Applied Information Technology And Computer Science Vol. 5 No. 1 (2024) 143-162

AITCS

Homepage: http://publisher.uthm.edu.my/periodicals/index.php/aitcs
e-ISSN :2773-5141

A Kindergarten Management System with Dual Authentication for the City Kindergarten

Naelah Farafisyah Yahaya[1], Nor Bakiah Abd Warif[1]*

[1] Faculty of Computer Science and Information Technology,
Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA

DOI: https://doi.org/10.30880/aitcs.2024.05.01.009
Received 27 September 2023; Accepted 23 May 2024; Available online 30 August 2024

Abstract: School Management System is a system to organize school administration, including student and teacher data, class timetable, student's examination result and school programs. Although the times are evolving, most of the kindergarten schools in Malaysia still use manual ways to store the data, such as storing the students' information in excel files, and update the students' progress in a report card. As a result, the data was easily lost and stolen. The vulnerability of the issue led to the invention of The City Kindergarten management system. The goal of this project is to create an online management system with dual authentication for The City Kindergarten's three target users: admin, teachers, and parents. Users can use this system to register teachers and students, manage class, timetable, and events. Dual authentication used in this system are strong password and email One-Time Password (OTP) for administrators and teachers, and strong password and email verification for parents. Besides that, this system enables users of The City Kindergarten to store the data securely. The methodology used in the project is a prototype model. Programming languages used in this system are HTML, CSS, PHP, and JavaScript. Several user testing are conducted involving two sections, section A for interface and section B functionality. Based on their feedback, it can be concluded that 4 out of 5 marks has been given, and the system is functioning effectively and satisfies their requirements.

Keywords: Management System, Dual Authentication, Web Based System, The City Kindergarten

1. Introduction

A school management system is an information system used by academic institutions to handle student information. It enables teachers to obtain information on students more quickly and easily, reducing their workload. School management systems prioritise exchange of information and online communication among instructors, students, and parents. [1]

The current management system in The City Kindergarten is conducted manually. The information of students was kept in excel files, and their results were updated in a report card. Other than that, the timetable for the teacher was generated manually, by handling it through the hard copy. Since personal information is among the most crucial types of information, this might result in data loss and theft.

*Corresponding author: norbakiah@uthm.edu.my

Furthermore, parents must visit school to obtain their children's scores. Some parents want to see their children's learning progress; thus, this system will record all assessment and test results from the start of school until the end of the preschool session.

The objectives of this project are to design, develop, and evaluate The City Kindergarten Management System with Dual Authentication. In this system, it only requires three roles which are administrators, teachers, and parents. Each of the roles have their own privilege access to the system. This system was designed with numerous security elements to make the data in the system more secure, including dual authentication which are strong password and email One-Time Password (OTP) for admin and teachers, strong password and email verification for parents, password hashing, and Role-based Access Control (RBAC).

The City Kindergarten Management System is a web-based system to utilize the management regarding The City school. The purpose of this system is to improve school-related information management to be more structured. This system is used for storing teachers' and students' details, managing the timetable for teachers, updating the students' results and activities related to the school. This system allows management to gather all information in one place.

2. Literature Review

This section explains the online school management system, authentication and some similar systems that will be compared with the proposed system.

2.1 Online School Management System

A school management system is a data management system used by educational institutions to handle information of students. It enables teachers to inputting daily information about their lessons, including topics covered, educational materials used, assigned homework, as well as details pertaining to their students, such as attendance records, disciplinary matters, homework completion, and grading [2]. This can involve activities like student registration, class timetabling, grading, and logbook. A school management system may also contain methods for communication with students, parents, and instructors, such as email or online communication. These technologies are created to simplify and organize many of the administrative activities associated with running a school, and they can assist to increase efficiency and effectiveness.

2.2 Authentication

Authentication technique controls system access by determining if a user's credentials match those in a database of authorized users or in a data authentication server. An account secured with dual authentication typically requires an individual to authenticate using something they know—typically a password—as well as something they have, such as a cell phone [3] . Passwords, biometric techniques, and other kinds of authentication are all available. The precise approach utilized may be determined by the system's security requirements together with the user's choices and capabilities.

