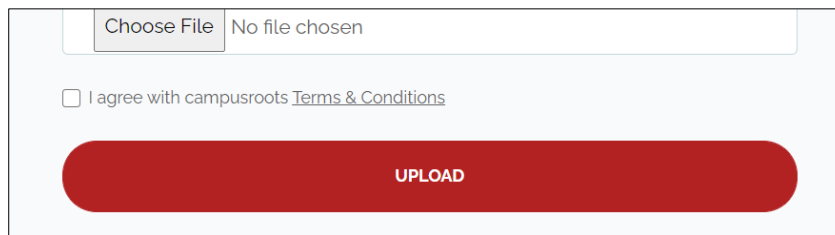
Figure 59: the date appears in page after set

No.Page: 62
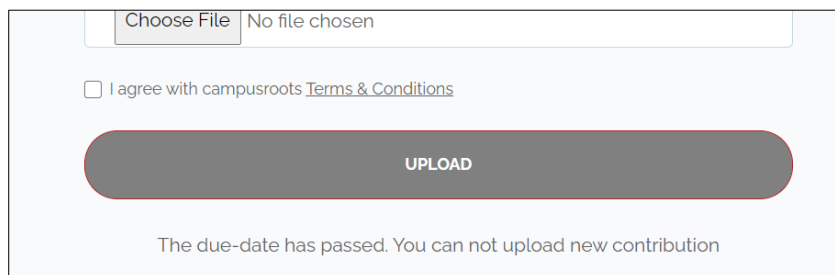
Test 6: Student should not upload after due date

For testing purposes, I changed due date and checked if upload button is disable or not.

Figure 60: upload button when due date has not passed
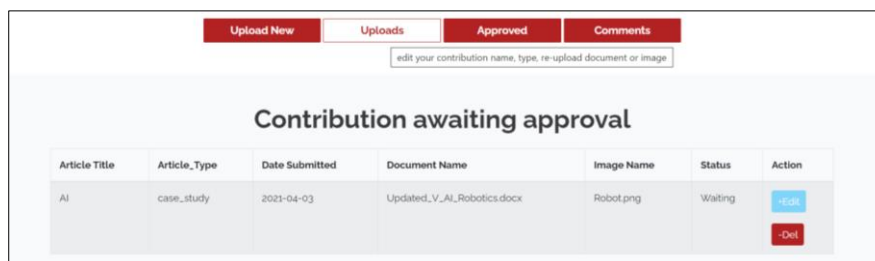
Figure 61: upload button disabled when due date has passed

Test 7: Student can view and edit contribution
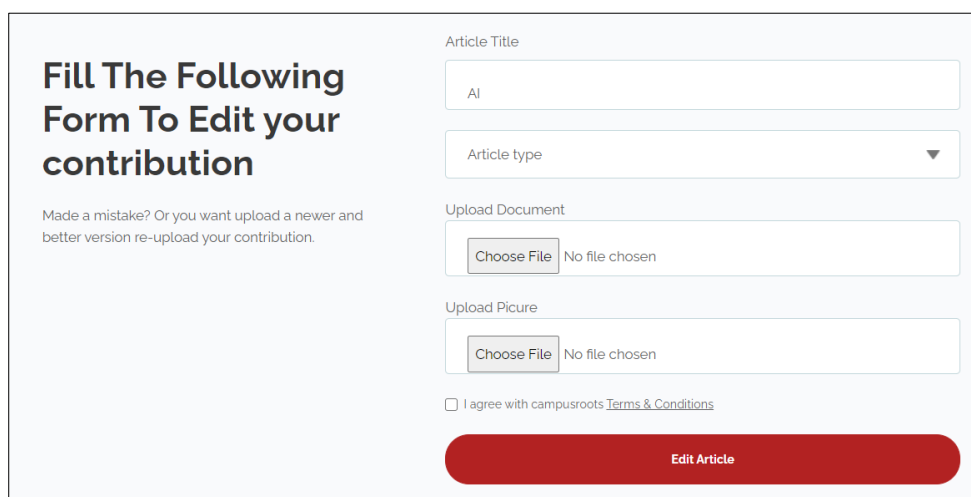
If contributions are uploaded the students can edit their contribution if they want to update information in it.

