CHAPTER 6

SYSTEM IMPLEMENTATION AND INTEGRATION

**6.1 INTRODUCTION:**

This chapter will include the implementation and integration process of the project, the implementation section will talk about the hardware and software that has been used to evolve our application and website. Also, it's include some of problem which faced us during the work, some parts of the project code for the main and major functions will be displayed. The second section will contain the different system component and how it's integrated with each other.

**6.2 HARDWARE AND SOFTWARE**

**6.2.1 HARDWARE**

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Device** | **Quantity** | **Description** |
| **1** | Windows laptop | 5 | Used for developing the application and website |
| **2** | Android mobile | 1 | To test the application |
| **3** | Printer | 1 | To test the system and print the project document |

**Table 6-1: Hardware specification**

**6.2.2 SOFTWARE:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Software** | **Developer** | **Version** | **Description** |
| Dropbox | Dropbox, Inc |  | Used to store our project and project document |
| MS Office Word | Microsoft | 2016 | To write our documentation |
| phpMyAdmin | ThephpMyAdmin Project | 5.6.30 | A tool to manage MySQL database by creating, modifying or deleting tables or row |
| Xampp | Apache Friends | - | Used to create a local web server for testing purposes. |
| Android studio and android  SDK | Andrew | - | To implement the application. |
| GitHub | GitHub, Inc. | - | A Git repository hosting server used for source code management and integration. |

**Table 6-2: Software specification**

6.3 IMPLEMENTATION DIFFICULTIES:

Many of fortuitous challenges faced the programmers and developers when they do their work but they avoid it and dealing with it expertly. We too as programmers had faced many difficulties, but we handled the issues well enough to exceed them. We will mention some of the challenges and problems that took a lot of time from us to discover and solve.

1. Some of the functions took a long time to implement, and the reason was that the ideas were new and we were dealing for the first time with these requirements, but solving them did not require more than effort and time and detailed research**.**
2. We had trouble translating parts of the code to appear in Arabic because we design our website in Arabic only and most of the libraries are supported in English. Examples of that, the text editor and the nationality dropdown list.
3. Dealing with Android was very difficult because of its delay to respond to as we know it supported by many features for several of platforms and our computer does not help much efficiency.
4. When we were implementing and testing the send email function, we encountered a problem with the server we used, it gives no response to our sending because of the huge number of attempts.

|  |  |  |
| --- | --- | --- |
| **Problem** | **Description** | **Solution** |
| SMTP ERROR: QUIT command failed:  SMTP connect() failed. | The email server blocking or not receiving our requirement to send email | Implement an array of solution which keep trying until the send done successfully. |

**Table 6-3: problem and it solution**

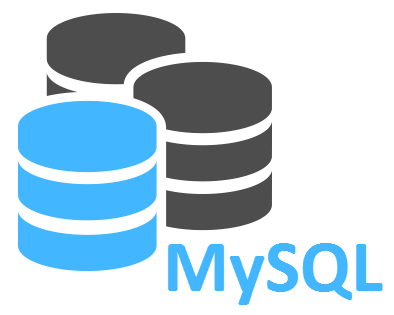
# 6.4 IMPLEMENTATION DETAILS:

As we see that our system includes both application and website, so here is specification for each platform .The application is advanced to be under android platform , we used PHP as API to allow the application being communicate with project server and MySQL as a database tool furthermore, JSON for data transformation and interchanging between our application and server.

API

MySQL DATABASE

Android application







JSON parser

**Figure 6-1 : Mobile application development**

In addition, for the website platform we used PHP as API to allow website pages being communicate with project server and MySQL as a database tool furthermore, JSON, AJAX for data transformation and interchanging between our pages and server to make the usability more interactive by sharing small amounts of data with web provider.

**6.4.1 Code:**

In this section of the implementation chapter, we are going to mention parts of source code of main functions for the application and website.

**6.4.1.1 Organizer privilege:**

The organizer has a privilege to do many tasks after using the website and the application. In this figure we're going to display most of the functions that the organizer can do it after using the website .

Log In

Manage prize

Manage event

Manage certificate

Manage badge

Manage email

Manage subevent

Statistics

Add badge

Send email

Add event

Chose prize winner

Manage rate

Edit event

Manage form

edit badge

Create email template

Manage TrackAttendee

Show event details

Manage onsite

Send badge

**Figure 6-2 : organizer privilege interfaces**

**Table 6-4:Add event function code**

|  |
| --- |
| **Function name:** Add event |
| **Description :** this function allow the organizer to add event while he using the website |
| |  | | --- | | if (isset($\_SESSION['emailconfirm']) and $\_SESSION['emailconfirm'] == 1) { | |  | if (isset($\_POST['add'])) { | |  | $eventName = $\_POST['eventName']; | |  | $EventDescription = $\_POST['description']; | |  | $sdate = $\_POST['sdaytime'] . ' 00:00:00.000'; | |  |  | |  | $edate = $\_POST['edaytime'] . ' 00:00:00.000'; | |  | $dateend = date('Y-m-d', strtotime($edate)); | |  | $location = $\_POST['location']; | |  | $organizationName = $\_POST['organizer']; | |  | $maxAttendee = $\_POST['maxAttendee']; | |  | if ($datestart > $dateend) { | |  | echo " <div class='alert alert-danger alert-dismissible'> | |  | <button type='button' class='close' data-dismiss='alert'>&times;</button> | |  | <strong> فشل </strong> يرجى التحقق من تاريخ بداية ونهاية الحدث | |  | </div> "; | |  | } else { | |  | $IDT = $\_SESSION['organizerID']; | |  | $sql = mysqli\_query($con, "INSERT INTO event ( event\_ID, name\_Event, descrption\_Event ,sartDate\_Event,endDate\_Event,location\_Event,organization\_name\_Event,maxNumOfAttendee,organizer\_ID) VALUES ('','$eventName','$EventDescription','$datestart','$dateend','$location','$organizationName','$maxAttendee','$IDT')") or die(mysqli\_error($con)); | |  | if ($sql) { | |  | header("location: /tactic/manageEvent.php"); | |  | exit; | |  | } else { | |  | echo " <div class='alert alert-danger alert-dismissible'> | |  | <button type='button' class='close' data-dismiss='alert'>&times;</button> | |  | <strong> فشل</strong> لم تتم عملية الاضافة بنجاح يرجى التحقق | |  | </div> "; | |  | } | |  | }} | |  | } else { | |  | echo " <div class='alert alert-danger alert-dismissible'> | |  | <button type='button' class='close' data-dismiss='alert'>&times;</button> | |  | <strong> يرجى</strong> تثبيت الايميل لكي تتمكن من أضافة حدث | |  | </div> "; | |  | } | |  | ?> | |