Dual authentication combines two types of authentication mechanisms to verify identification. This is done to protect the user's account and prevent unwanted access. The combination of a password plus an email one-time password (OTP), sometimes known as a one-time passcode, is a frequent method of dual authentication. Other than that, email verification is the method of determining whether an email address is legitimate and operational. Upon submission, an OTP will be generated and subsequently sent to the provided email address, which the user must enter to complete the process [4]. A verification email is an email sent as part of the verification procedure to a user's registered email address. Its goal is to validate or authenticate the ownership of the email address entered during user registration or creating an account. The verification email usually includes a one-of-a-kind link or a verification code.

To verify that they have access to the email account linked with the registration, the user is prompted by clicking on the link provided or inputting the verification code on the website.

2.3 Role-based Access Control

Role-based access control (RBAC) operates by linking permissions to specific roles and then assigning users to the relevant roles, granting them access based on the permissions associated with those roles [5]. Role-based access control (RBAC) is a way of managing access to system resources or actions based on the responsibilities of individual system users. Permissions to access resources or complete actions in RBAC are decided by the user's allocated role, not their individual identity.

2.4 Hashing

Password hashing is a type of hashing that is employed to securely store passwords. When users establish an account or set a password, the password is not saved instantly in its raw form. Instead, it passes through a procedure known as password hashing. Password hashing is used to safeguard passwords in the event of an information theft or unauthorized access to the password database. The original password is turned into a fixed-size hash value using a hash algorithm, which is then saved in the database. When a user tries to log in, the password is hashed via the same process, and the resultant hash value will be compared to the previously stored hash value. If they align, the password is accepted, and the user is permitted access.

2.5 Study on Existing System

This section explains two existing school systems, which are *Sistem Analisis Peperiksaan Sekolah* and *Sistem Maklumat Prasekolah Kebangsaan*.

2.5.1 *Sistem Analisis Peperiksaan Sekolah*

Sistem Analisis Peperiksaan Sekolah is a Malaysian government system that serves as a platform for obtaining student exam results. The system can be accessed by parents, from primary to secondary schools. The system is created to store the information of the students and their learning progress. This system consists of three roles, which are administrator of the school, teacher, and parents. This system only allows user to update the students' examination result.

Figure 1: Home page of *Sistem Analisis Peperiksaan Sekolah.* [6]

2.5.2 *Sistem Maklumat Prasekolah Kebangsaan*

Sistem Maklumat Prasekolah Kebangsaan is one of existing system, which is a web-based system. This system is an open system developed by our government, Ministry of Education [7] . The system can be access by all kindergartens and preschools in Malaysia. The system is created to store the information details of the students and teachers. This system consists of two roles, which are administrator of the school and teacher. It only allows the user to update the information of students and teachers and update the attendance of the students.

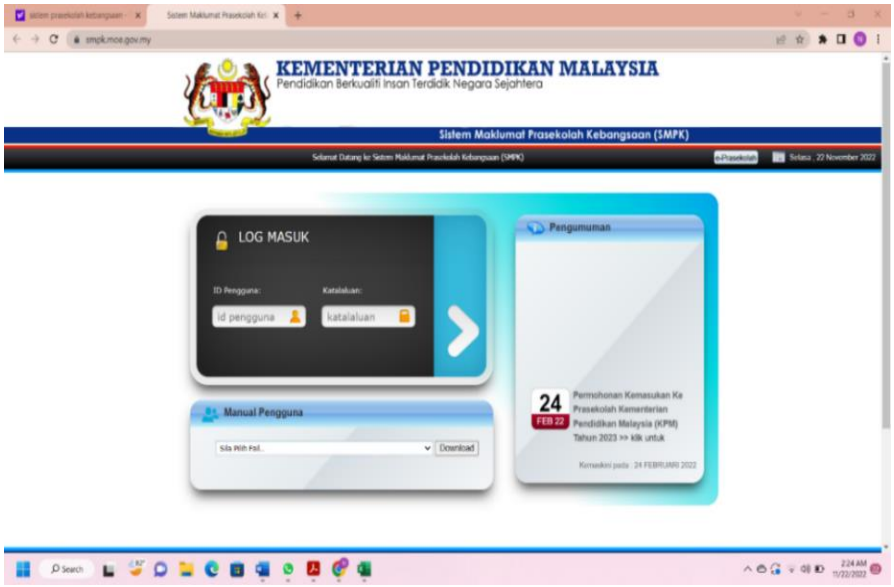

Figure 2: Home page of *Sistem Maklumat Prasekolah Kebangsaan.* [8]