Figure 62: Edit page for selecting article to edit

Figure 63: editing article form
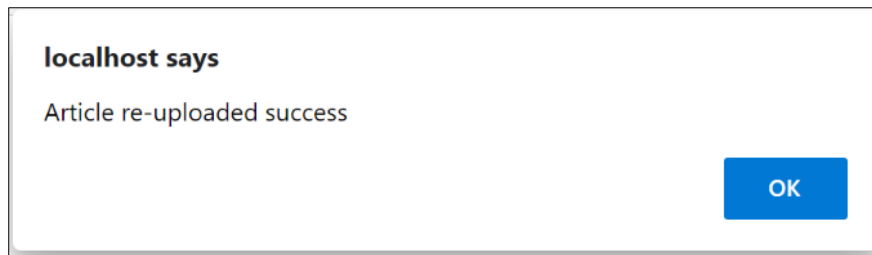
Figure 64: if editing article is successful

Test 8: Student should not edit after closure date

Figure 65: edit article button when closure date has not passed

Figure 66: edit article button disabled when closure date has passed

Test 9: Marketing Coordinator can view contributions in their faculty

For example, I tested on faculty of law. If they press button it will download and they can view the article.

Figure 67: coordinator can view articles

Test 10: Coordinator can comment on article

They can also comment using the form if not entered error message below and if characters count reaches 0 the user cannot enter more else if everything corrects than comment is added.

Figure 68: comment form for coordinator to communicate with student and if not filled error below input box

Figure 69: form allowing for comment and word count left displayed

Figure 70: message when comment is successful

Test 11: Select student contribution for publishing

Coordinator can choose contribution for publishing by clicking the waiting button which changes to approved.

Figure 71: shows article status that is waiting

Email	Article Title	Article Type	Submission Date	Document	Image	Status	Comment
jeon@gmail.com	Laws in Zambia part 2	case_study	2021-04-01	✓ D	✓ D	Approved	+Comment
jeon@gmail.com	Laws in Zambia	case_study	2021-04-01	✓ D	✓ D	Waiting	+Comment

Figure 72: after waiting button pressed for approval status changes to approved

Test 12: Student view comments and approved documents

They can view comments and those with if none it displays no comment.

Check your comments below

Article Title	Article Type	Comment
Laws in Zambia part 2	case_study	Good but improve on section 1
Laws in Zambia	case_study	No comment

Figure 73: table displaying comment

The contributions that are approved for publishing and those that still waiting are displayed.

Contribution awaiting approval

Article Title	Article_Type	Date Submitted	Document Name	Image Name	Status	Action
Laws in Zambia	case_study	2021-04-01	Jeon_Laws.docx	law_j.jpg	Waiting	+Edit -Del

Figure 74: table displaying contributions not approved

Approved Contribution

Article Title	Article_Type	Date Submitted	Document Name	Image Name	Status
Laws in Zambia part 2	case_study	2021-04-01	Jeon_Laws2.docx	law_j2.jpg	Approved

Figure 75: table displaying contributions that are approved

No.Page: 66

Test 13: Marketing Manager should view contributions

They can view all contributions in sections like student awaiting approval and those selected for publish.

Figure 76: Manager can view table displaying contributions not approved

Figure 77: Manager can view table displaying contributions approved to download

Test 14: Marketing Manager can download contributions

The Manager can download all the approved contributions at once using the download button, the contributions are downloaded in zip format.