**Table 6-5: send email function code**

|  |
| --- |
| **Function name:** Send email |
| **Description :** This function allow the organizer to send email for all type of attendees |
| |  |  |  | | --- | --- | --- | | $mail = new PHPMailer\PHPMailer\PHPMailer(); | | | |  | | $mail = new PHPMailer\PHPMailer\PHPMailer(); | | |  | $mail->CharSet = "utf-8"; | | | |  | $mail->SMTPKeepAlive = true; | | | |  | $mail->SMTPOptions = array( | | | |  | 'ssl' => array( | | | |  | 'verify\_peer' => false, | | | |  | 'verify\_peer\_name' => false, | | | |  | 'allow\_self\_signed' => true | | | |  | ) | | | |  | ); | | | |  | $mail->IsSMTP(); // enable SMTP | | | |  | //$mail->SMTPDebug = SMTP::debugDumpParams; | | | |  | $mail->SMTPDebug = 0;// debugging: 1 = errors and messages, 2 = messages only | | | |  |  | | | |  | $mail->SMTPAuth = true; // authentication enabled | | | |  | $mail->SMTPSecure = 'tls'; // secure transfer enabled REQUIRED for Gmail | | | |  | $mail->Host = "smtp.gmail.com"; | | | |  | $mail->Port = 587; // or 587 | | | |  | $mail->IsHTML(true); | | | |  | $mail->Username = 'tacticevent131@gmail.com'; | | | |  | $mail->Password = 'marwa123456789'; | | | |  | $mail->setfrom('tacticevent131@gmail.com', 'tactic'); | | | |  | ////////////////////////////////////////////////////////////// | | | |  | if (isset($\_GET['event\_ID'])) { | | | |  | $event\_ID = $\_GET['event\_ID']; | | | |  | $query = mysqli\_query($con, "SELECT \* FROM event WHERE event\_ID = '$event\_ID' ") or die(mysqli\_error($con)); | | | |  | $rows = mysqli\_fetch\_array($query); | | | |  | $eventName = $rows['name\_Event']; | | | |  | $eventDate = $rows['sartDate\_Event']; | | | |  | if (isset($\_GET['all'])) | | | |  | $sql = mysqli\_query($con, "SELECT \* FROM attendee WHERE eventId = '$event\_ID' ") or die(mysqli\_error($con)); | | | |  | elseif (isset($\_GET['VIP'])) | | | |  | $sql = mysqli\_query($con, "SELECT \* FROM attendee WHERE eventId = '$event\_ID'AND VIPcode !=0") or die(mysqli\_error($con)); | | | |  | elseif (isset($\_GET['normal'])) { | | | |  | $sql = mysqli\_query($con, "SELECT \* FROM attendee WHERE eventId = '$event\_ID' AND VIPcode = 0 ") or die(mysqli\_error($con)); | | | |  | } //isset($\_GET['normal']) | | | |  | } //isset($\_GET['event\_ID']) | | | |  | if ($sql) { | | | |  | $x = 0; | | | |  | while ($row = mysqli\_fetch\_array($sql)) { | | | |  | $email\_Att[$x] = $row['email']; | | | |  | $mail->AddAddress($row['email']); | | | |  | $name[$x] = $row['name']; | | | |  | $mensaje = file\_get\_contents('emailTemplate.html'); | | | |  | $mensaje = str\_replace("{{USERNAME}}", $name[$x], $mensaje); | | | |  | $mensaje = str\_replace("{{eventName}}", $eventName, $mensaje); | | | |  | $mensaje = str\_replace("{{eventDate}}", $eventDate, $mensaje); | | | |  | $x++; | | | |  | if (isset($\_POST['submit'])) { | | | |  | $subject = $\_POST['subject']; | | | |  | $body = $\_POST['message']; | | | |  | if (isset($\_POST['logo'])) | | | |  | $logo = $\_POST['logo']; | | | |  | $mail->Subject = $subject; | | | |  | $mensaje = str\_replace("{{subject}}", $body, $mensaje); | | | |  | $mensaje = str\_replace("{{eventLogo}}", $logo, $mensaje); | | | |  |  | | | |  | $mail->Body = $mensaje ; | | | |  | if (!$mail->send()) { | | | |  | $masg = " <div class='alert alert-danger alert-dismissible'> | | | |  | <button type='button' class='close' data-dismiss='alert'>&times;</button> | | | |  | <strong> حدث خطأ </strong> غير متوقع ! فضلا حاول مرة اخرى. | | | |  | </div> "; | | | |

**Table 6-6:chose prize winner function code**

|  |
| --- |
| **Function name:** chose prize winner. |
| **Function description:** this function chose a prize winner from the attendees with his/her name, phoneNumber , email , nationality . |
| |  |  | | --- | --- | |  | if (isset($\_POST['save'])) {  $prizeId = $\_GET['prizeId']; | |  | // in case the user try to save more the one time we will delete old values and add the new winners | |  | $query = "DELETE from attendeewinners where prizeId=$prizeId"; | |  | // to get the record of prize that already added from add page | |  | $prizeQuery = mysqli\_query($con, "SELECT \* FROM prize WHERE prize\_ID ='$prizeId' ") or die(mysqli\_error($con)); | |  | $row = mysqli\_fetch\_row($prizeQuery); | |  | $numOfPrize = $row[2]; | |  | for ($countPrize = 0; $countPrize < $numOfPrize; $countPrize++) { | |  |  | |  | $attendeeId = $\_POST['attendeeId' . $countPrize]; | |  |  | |  | $query = "INSERT INTO attendeewinners (attendeeId, prizeId) | |  | VALUES ($attendeeId,$prizeId)"; | |  |  | |  | $sql = mysqli\_query($con, $query) or die(mysqli\_error($con)); | |  | if (!$sql) { | |  | $isSuccessOperation = false; | |  | break; | |  | } | |  | } | |
| |  | | --- | |  | |  |  | |  | if ($isSuccessOperation) { | |  | header("location: /tactic/managePrize.php"); | |  | exit; | |  | } else { | |  | // in case we have any problem with saving winners we will delete all winner related with this | |  | //prize to start another clean operation | |  |  | |  | $query = "DELETE from attendeewinners where prizeId=$prizeId"; | |  | $sql = mysqli\_query($con, $query) or die(mysqli\_error($con)); | |  | echo "<div class='alert alert-danger alert-dismissible'> | |  | <button type='button' class='close' data-dismiss='alert'>&times;</button> | |  | لم تتم عملية حفظ الفائزين بنجاح يرجى الإعادة | |  | </div> "; | |  | } | |  | } | |  | if (isset($\_GET['prizeId'])) { | |  | $prizeId = $\_GET['prizeId']; | |  |  | |  | // to get the record of prize that already added from add page | |  | $prizeQuery = mysqli\_query($con, "SELECT \* FROM prize WHERE prize\_ID ='$prizeId' ") or die(mysqli\_error($con)); | |  | $row = mysqli\_fetch\_row($prizeQuery); | |  | $namePrize = $row[1]; | |  | $numOfPrize = $row[2]; | |  | $eventId = $row[3]; | |  | $subEventId = $row[4]; | |  |  | |  |  | |  | // to load the winners that already selected in case we have them | |  | if (!isset($\_POST['chooseWinner'])) { | |  | // to get the record of prize that already added from add page | |  | $query = "Select att.id , att.name,phone,email,na.Name as | |  | nationalityName ,DOB from attendeewinners at winner join attendee att on att.id=atw.attendeeId | |  | inner join nationality na on na.Id=att.nationalityId where atw.prizeId =$prizeId"; | |  | $winnerQuery = mysqli\_query($con, $query)or die(mysqli\_error) | |  | |