The proposed system and existing system have been compared in table 1. Examples of existing systems are *Sistem Analisis Peperiksaan Sekolah* and *Sistem Maklumat Prasekolah Kebangsaan*. The features that have been compared for each system are type of the system, scope of user, dual authentication, admin login, parents' login, registration, hash and salt password, subject routing, and timetable management.

Table 1: Comparison of Existing System

Features	The City Kindergarten Manual System	*Sistem Analisis Peperiksaan Sekolah*	*Sistem Maklumat Prasekolah Kebangsaan*	Proposed System
Types of System	Manual	Online System	Online System	Online System
Scope of User	Administrator and teacher	Administrator, teacher, and parents	Administrator and teacher	Administrator, teacher, and parents
Dual Authentication	No	Yes, verification code	No	Yes, strong password and email OTP
Admin Login	No	Yes	Yes	Yes
Parents Login	No	Yes	No	Yes
Registration	Yes	Yes	Yes	Yes
Hash and Salt password	No	No Information Available	No Information Available	Yes
Subject routing	Yes	No	No	Yes
Timetable Management	No	No	No	Yes
Examination Result	Yes	Yes	No	Yes

Based on Table 1, the two existing systems which are The City Kindergarten Manual System, *Sistem Analisis Peperiksaan Sekolah* and *Sistem Maklumat Prasekolah Kebangsaan* have similarities and differences. For The City Kindergarten Manual System, it uses manual system while *Sistem Analisis Peperiksaan Sekolah* and *Sistem Maklumat Prasekolah Kebangsaan* are online system. For the scope, The City Kindergarten Manual System and *Sistem Maklumat Prasekolah Kebangsaan* have similar scopes, which are administrator and teacher only can use the system, while *Sistem Analisis Peperiksaan Sekolah* have similar scope with the proposed system, which are the administrator, teacher, and parents can access to the system. The City Kindergarten System does not have security features, while *Sistem Analisis Peperiksaan Sekolah* and *Sistem Maklumat Prasekolah Kebangsaan* did not provide any information about their security features. For timetable management, the existing systems did not provide the timetable for the teacher. All the security features will be applied to the proposed system. Furthermore, *Sistem Analisis Peperiksaan Sekolah* only provide the list of examination results of the students, while The City Kindergarten Manual System only recorded the information manually, which can lead to any loss of data and data theft. Thus, the proposed system will combine search forgether in one system, to make it easier for users to search related information in one place.

3. Methodology/Framework

The Prototype model is a part of the software development life cycle models that builds a prototype with minimum needs. To guarantee that the final system is performing, the prototype will be adjusted based on the client's evaluation. The final prototype, often known as the result, is a sort of entire system.