Figure 78: download button for manager
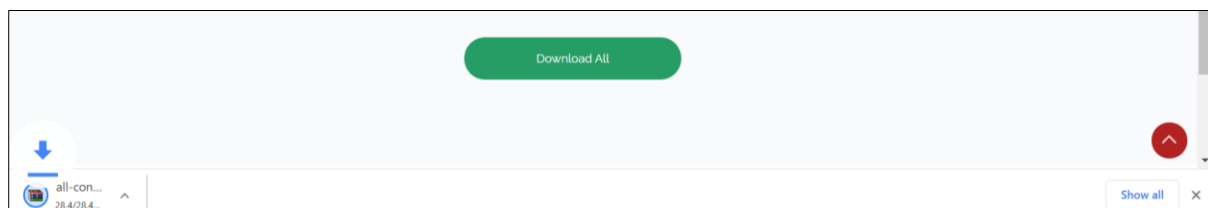
Figure 79: when download button pressed

No.Page: 67
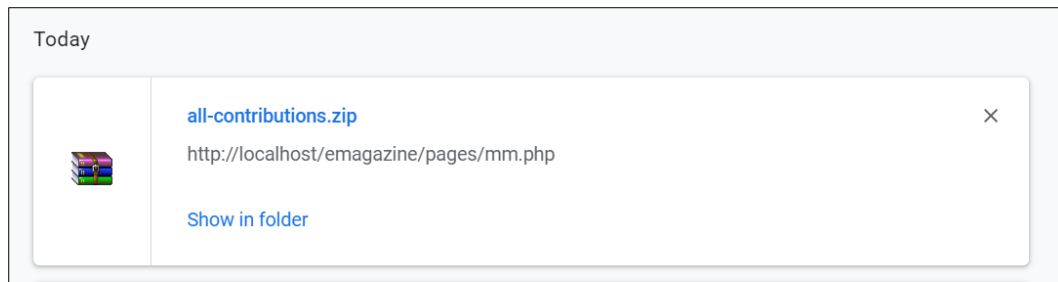
Figure 80: evidence of zip file downloaded

Test 15: Marketing Manager can view statistics

Under statistics they can view contributions from each faculty for any academic year and they can view it in percentage too. They can view exception report for those contributions without comment. Manager can also generate reports for different academic year.
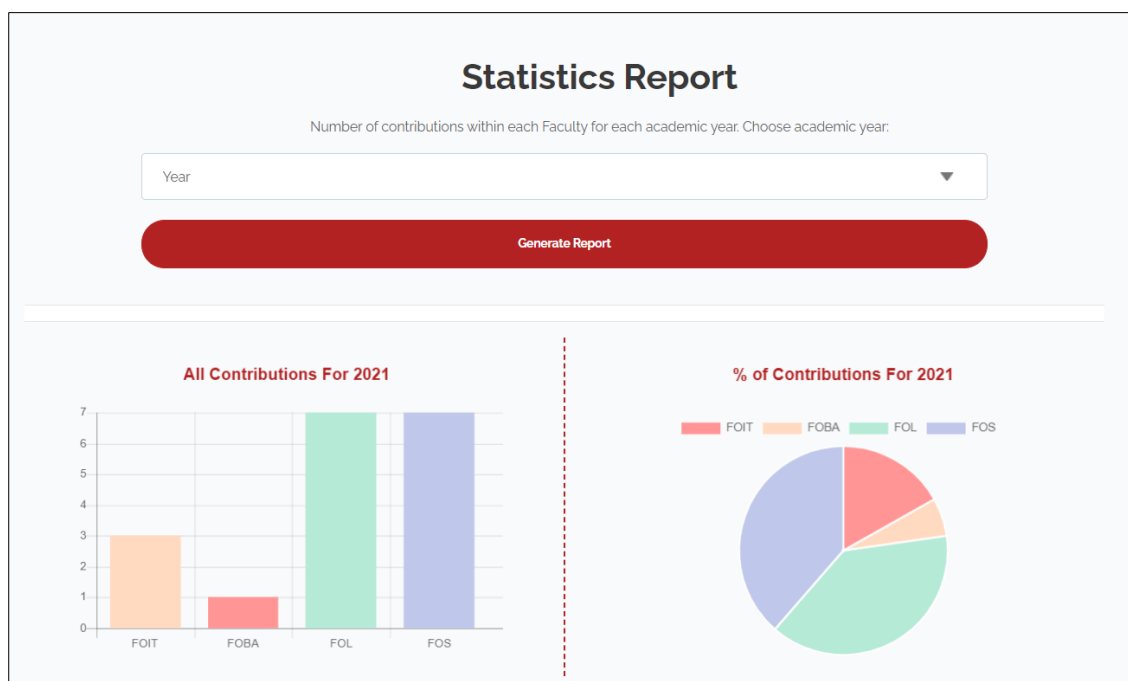
Figure 81: Statistical report for all contribution reports per year
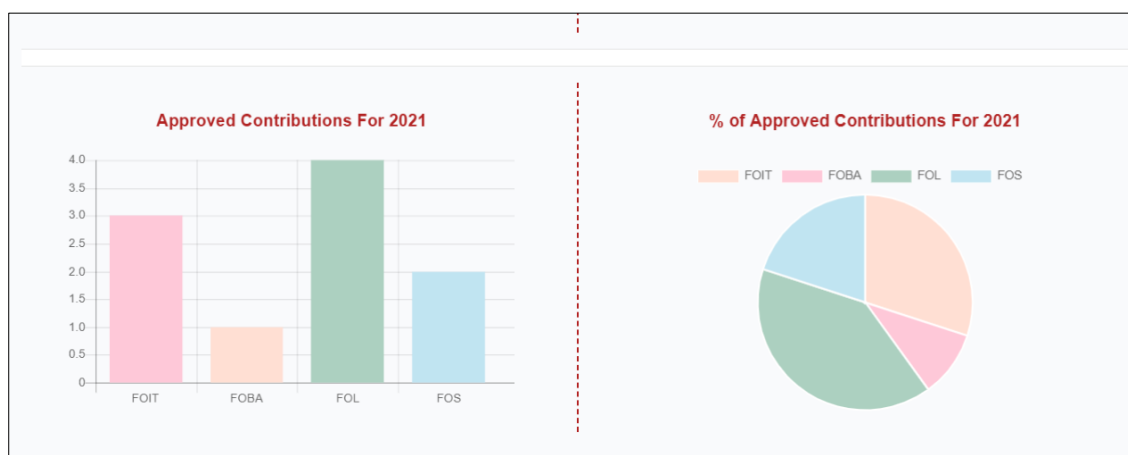
Figure 82: Statistical report for approved contribution reports per year