**Table 6-7: edit badge function code**

|  |
| --- |
| **Function name:** edit attendee badge |
| **Function description:**  this function give the organizer the ability to edit the organizer badge when ever he want with specific color ,size ,code place. |
| |  | | --- | |  | |  | |  | function calculateX($value)  { | |  | $yposition = strpos($value, "Y"); | |  | $xpostion = strpos($value, "X"); | |  | $left = substr($value, $xpostion + 1, $yposition - 1); | |  | return $left; | |  | } | |  |  | |  | function calculateY($value) | |  | { | |  | $yposition = strpos($value, "Y"); | |  | $top = substr($value, $yposition + 1); | |  | return $top; | |  | } | |  |  | |  | // this section for get the event name from DB for curent organizer | |  | $query = mysqli\_query($con, "SELECT \* FROM event where organizer\_ID='$organizerID ' ") or die(mysqli\_error()); | |  | //get the value of drop-down list for barcode size, font color and font size from DB | |  | $barcodesize = mysqli\_query($con, "SELECT \* FROM barcodesize ") or die(mysqli\_error()); | |  | $color = mysqli\_query($con, "SELECT \* FROM color ") or die(mysqli\_error()); | |  | $fontsize=mysqli\_query($con, "SELECT \* FROM fontsize ") or die(mysqli\_error()); | |  |  | |  | // to get the badge type from lookup tabel | |  | $badgeTypeQuery = mysqli\_query($con, "SELECT \* FROM badgetype") or die(mysqli\_error()); | |  |  | |  | $badgeIfo = mysqli\_query($con, "SELECT \* FROM badge b INNER JOIN imageinfo img ON img.badgeId = b.badge\_ID | |  | where b.badge\_ID='$badgeIdForEdit ' ") | |  | or die(mysqli\_error()); | |  |  | |  | $row = mysqli\_fetch\_array($badgeIfo); | |  | $eventIdEd= $row[2]; | |  | $BadgeTypeEd= $row[1]; | |  | $colorEd=$row[8]; | |  | $barSizeEd=$row[9]; | |  | $fontSizeEd=$row[10]; | |  | $imageEd=$row[6]; | |  | $imageinfoId=$row[7]; | |  | $namePosition=$row[12]; | |  | $careerPosition=$row[13]; | |  | $barcodePosition=$row[14]; | |  | $imagePosition=$row[15]; | |  |  | |  | $name = $row[3]; | |  | $size = $row[4]; | |  | $type = $row[5]; | |  | $location = $row[6]; | |  | $badgeName=substr($type,6); | |  | $chechUpload=1; | |  | if (isset($\_POST['add'])) { | |  | $eventId = $\_POST['eventId']; | |  | $badgeTypeId = $\_POST['badgeTypeId']; | |  |  | |  | if ($BadgeTypeEd!= $badgeTypeId){ | |  | $checkQuery = mysqli\_query($con, "SELECT \* FROM badge WHERE event\_ID='$eventId' and BadgeTypeId='$badgeTypeId'") or die(mysqli\_error()); | |  | if (mysqli\_num\_rows($checkQuery) > 0) { | |  | $chech=0; | |  | }else{ | |  | $chech=1;} | |  | }else{ | |  | $chech=1;} | |  |  | |  | if ($chech== 0) { | |  | $message= " <div class='alert alert-danger alert-dismissible'> | |  | <button type='button' class='close' data-dismiss='alert'>&times;</button> | |  | <strong> فشل</strong> الحدث مرتبط ببطاقة مسبقا لايمكن اتمام العملية | |  | </div> "; | |  |  | |  | } else { | |  |  | |  | if ( !empty($\_FILES["fileToUpload"]["name"])) { | |  |  | |  | $name = $\_FILES['fileToUpload']['name']; | |  | $size = $\_FILES['fileToUpload']['size']; | |  | $type = $\_FILES['fileToUpload']['type']; | |  | $tmp\_name = $\_FILES['fileToUpload']['tmp\_name']; | |  | $badgeName=substr($type,6); | |  | // add eventId & badgeTypeId to the name of file to make sure the name is unique | |  | $location = "UploadFile/".$eventId."/".$badgeTypeId.$eventId.".".$badgeName; | |  |  | |  | // save image in Specific position | |  | if (move\_uploaded\_file($tmp\_name, $location)) { | |  | true; | |  | }else {$message= " <div class='alert alert-danger alert-dismissible'> | |  | <button type='button' class='close' data-dismiss='alert'>&times;</button> | |  | <strong> فشل</strong> يوجد خطأ في حفظ الملف | |  | </div> "; | |  | $chechUpload=0; | |  | } | |  | } | |  |  | |  | if ($chechUpload=1||$chechUpload=0){ | |  | $max\_size = 100000; | |  | if ($size <= $max\_size) { | |  | // check the type of image | |  | if ($type=="image/jpg"|| $type=="image/JPG" ||$type=="image/jpeg"|| $type=="image/JPEG"){ | |  |  | |  | if($chechUpload!=0){ | |  | // add info of new badge to the DB | |  |  | |  | $visitorName = $\_POST["name"];//position | |  | $visitorCareer = $\_POST["career"];//position | |  | $visitorBarcode = $\_POST["barcode"];//position | |  | $imgPosition= $\_POST["imgPosition"];//position | |  | $color=$\_POST["color"]; | |  | $barSize=$\_POST["barSize"]; | |  | $fontSize=$\_POST["fontSize"]; | |  |  | |  | $sql = mysqli\_query($con, "UPDATE badge set BadgeTypeId='$badgeTypeId',event\_ID='$eventId',badgeTemplateName='$name',badgeTemplateSize='$size',badgeTemplateType='$type',badgeTemplateLocation='$location' | |  | where badge\_ID='$badgeIdForEdit' ") or die(mysqli\_error($con)); | |  |  | |  | $sqlimage = mysqli\_query($con, "UPDATE imageinfo set color='$color' ,barSize='$barSize' ,fontSize='$fontSize' ,badgeId='$badgeIdForEdit',namePosition='$visitorName ',careerPosition='$visitorCareer', | |  | barcodePosition='$visitorBarcode',imgPosition='$imgPosition' where imageId='$imageinfoId'") or die(mysqli\_error($con)); | |  | ///Check if add badge to DB has been done Successfully | |  | if ($sql && $sqlimage) { | |  | header("location: /tactic/manageBadge.php"); | |  | } else { | |  | $message=" <div class='alert alert-danger alert-dismissible'> | |  | <button type='button' class='close' data-dismiss='alert'>&times;</button> | |  | <strong> فشل</strong> لم تتم عملية التعديل بنجاح يرجى التحقق | |  | </div> "; | |  | } | |  | } | |  | } else { | |  | // for size message | |  | $message= " <div class='alert alert-danger alert-dismissible'> | |  | <button type='button' class='close' data-dismiss='alert'>&times;</button> | |  | يرجى التحقق من صيغة الملف يجب ان تكون من نوع (JPG)(JPEG) | |  | </div> "; | |  | } | |  | } else | |  | $message= " <div class='alert alert-danger alert-dismissible'> | |  | <button type='button' class='close' data-dismiss='alert'>&times;</button> | |  | تنبيه أكبر حجم للملف هو 10 ميغا | |  | </div> "; | |  | }//end | |  | }// end else | |  | } //end add | |  | ?> | |  |  | |  | <!DOCTYPE html> | |  | <html lang="ar"> | |  |  | |