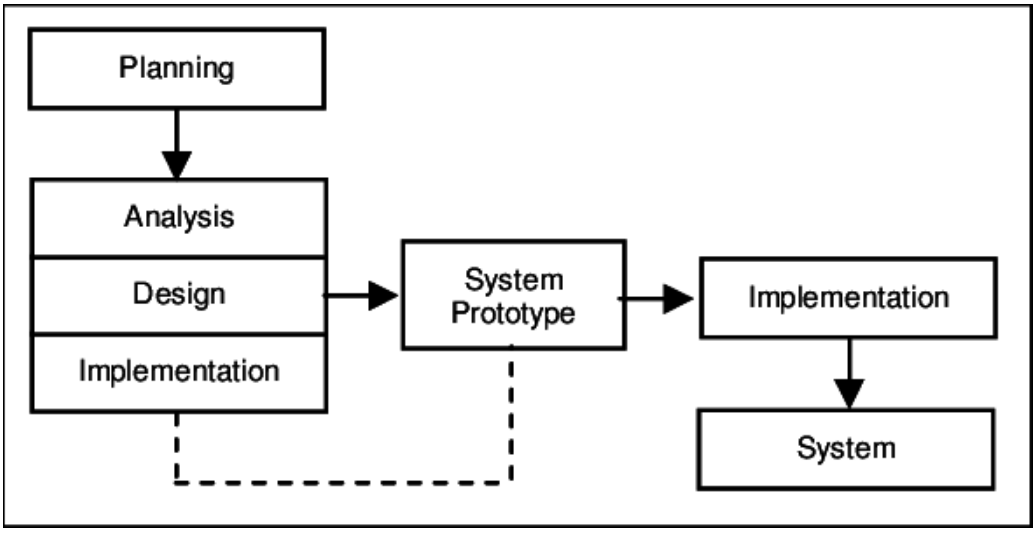

Figure 3: Phases in Prototype Methodology [9]

3.1 Planning Phase

Planning is the initial step in developing a prototype model [10] . During the procedure, system users are interviewed to learn about their expectations from the system. It plans the system's objectives throughout this phase to guarantee that they are achieved.

3.2 Analysis Phase

Analysis Phase in which an evaluation of the user of The City's kindergarten school management system is required. To ensure that the system is approved by users, all data must be collected from teachers and principals. Every piece of information gathered is critical for the construction of a system based on the information evaluated.

3.3 Design Phase

The design phase of a user system is the stage in which the system developer drafts the system design interface and builds a prototype. During this design phase, the user interface and database for The City's kindergarten school administration system were also created. The design phase is built on the knowledge gathered throughout the planning and analysis phases.

3.4 Implementation Phase

In this system, the Implementation Phase will be performed twice. This is due to if there is an issue with the system from the beginning, it may be fixed in the second phase of implementation. This phase marks the start of system development employing hardware, software, and programming.

3.5 System Prototype Phase

If the user is dissatisfied with the present prototype, the developer must enhance it based on the user's request and suggestions. Both the requirements and the prototype may be improved utilizing the input. This phase will not be completed until all of the user's needs have been satisfied. Once the user is impressed with the constructed prototype, a final system based on the authorized final prototype is created.

3.6 Testing Phase

The produced system will be evaluated throughout the testing phase to ensure that each navigation, menus, and security system works properly and can be utilized. This is due to its ability to suit the demands of users. The testing process is repeated several times. This is due to the fact that if the first round of testing is unsuccessful, the second step of testing will reveal improvements to the system.

4. System Analysis and Design

Functional requirements are the exact activities or operations that a system must be able to complete. They explain the functions that the system must be able to perform, as well as the input and output for each function. It also is an essential component of system requirement analysis since they assist specify the scope of the system and guarantee that it will fulfil the demands of the users. Non-functional requirements are traits or elements that a system should have but are not related to a particular function or activity performed by the system. Table 2 shows the functional requirement in the project, while table 3 shows the non-functional requirements.

Table 2: List of Functional Requirements for Kindergarten Management System with Dual Authentication for The City Kindergarten

Module	Functionalities
Login	• Users can login using their valid email address and password. • Only verified email address and password can enter into the system. • When login, users will receive email of OTP code (for admin and teachers) and verification link (for parents) to verify the users.
Registration	• Administrators can register teachers and students to the system. • Teachers can sign in using ID and password as registered. • Administrators can create, read, update, and delete (CRUD) the information of students and teachers.
Class Subject	• Administrators can add subjects and assign the subjects to the teachers. • Administrators can add class name and set the total of students in the class.
Timetable	• Administrators can generate the timetable for the teachers.
Exam	• Administrators can add the examination names. • Teachers can add student’s marks based on the examination types.

Table 3: List of Non-functional Requirements for Kindergarten Management System with Dual Authentication for The City Kindergarten

Module	Functionalities
Operation requirement	• The system can be browsed on a website. • Navigation, menus, and buttons should be simple to operate and located in correct place.
Implementation	• The need for the internet to ensure the system can operate smoothly.
Security	• Users need to be registered before using the system. • All users must login using verified ID and password. • Admin and Teacher must be verified by entering the OTP code that has been sent to their email. • Parents must be verified by clicking the unique link that has been sent to their email.