No.Page: 68
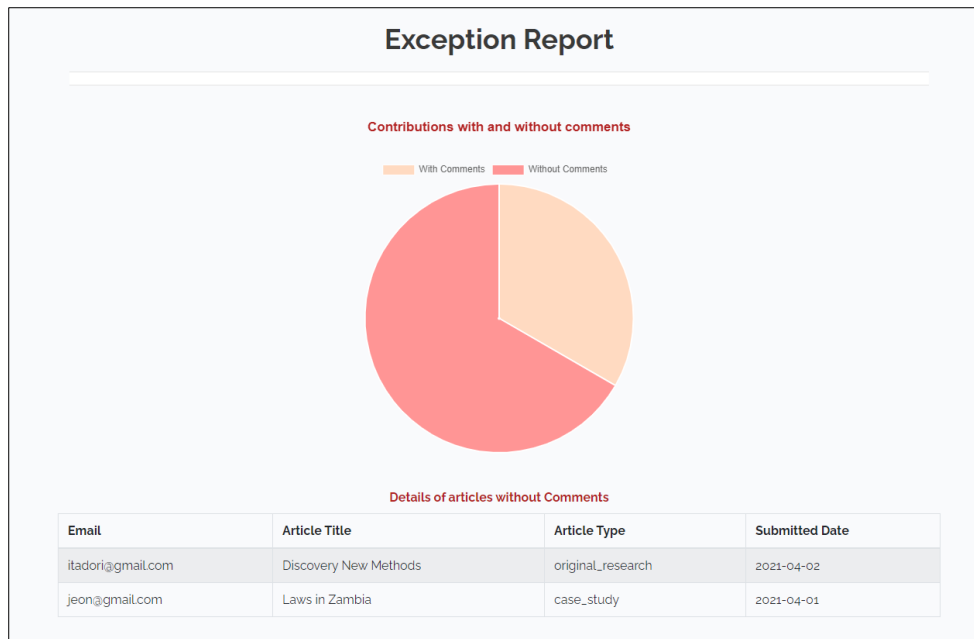
Figure 83: Exceptional report for no comment articles