**Table 6-8: add certificate function code**

|  |
| --- |
| **Function name:** add certificate |
| **Function description:** this code allow the organizer to add certificate for each attendee who attend |
| |  | | --- | |  | |  | // this section for get the event name fro DB $query = mysqli\_query($con, "SELECT \* FROM event where organizer\_ID= '$organizerid'") or die(mysqli\_error()); | |  |  | |  | if (isset($\_POST['add']) && !empty($\_FILES["fileToUpload"]["name"])) { | |  |  | |  | $eventId = $\_POST['eventId']; | |  |  | |  | $checkQuery = mysqli\_query($con, "SELECT \* FROM certificate WHERE event\_ID='$eventId'") or die(mysqli\_error()); | |  |  | |  | if (mysqli\_num\_rows($checkQuery)>0) { | |  | echo " <div class='alert alert-danger alert-dismissible'> | |  | <button type='button' class='close' data-dismiss='alert'>&times;</button> | |  | <strong> فشل</strong> الحدث مرتبط بشهادة مسبقا لايمكن اتمام العملية | |  | </div> "; | |  |  | |  | } else { | |  |  | |  | $name = $\_FILES['fileToUpload']['name']; | |  | $size = $\_FILES['fileToUpload']['size']; | |  | $type = $\_FILES['fileToUpload']['type']; | |  | $tmp\_name = $\_FILES['fileToUpload']['tmp\_name']; | |  | $location = "UploadFile/certificate/"; | |  |  | |  | $max\_size = 100000; | |  | if ($size <= $max\_size) { | |  | if (move\_uploaded\_file($tmp\_name, $location . $name)) { | |  |  | |  | // add info of new badge to the DB | |  |  | |  | $sql = mysqli\_query($con, "INSERT INTO certificate | |  | (event\_ID,templateName, | |  | templateSize,templateType, templateLocation) | |  | VALUES ('$eventId','$name' ,'$size', '$type', '$location$name')") or die(mysqli\_error($con)); | |  | ///Check if add badge to DB has been done Successfully | |  | if ($sql) { | |  | header("location: /tactic/manageCertificate.php"); | |  | } else { | |  | echo " <div class='alert alert-danger alert-dismissible'> | |  | <button type='button' class='close' data-dismiss='alert'>&times;</button> | |  | <strong> فشل</strong> لم تتم عملية الاضافة بنجاح يرجى التحقق | |  | </div> "; | |  | } | |  | } else { | |  | echo " <div class='alert alert-danger alert-dismissible'> | |  | <button type='button' class='close' data-dismiss='alert'>&times;</button> | |  | <strong> فشل</strong> يوجد خطأ في حفظ الملف | |  | </div> "; | |  |  | |  | } | |  | } else { | |  | // for size message | |  | echo " <div class='alert alert-danger alert-dismissible'> | |  | <button type='button' class='close' data-dismiss='alert'>&times;</button> | |  | <strong> يرجى</strong> أكبر حجم للملف هو 10 ميغا | |  | </div> "; | |  |  | |  |  | |  |  | |  |  | |

**6.4.1.2 Organizer privilege(app):**

As we said before that the organizer could do many tasks on the application, in the figure bellow most of the important functions he / she able to do.

**Figure 6-3 : organizer privilege on app interfaces**

Log in

Scan barcode

Print badge

**Table 6-9: print badge code**

|  |
| --- |
| **Function name:** Print Badge. |
| **Function description:** This function allow the organizer to print a badge for the attendee . |
| @Override **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.**activity*\_print***);  **eventID** = getIntent().getExtras().getString(**"event\_ID"**,**""**);  **Attendee\_ID** = getIntent().getExtras().getString(**"Attendee\_ID"**,**""**);  **final** ImageView imageView = findViewById(R.id.**imageView**);   **printBTN** = (Button) findViewById(R.id.***printBTN***);  **url** = **"http://192.168.1.5/UploadFile/"**+**eventID**+**"/badge/"**+**Attendee\_ID**+**".jpg"**;  Picasso.*get*().load(**url**).into(imageView);  **printBTN**.setOnClickListener(**new** View.OnClickListener() {  @Override  **public void** onClick(View view) {  BitmapDrawable drawable = (BitmapDrawable) imageView.getDrawable();  printImage(**url**, drawable.getBitmap());  }  });  } **private void** printImage(String fileName, Bitmap bitmap) {  **if** (PrintHelper.systemSupportsPrint()) {  PrintHelper printHelper = **new** PrintHelper(**this**);  printHelper.setColorMode(PrintHelper.***COLOR\_MODE\_COLOR***);  printHelper.setScaleMode(PrintHelper.***SCALE\_MODE\_FIT***);  printHelper.printBitmap(fileName, bitmap);  } **else** {  Toast.*makeText*(**this**, **"error in printing"**, Toast.**LENGTH*\_SHORT***).show();  } } |