4.1 General System Architecture

Figure 4 shows the system architecture for an online school management system in which users will log in and the system will use a username and password. Then, the system will authenticate via dual authentication, which is email One-Time Password (OTP) for admin and teachers, while for parents is email verification. Users will go through their own dashboard based on their role. Admin can register teachers and students, add class and subject, generate the timetable, view events and examination results. For teachers, they can view timetable, update student's result and events, view their own information and students' information. Lastly, parents can view their own information and their child's, and view the events that are conducted in school.

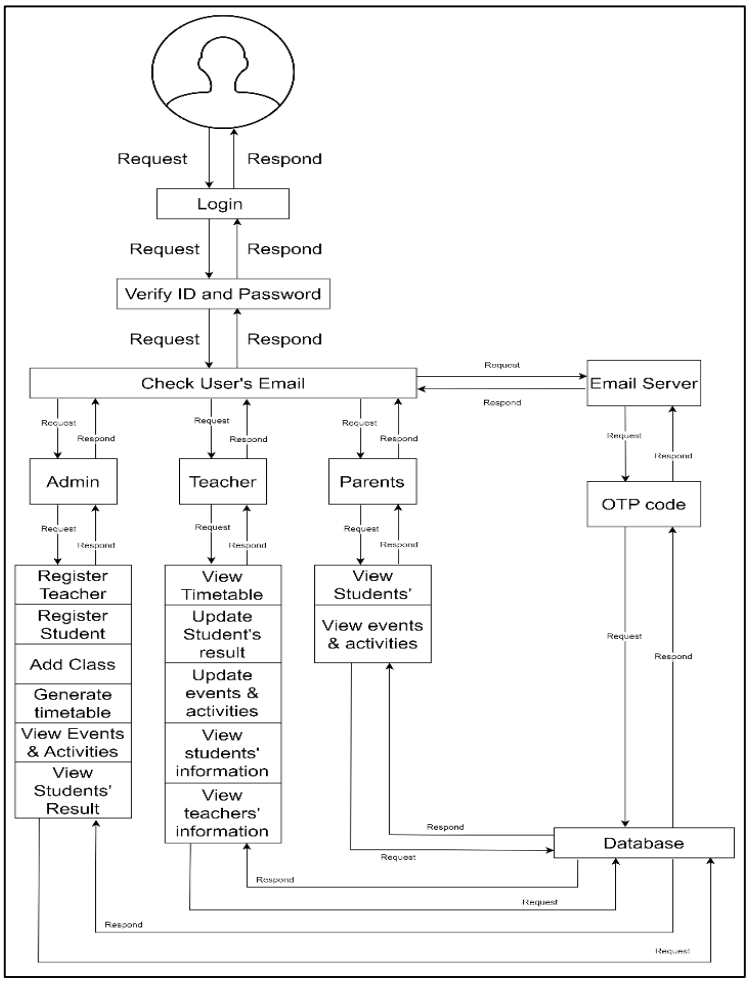

Figure 4: General System Architecture

4.2 Context Diagram & Data Flow Diagram (DFD)

A context diagram is an illustration that depicts the boundaries and restrictions of a system. It gives a summary of how the system engages with external elements such as users. A data flow diagram (DFD) depicts the flow of data through The City Kindergarten management system in graphical form. Figure 5 shows the Context Diagram, while Figure 6 shows the Data Flow Diagram that will illustrate the connections between the users such as admin, teachers, and parents with the system.