Test 16: Guests can view reports of their faculty

Guests can choose faculty and after choice is made, they can view statistical and exception reports displaying contributions per academic year in value and in percentage, they can also view the contributions without comments.
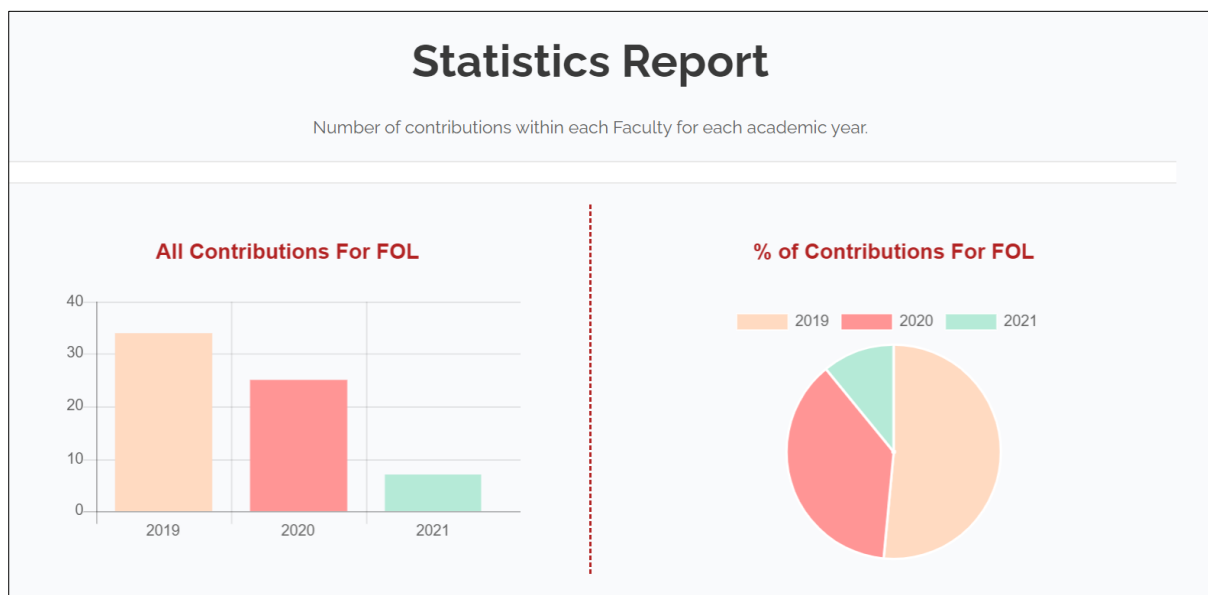
Figure 84: Statistical report for guest according to faculty
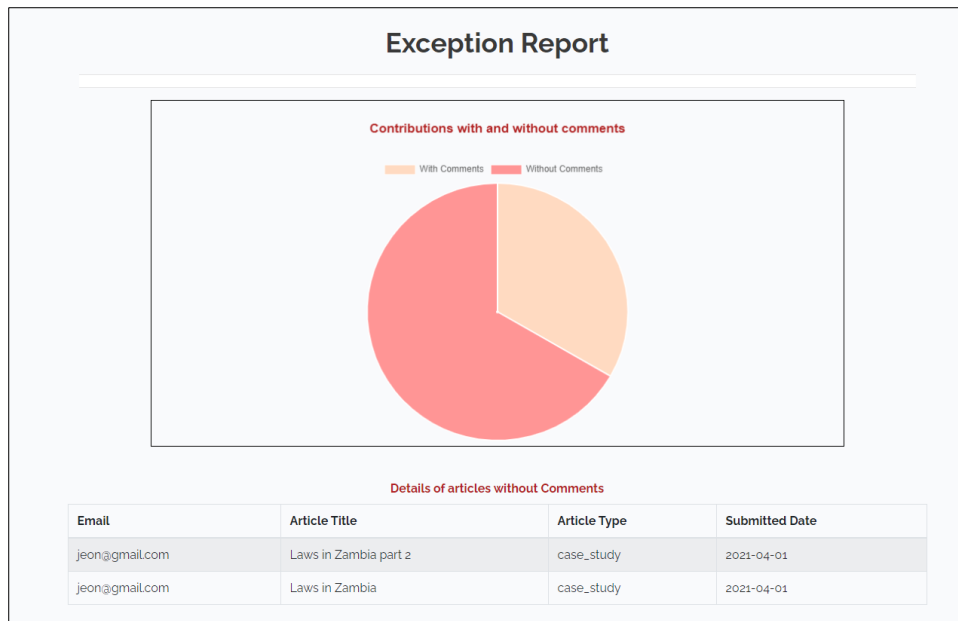
Figure 85: Exceptional report for guest according to faculty