**Table 6-10: scan barcode code**

|  |
| --- |
| **Function name:** Scan Barcode |
| **Function description:** This function allow the organizer to scan the attendee barcode for tracking . |
| **public class** ScanActivity **extends** AppCompatActivity **implements** ZXingScannerView.ResultHandler{  **private static final int *REQUEST\_CAMERA*** = 1;  **private** ZXingScannerView **scannerView**;  String **Attendee\_ID**;  InputStream **is**=**null**;  String **result**=**null**;  String **line**=**null**;  **private static int** came= Camera.CameraInfo.***CAMERA\_FACING\_BACK***;  **int code**;   **public** String **eventID**,**subevent\_ID**;  String **attendeeName**;  **private** String **URL\_AttendeeName** = **"http://192.168.1.5/hi/getAttendeeName.php?Attendee\_ID="**;  **private** String **url\_update**=**"http://192.168.1.5/hi/update.php"**;  **public static** String title ;  @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.**activity\_scan**);   **eventID** = getIntent().getExtras().getString(**"eventID"**,**""**);  **subevent\_ID** = getIntent().getExtras().getString(**"eventIdSubSelect"**,**""**);  **scannerView** = **new** ZXingScannerView(**this**);  setContentView(**scannerView**);   **int** currentApiVersion = Build.VERSION.**SDK\_INT**;   **if**(currentApiVersion >= Build.VERSION\_CODES.**M**)  {  **if**(checkPermission())  {  // Toast.makeText(getApplicationContext(), "Permission already granted!", Toast.LENGTH\_LONG).show();  }  **else** {  requestPermission();  }  }  }   **private boolean** checkPermission()  {  **return** (ContextCompat.checkSelfPermission(getApplicationContext(), **CAMERA**) == PackageManager.**PERMISSION\_GRANTED**);  } |

6.5 SYSTEM INTEGRATION:

In this section, we will show how the component of our system will be integrated together. At this level which is the first we divided the function into two groups: First, Group 1 contains the function of Attendee. Then, Group 2, contains the function of Organizer.

**Level 1:**

**Figure 6-4 : system integration on website (level 1)**

Edit profile

Manage SubEvent

Manage Prize

Prize winner

Register

Manage Badge

Register into the form

Log In

Manage Certificate

Confirm Register

Reset Password

Log Out

Manage Event

Manage Form

Manage OnSite

Manage Rating

Manage Email

Group #1

Group #2

Rate the event/subevent

Statistics

**Figure 6-5 : system integration on app (level 1)**

Log In

Rate event/subevent

Print badge

Scan barcode

Track attendee

Group #2

Group #1

At level two, we're going to combine each group component with each other. So, each group has its own functionality cooperating together of the end of the system.

**Level 2**:

**Figure 6-6 : system integration on website (level 2).**

Organizer privilege group #2

Register

Attendee privilege group #1

Confirm Register

Statistic

Manage Email

Reset password

Manage badge

Manage subevent

Manage form

Manage Onsite

Manage rate

Manage certificate

Manage event

Manage prize

Edit profile

Log out

Log In

Attendee Register into the form

Chose Prize winner

Attendee Rate the event/subevent

**Figure 6-7 : system integration on app (level 2).**

Organizer privilege

Log In

Attendee privilege

Print badge

Scan barcode

Rate event/subevent

Track attendee

At the last level, we're going to integrate the component of group one(functions of the attendee) with group two (functions of the organizer)to reach that all the system components integrated together **.**

**Level 3**:

**Figure 6-8 : system integration on website (level 3).**

Home page

Confirm Register

Register

Statistic

Manage Email

Prize winner

Reset password

Manage badge

Attendee Register into the form

Attendee Rate the event/subevent

Manage subevent

Manage form

Manage Onsite

Manage rate

Manage certificate

Manage event

Manage prize

Edit profile

Log out

Log In

**Figure 6-9 : system integration on app (level 3).**

Home page

Log In

Print badge

Track attendee

Scan barcode

Rate event/subevent

6.6 SUMMARRY:

After the sections completion of the implementation and integration chapter, we give a detailed description about the hardware and software used, the difficulties and main problems we faced while we are implementing our system and we mention the main functions with their code. Finally, the components of the system and how they integrated together.

CHAPTER 7

SYSTEM TESTING

7.1 INTRODUCTION:

In this chapter, we will describe the stages of testing the system. The purpose of this chapter to ensure that our website/application working as we predicted, effective, free of mistakes and problems and work under the requirements. Diverse tests will be under considerations which are: unit testing, integration and regression testing, performance and stress testing, user acceptance testing and test cases testing.

7.2 UNIT TESTING:

In this section the individual units of the main functions for the website/application will be tested to determine if it works properly and as predicted.

|  |  |  |  |
| --- | --- | --- | --- |
| **User** | **Component Name** | **Component description** | **Is working as expected?** |
| **Main function** | Component A | Register | Pass |
| Component B | Confirm register | Pass |
| Component C | Log in | Pass |
| Component D | Update profile | Pass |
| Component E | Log out | Pass |
| Component F | Forgot password | Pass |
| **Organizer** | Component G | Manage event | Pass |
| Component H | Add event | Pass |
| Component I | Edit event | Pass |
| Component J | Show event details | Pass |
| Component K | Manage Subevent | Pass |
| Component L | Add subevent | Pass |
| Component M | Edit subevent | Pass |
| Component N | Show subevent details | Pass |
| Component O | Manage badge | Pass |
| Component P | Add badge | Pass |
| Component Q | Edit badge | Pass |
| Component R | Manage prize | Pass |
| Component S | Add prize | Pass |
| Component T | Edit prize | Pass |
| Component U | Show prize details | Pass |
| Component V | Chose Prize winner | Pass |
| Component W | Manage Certificate | Pass |
|  | Component X | Add certificate | Pass |
| Component Y | Edit certificate | Pass |
| Component Z | Manage ,create, edit form | Pass |
| Component A1 | Manage onsite,onsiteRegister | Pass |
| Component B1 | Manage ,send email | Pass |
| Component C1 | Statistics | Pass |
| Component D1 | Manage rate | Pass |
| Component E1 | Manage track Attendee | Pass |
| **Attendee** | Component F1 | Register into form | Pass |
| Component G1 | Receive email | Pass |
| Component H1 | Rate event/subevent | Pass |