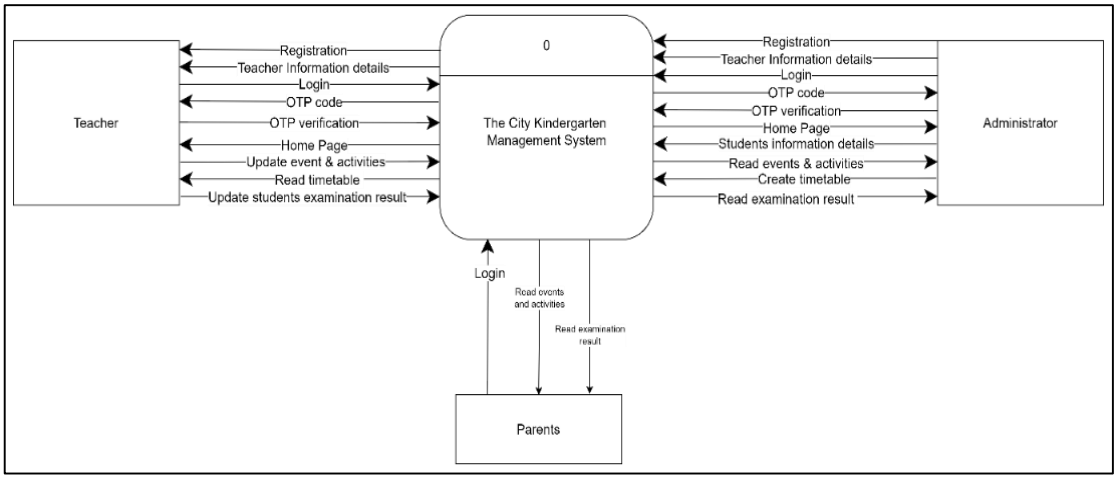

Figure 5: Context Diagram

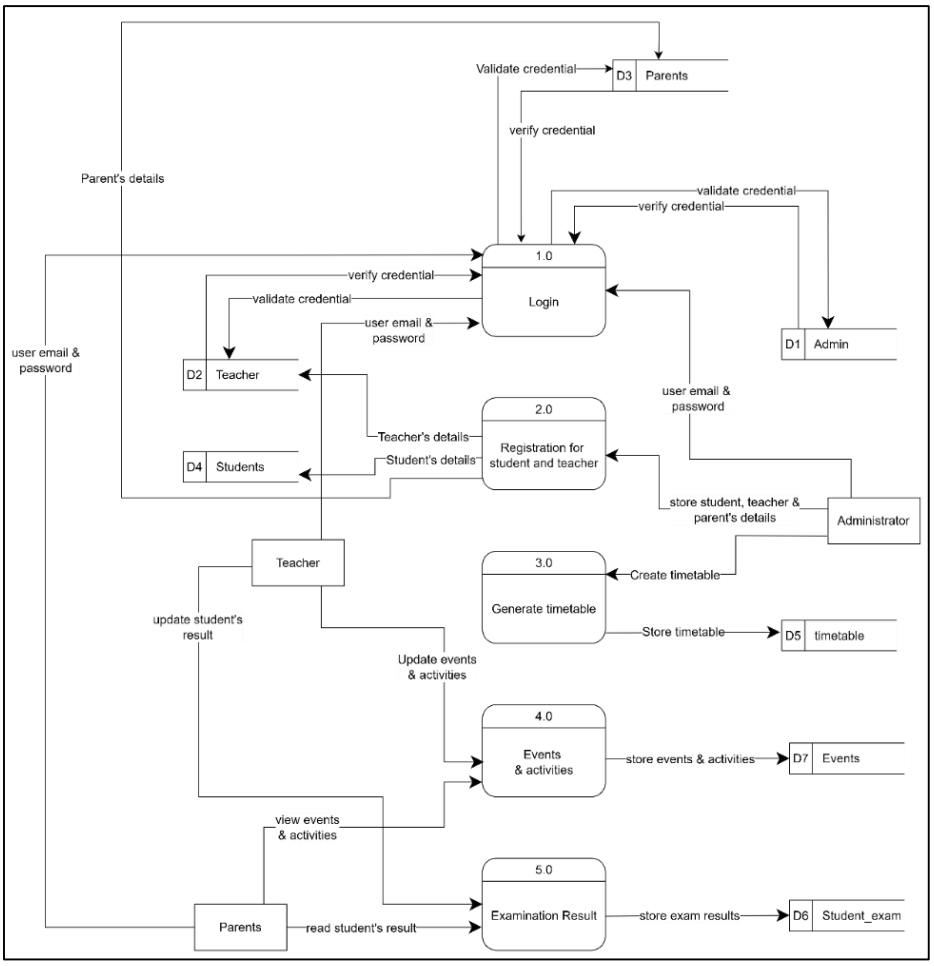

Figure 6: Data Flow Diagram at Level 0

4.3 Entity Relation Diagram (ERD)

Entity relationship diagram, or ERD, depicts the connections between entities in a database such as people, objects, or concepts [11] . It is frequently used in database design because it demonstrates the links between distinct data entities and can aid in identifying the data pieces that must be included in the database. Figure 7 shows the entity relationship diagram.