9.7 Functionality Code snippets

Below are some codes for important functionality.

- Admin creating users:

```php
// REGISTER USER
function create_user_stu(){
    global $connection, $errors;

    // receive all input values from the form
    $first_name = mysqli_real_escape_string($connection, $_POST['first_name']);
    $last_name = mysqli_real_escape_string($connection, $_POST['last_name']);
    $email = mysqli_real_escape_string($connection, $_POST['email']);
    $phone_no = mysqli_real_escape_string($connection, $_POST['phone_no']);
    $password = mysqli_real_escape_string($connection, $_POST['password']);
    $f_code = $_POST['f_code'];

    //Check if user exists
    $sql = "SELECT * FROM accounts WHERE email = '$email' ";
    $ret = mysqli_query($connection, $sql);
    $retresults = mysqli_num_rows($ret);

    if ($retresults == 0) {
        if (count($errors) == 0) {
            //Encrypt Password
            $hash = password_hash($password, PASSWORD_DEFAULT);

            $sql_ac = mysqli_query($connection, "INSERT INTO `accounts` (`email`, `password`, `u_id`)
            VALUES ('$email', '$hash', 'STU')");

            // You, 2 days ago • delete staff and students

            if ($sql_ac > 0){
                $sql_stu = "INSERT INTO `students` (`email`, `first_name`, `last_name`, `phone_no`, `f_code`)
                VALUES ('$email', '$first_name', '$last_name', '$phone_no', '$f_code')";
                mysqli_query($connection, $sql_stu);
            }
            //Add sucess to alert
            echo '<script>alert("Account created")</script>';
        }
        // die(print_r($re2));
    }else {
        array_push($errors, "Email is in use");
    }
}
```

Figure 86: code for admin creating users

No.Page: 70

- Student editing contribution

```php
function ed_art(){
    global $connection;
    //Get inputs from user
    $article_title = $_POST['article_title'];
    $article_type = $_POST['article_type'];
    $art_id = $_SESSION['art_id'];
    $old_doc_name = $_SESSION['old_doc_name'];
    $old_img_name = $_SESSION['old_img_name'];
    You, 2 weeks ago • Edit uploaded article
    $doc_sou = '../upload/' . $old_doc_name; // Get old document file from folder
    $img_sou = '../upload/' . $old_img_name; // Get old image file from folder
    //Insert input into db
    if (isset($_SESSION['art_id'])) {//Student insertion
        // Upload files
        if (isset($_POST['ed_art'])) { // if save button on the form is clicked
            // name of the uploaded files
            $doc_name = $_FILES['upload_doc']['name'];
            $img_name = $_FILES['upload_image']['name'];

            // destination of the file on the server
            $destination = '../upload/' . $doc_name;
            $img_destination = '../upload/' . $img_name;

            // get the file extension
            $extension = pathinfo($doc_name, PATHINFO_EXTENSION);
            $img_extension = pathinfo($img_name, PATHINFO_EXTENSION);

            //Check for the correct file extension
            if (!in_array($extension, ['docx', 'doc'])) {
                echo array_push($errors, "Sorry the document you've uploaded must be .docx/.doc");
            }
            if (!in_array($img_extension, ['png', 'jpeg', 'jpg'])) {
                echo array_push($errors, "Sorry the image you've uploaded must be .png/.jpeg/.jpg");
            }

            // the physical file on a temporary upload directory on the server
            $doc_file = $_FILES['upload_doc']['tmp_name'];
            $img_file = $_FILES['upload_image']['tmp_name'];

            // move the uploaded (temporary) file to the specified destination
            if (move_uploaded_file($doc_file, $destination) and move_uploaded_file($img_file, $img_destination)) {
                $sql = "UPDATE articles SET title='$article_title', art_type='$article_type', doc_name='$doc_name', img_name='$img_name' WHERE id='$art_id'";

                if (mysqli_query($connection, $sql)) {
                    echo '<script>alert("Article re-uploaded success")</script>';
                    echo "<script>window.location='../pages/article.php'</script>";
                }
            } else {
                echo '<script>alert("Article uploaded failed")</script>';
```