**Table 7-1: Unit Testing** **for website**

**Table 7-2: Unit Testing** **for app**

|  |  |  |  |
| --- | --- | --- | --- |
| **User** | **Component Name** | **Component description** | **Is working as expected?** |
| **Organizer** | Component A | Log In | Pass |
| Component B | Print badge | Pass |
| Component C | Scan barcode | Pass |
| Component D | Track attendee | Pass |
| **Attendee** | Component D | Rate event/suevent | Pass |

7.3 INTEGRATION AND INTERACTION TESTING:

**7.3.1 INTEGRATION TESTING:**

In the previous section, the components have been tested separately. We divided them into two groups, group 1 hold the components of Attendee and group 2 contains the components of Organizer. In this section, we will integrate the different components for the website and the application separately to ensure the correctness of the system as all and thus we integrate and test the component of the organizer with the main functions. After that, we integrate and test the attendee components. At the end to conclude the testing, the integration of group 1 is combined with group 2.

First, registration (main function) :

**Table7.3: integration testing for registration function** (**website**)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Test sequence** | | | |
| **Test number** | Test 1 | Test 2 | Test 3 | Test 4 |
| **Component** | Component A,B and C | Component of test1 with D | Component of test2 with E | Component of test3 with F |
| **Description** | Test registration with confirmation and log in. | Test update profile with component of Test 1 | Test log out with component of Test 2 | Test Forgot password with component of Test 3 |
| **Graph** | T1  C  B  A | T2  T1  D  C  B  A | T3  E  D  T2  T1  C  B  A | T4  T3  T2  F  E  D  T1  C  B  A |
| **Status** | Pass | Pass | Pass | Pass |

Second, group 2 (Organizer user):

**Table7.4:integration testing for Organizer.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Test sequence** | | | |
| **Test Number** | Test 5 | Test 6 | Test 7 | Test 8 |
| **Component** | Component G,H,I,J | Component K,L,M,N | Component O,P,Q | Component R,S,T,U,V |
| **Description** | Test manage ,add, edit, show details of event. | Test manage ,add, edit, show  details of subevent with Test5 | Test manage ,add, edit,  badge with Test6 | Test manage,add,edit,show details, chose winner of prize with Test7 |
| **Graph** | T5  J  I  H  G | N  L  M  T6  K  T5  J  I  H  G | I  K  N  M  L  T6  T5  J  H  G  T7  O  P  Q | U  T  S  R  V  P  Q  O  N  L  M  K  T8  T7  H  T6  T5  J  I  G |
| **Status** | Pass | Pass | Pass | Pass |

**Table7.5:integration testing for Organizer-1**

|  |  |  |
| --- | --- | --- |
|  | **Test sequence** | |
| **Test Number** | Test 9 | Test10 |
| **Component** | Component of Test 8 and W,X,Y | Component of Test 9 and Z. |
| **Description** | Test Manage, add, edit certificate with Test8 | Test Mange, create, edit form with Test9 |
| **Graph** | P  Q  T9  Y  X  W  I  H  G  T8  T7  T6  T5  J  O  V  N  L  M  K  R  S  T  U | T10  Z  I  H  G  T6  T5  J  L  T7  P  K  T9  Y  X  W  T8  Q  O  N  M  V  U  T  S  R |
| **Status** | Pass | Pass |

|  |  |  |
| --- | --- | --- |
|  | **Test Sequence** | |
| **Test Number** | Test 11 | Test 12 |
| **Component** | Component of Test 10 with A1. | Component of Test11 with B1. |
| **Description** | Test manage onsite,onsiteRegister with component of Test10 | Test manage, send email with component of Test11 |
| **Graph** | T7  T10  T11  Z  A1  T9  Y  X  W  T  T8  V  U  S  R  T6  Q  P  O  N  L  M  K  H  T5  G  J  I | T7  T9  T10  T12  B1  T11  A1  Z  Y  X  W  T8  V  U  T  S  R  T6  Q  P  O  N  L  M  K  I  H  G  T5  J |
| **Status** | Pass | Pass |

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Test Sequence** | | |
| **Test Number** | Test13 | Test14 | Test15 |
| **Component** | Component of Test12 with C1. | Component of Test13 with D1. | Component of Test14 with E1. |
| **Description** | Test Statistics with component of Test12 | Test Manage rate with component of Test13 | Test Manage track attendee with component of Test14 |
| **Graph** | T5  T6  T7  W  T13  A2  T8  V  U  T  S  R  Q  O  P  N  L  M  K  J  I  H  G | T5  T6  T7  T13  A2  W  T8  V  U  T  S  R  Q  O  P  N  L  M  K  G  I  H  J | T5  T6  T7  T8  V  W  U  T  S  R  Q  O  P  N  L  M  K  G  I  H  J |
|  | T12  B1  T9  A1  Z  T10  T11  Y  X  T13  C1 | T14  T13  D1  C1  T12  B1  T11  T10  T9  A1  Z  Y  X | T15  T14  T13  E1  D1  C1  T12  T11  T10  T9  B1  A1  Z  Y  X |
| **Status** | Pass | Pass | Pass |

**Table7.6: integration testing for Attendee**

|  |  |  |
| --- | --- | --- |
|  | **Test sequence** | |
| **Test Number** | Test16 | Test17 |
| **Component** | Component of F1 and G1 | Component of Test16 and H1 |
| **Description** | Test receive email and register into the form | Test rate the event/subevent |
| **Graph** | T16  G1  E1 | T17  H1 |
| **status** | Pass | Pass |

**Table7.7:integration testing for Organizer on application**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Test sequence** | | | |
| **Test Number** | Test 1 | Test 2 | Test 3 | Test 4 |
| **Component** | Component A | Component of Test1 with B | Component of Test2 with C | Component of Test3 with D |
| **Description** | Test Log In | Test print Badge with Test1 | Test Scan barcode with Test2 | Test Track attendee with Test3 |
| **Graph** | T1  A | T2  T1  B  A | T4  C  T2  T1  B  A | T4  T3  D  C  T2  T1  B  A |
| **Status** | Pass | Pass | Pass | Pass |

**Table7.8: integration testing for Attendee on application**

|  |  |
| --- | --- |
|  | **Test sequence** |
| **Test Number** | Test5 |
| **Component** | Component D |
| **Description** | Test Rate event/subevent |
| **Graph** | D  T5 |
| **Status** | Pass |

Now we insure that all the components are working successfully after the integration been done.