Figure 87: code for student editing contribution

- Marketing coordinator adding comments

```php
// setting session
if (isset($_POST['comment'])) {
    $id= $_POST['comid'];
    $_SESSION["cid"] = $id;
}

// to add comments
if (isset($_POST['s_comment'])) {
    $nid =  $_SESSION["cid"];

    $mes= $_POST['mes'];


    $query = "UPDATE articles SET comment='$mes' WHERE id='$nid'";

    $result = $connection->query($query);

    if ($result) {
        echo '<script>alert("Comment has been added")</script>';
        echo "<script>window.location='mc.php'</script>";
    }
    else {
        echo '<script>alert("Something went wrong")</script>';
    }
}
```

Figure 88: code for adding comments

No.Page: 71

- Coordinator view contributions

```php
//Downloads Document
function downdoc(){
    global $connection;

    $id = $_GET['doc_id'];
    // fetch file to download from database
    $sql = "SELECT * FROM articles WHERE id=$id";
    $result = mysqli_query($connection, $sql);
    $file = mysqli_fetch_assoc($result);
    $filepath = '../upload/' . $file['doc_name'];

    if (file_exists($filepath)) {
        header('Content-Description: File Transfer');
        header('Content-Type: application/octet-stream');
        header('Content-Disposition: attachment; filename=' . basename($filepath));
        header('Expires: 0');
        header('Cache-Control: must-revalidate');
        header('Pragma: public');
        header('Content-Length: ' . filesize('../upload/' . $file['doc_name']));
        readfile('../upload/' . $file['doc_name']);
        exit;}
}
//Downloads Image
function downimg(){
    global $connection;

    $id = $_GET['img_id'];

            // fetch file to download from database
            $sql = "SELECT * FROM articles WHERE id=$id";
            $result = mysqli_query($connection, $sql);

            $file = mysqli_fetch_assoc($result);
            $filepath = '../upload/' . $file['img_name'];

            if (file_exists($filepath)) {
                header('Content-Description: File Transfer');
                header('Content-Type: application/octet-stream');
                header('Content-Disposition: attachment; filename=' . basename($filepath));
                header('Expires: 0');
                header('Cache-Control: must-revalidate');
                header('Pragma: public');
                header('Content-Length: ' . filesize('../upload/' . $file['img_name']));
                readfile('../upload/' . $file['img_name']);
                exit;
            }
}
```