**7.3.2 REGRESSION TESTING:**

Regression testing is the process of testing changes to the system to make sure that the older function still works with the new changes. We applied the regression testing after any required changes been done to ensure that the system still workable.

# 7.4 PERFORMANSE AND STRESS TESTING:

Performance Testing is a type of testing to ensure software applications will perform well under their expected workload. Furthermore, stress testing is used to test the reliability of the system. Also, it determines the system error handling under extremely heavy load and unusual conditions. <https://www.guru99.com/performance-testing.html>

**7.4.1 PERFORMANCE TESTING:**

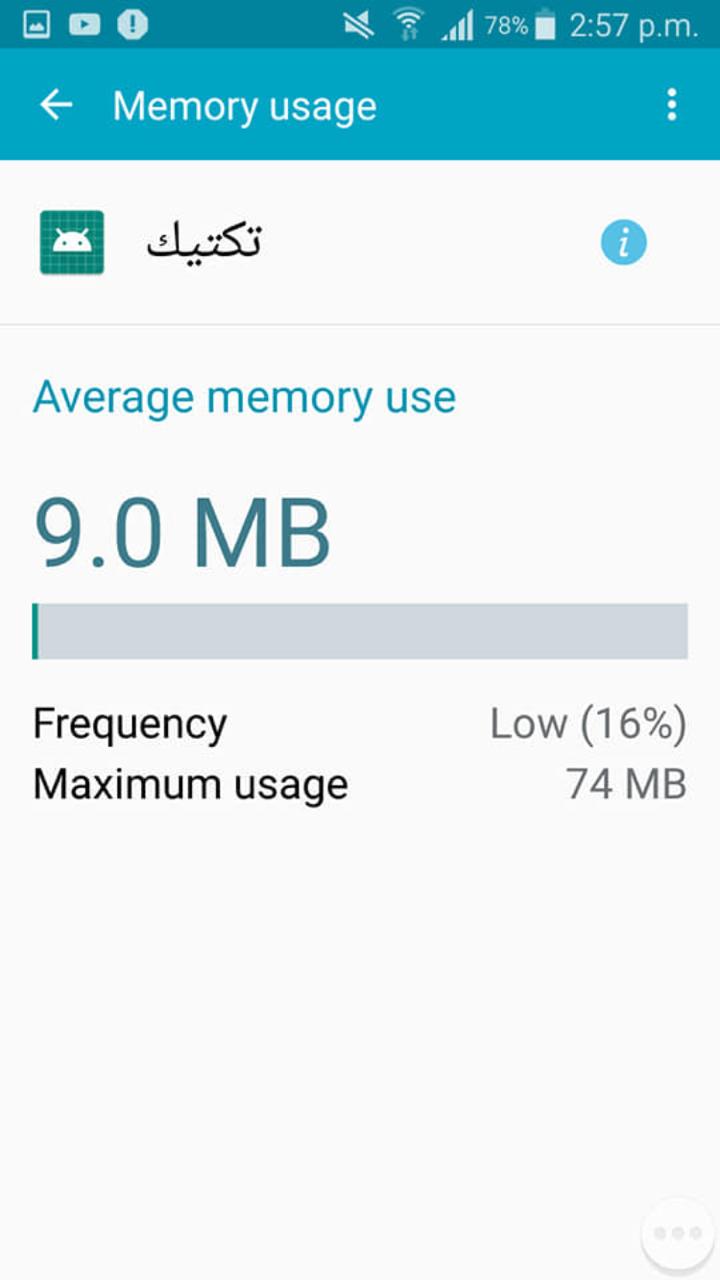
Under this section, we will show the memory usage space, the storage, and the disk allocation that the application takes for monitoring the performance and the behavior of the Android application.

1. Memory usage for application :

In Figure 7.1, [9.0 MB] shows the average usage for the application memory with the

[74 MB] maximum of usage and [16%] frequency.

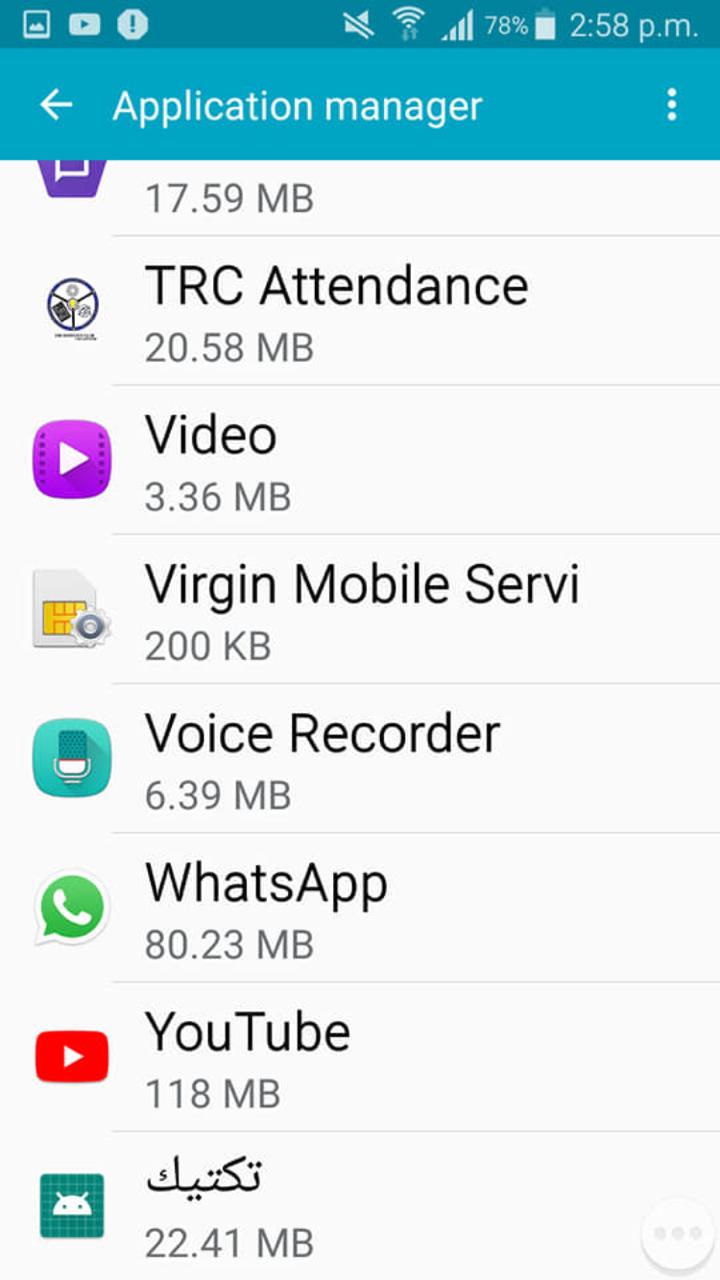
**Figure7-1: Application memory use**



1. Disk Allocation for application :

"تكتيك" used 22.41 MB of the disk, which makes it appropriate and suitable for use comparing with other applications.

**Figure7-2: Disk Allocation**

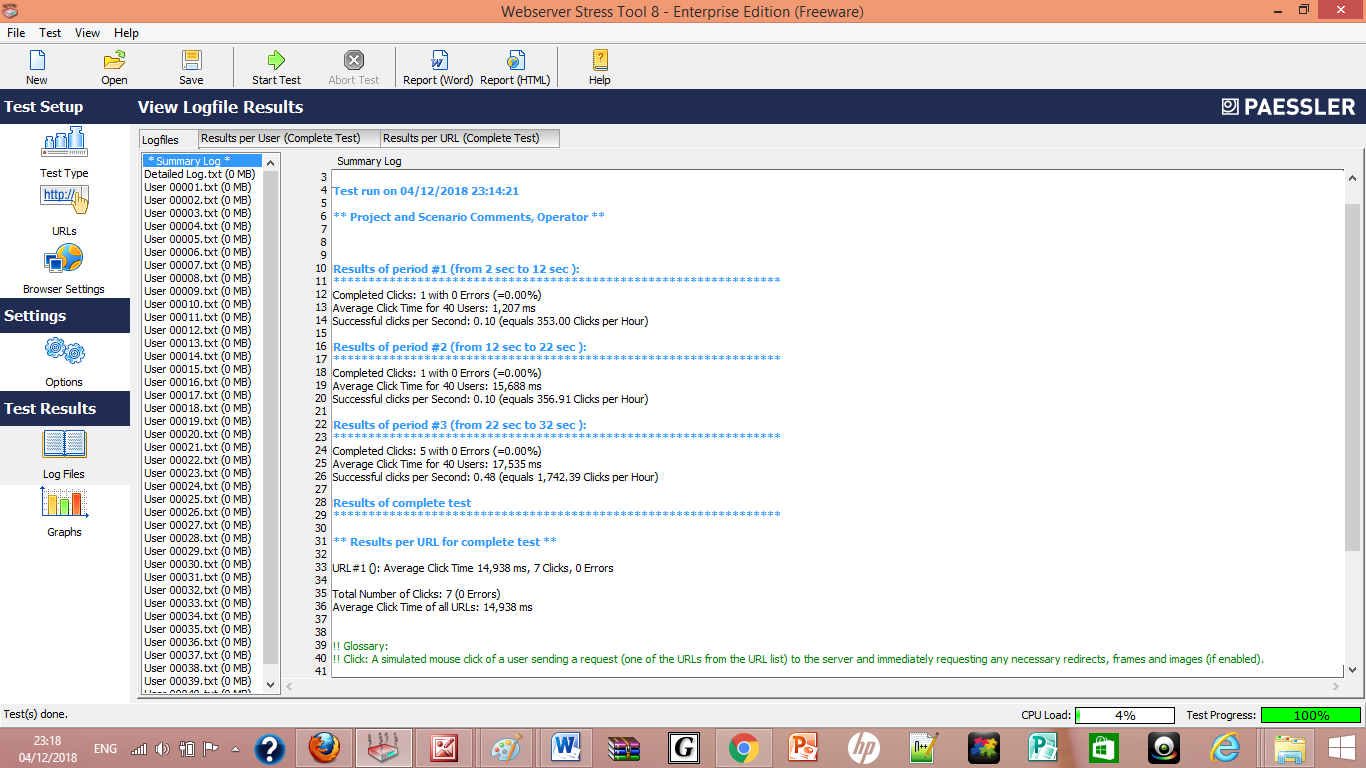


* + 1. **STRESS TESTING:**

On the other hand for the website, Web Server Stress tool was used to identify if there is any tolerate stress and to monitor the percentage of the site to withstand the pressure.

1. Stress testing for website :

To determine whether the website afford the pressure or not, we assumed that it was used by 40 users so on the result was satisfying and the rate of error is negligible.



**Figure7-3: stress testing for website**

Customarily, when our website handles objects such as images, it should keep track of these objects to reduce the number of possible errors as possible. This figure show handling the loading of images and its effect on the website.

****

**Figure7-3: Times and errors for website**

* 1. USER ACCEPTANCE TESTING:

User acceptance testing (UAT) is the last phase of the software testing process. During UAT, actual users test the system to make sure it can handle required tasks according to specifications and to ensure the correctness of the entire system before release it. Also, to obtain satisfaction for users depending on effectiveness and efficiency after testing the system.

<https://usersnap.com/blog/user-acceptance-testing-right/>

The number of users who participated in the acceptance testing is six: three of them represent the role of the organizer and the others represent the attendee for both the website and application. The ages of the participants are between 22-27.

Group1 (organizer):

**Table7.9: organizer ages and genders**

|  |  |  |
| --- | --- | --- |
| **Participant number** | **Age** | **Gender** |
| **1** | 23 | Female |
| **2** | 25 | Female |
| **3** | 24 | Female |

**Table7.9: Measure system effectiveness (organizer)**

|  |  |  |
| --- | --- | --- |
| **Tasks** | **Average number of errors** | **Average time in seconds** |
| **Register** | 0 | 58.6 |
| **Log In** | 0 | 24.5 |
| **Reset password** | 0 | 40.4 |
| **Edit profile** | 0 | 57.9 |
| **Manage event** | 0.3 | 60.0 |
| **Manage subevent** | 0 | 58.9 |
| **Manage badge** | 0.3 | 55.4 |
| **Manage prize** | 0.1 | 59.9 |
| **Manage certificates** | 0.2 | 64.3 |
| **Manage form** | 0 | 65.9 |
| **Manage onsite registration** | 0 | 50.3 |
| **Manage email** | 0 | 70.5 |
| **Statistics** |  |  |
| **Manage rate** |  |  |
| **Manage track attendee** |  |  |
| **Log out** |  |  |