Figure 89: code for coordinator to view contributions submitted

- manager can zip selected articles

```php
function all_zip() {//Add all documents in zip file
    if(isset($_REQUEST['all_zip'])){
        extract($_REQUEST);

        global $connection;
        $findDoc = "SELECT * FROM articles WHERE status = 'Approved'";
        $checkDoc = mysqli_query($connection, $findDoc);

        $filename   =   'all-contributions.zip';

        $zip = new ZipArchive;
        if ($zip->open($filename,  ZipArchive::CREATE)){
            while($row  =   $checkDoc->fetch_assoc()){
                $zip->addFile(getcwd().'/'.'../upload/'.$row['doc_name'], $row['doc_name']);
                $zip->addFile(getcwd().'/'.'../upload/'.$row['img_name'], $row['img_name']);
            }


            $zip->close();

            header("Content-type: application/zip");
            header("Content-Disposition: attachment; filename=$filename");
            header("Content-length: " . filesize($filename));
            header("Pragma: no-cache");
            header("Expires: 0");
            readfile("$filename");
            unlink($filename);
        }else{
            echo 'Failed!';
        }
            You, 3 weeks ago • zip downloads
    }
}
```

Figure 90: code for zipping files

- Admin setting due date

```php
// else if there is no date in table add
$year = date("Y");

// post from form
$due_date = $_POST['dued'];

// set as date and add 14 days to it
$date=date_create($due_date);
date_add($date,date_interval_create_from_date_string("14 days"));
$fd = date_format($date,"Y-m-d");

// insert into date when admin sets date
$sql = "INSERT INTO dates (Year,due_date, final_closure_date) VALUES ('$year','$due_date','$fd') ";
$result = $connection->query($sql);

// show if date is added
if ($result) {
    echo "<script>alert('Date has been added')</script>";
}
else {
    echo "<script>alert('Some error occured')</script>";
}
}
```

Figure 91: code for setting dates by admin

No.Page: 73

- Generate reports in guest page for their faculty

Statistical report:

```php
// if the guest logs in
if(isset($_POST['guest_login'])){
    // collect their faculty code         You, seconds ago • Uncommitted changes
    $f_type = $_POST['f_type'];
    //for first chart store values on
    $sql ="SELECT * FROM statistics WHERE f_code='$f_type'";
    $result = mysqli_query($connection,$sql);
    $chart_data="";
    // store year and artcile values in array
    while ($row = mysqli_fetch_array($result)) {
            $yearoa[]  = $row['Year']  ;
            $no[] = $row['articles'];
    }


    //percentage pie chart
    $query = "SELECT * FROM statistics WHERE f_code='$f_type'";
    $query_run = mysqli_query($connection,$query);
    $qty= 0;
    // add all values in article table
    while ($num = mysqli_fetch_assoc ($query_run)) {
            $qty += $num['articles'];
    }

    $qu = "SELECT * FROM statistics WHERE f_code='$f_type'";
    $qu_run = mysqli_query($connection,$query);
    $chart_data="";
    // divide by total $qty and multiply by 100 for percentage value
    while ($row = mysqli_fetch_array($qu_run)) {
            $artic = $row['articles'];
            $per = ($artic / $qty) * 100;
            echo "<br>";

            // store percentage value and year in array
            $percentage[]  = round($per);
            $l[] = $row['Year'];
    };
```

Figure 92: code for statistical report for guests

No.Page: 74

Exceptional report:

```php
// comments grapgs
// with comments
$sqll = "SELECT * FROM articles WHERE f_code='$f_type' ";
$resultt = $connection->query($sqll);

  if ($resultt->num_rows > 0) {

    $com = "SELECT * FROM articles WHERE f_code='$f_type' AND comment IS NOT NULL";
    $com1 = mysqli_query($connection,$com);

        if($com1->num_rows > 0){
            while($row = $com1->fetch_assoc()) {
            // store value of article number in array
            $aid[] = $row['id'];
            }
                // count the value stored
                $i[] = count($aid);
        } else{
                // if no comment is null store 0 in array
                $i[] = 0;
        }

    // without comments
    $comm = "SELECT * FROM articles WHERE f_code='$f_type' AND comment IS NULL";
    $com2 = mysqli_query($connection,$comm);
        if($com2->num_rows > 0){
            while($row = $com2->fetch_assoc()) {
            // store value of article number in array
            $id[] = $row['id'];
            }
                // count the value stored
                $ci[] = count($id);
        } else{
                // if no comment is null store 0 in array
                $ci[] = 0;
        }
  }
```

Figure 93: code for exceptional report for guests