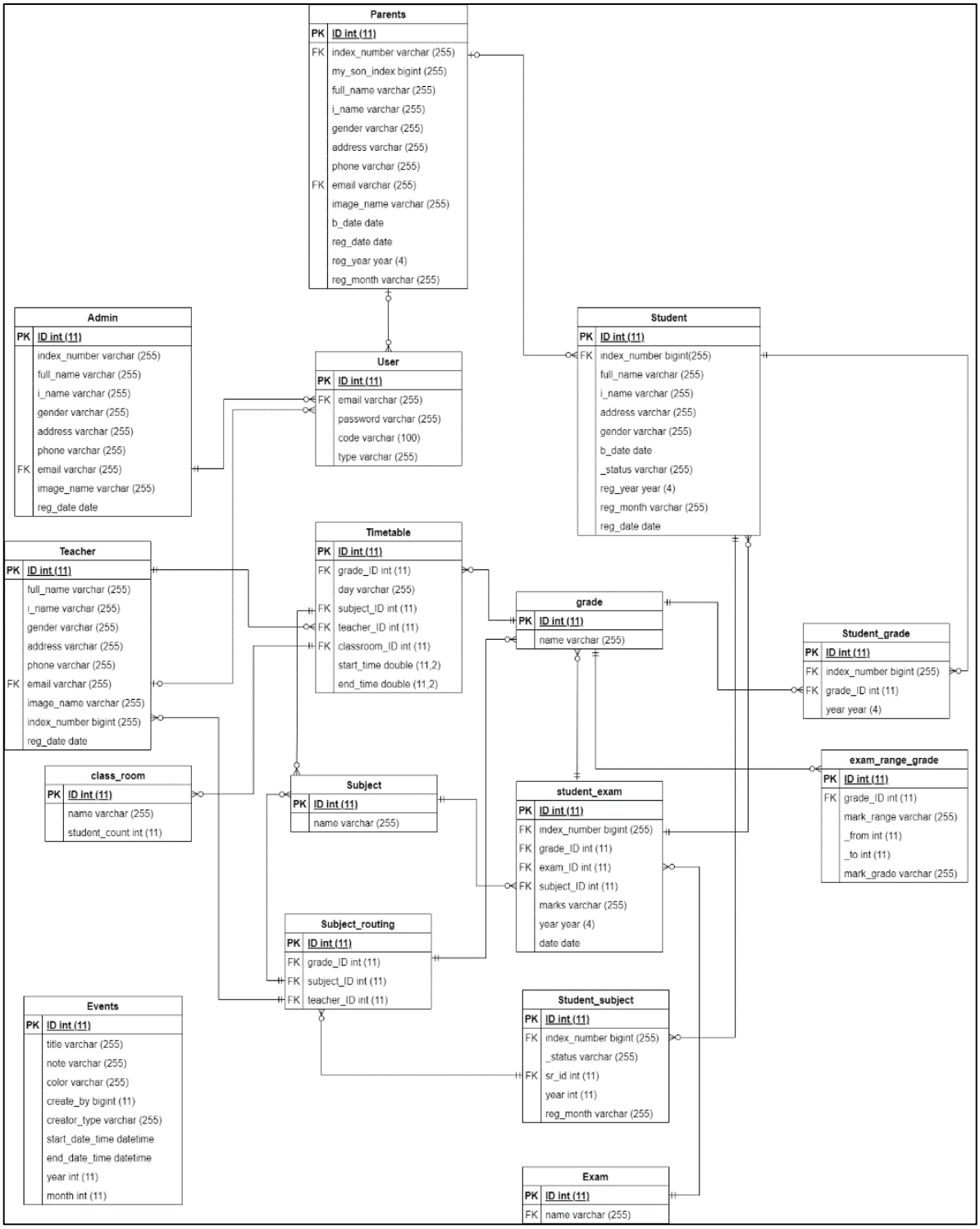

Figure 7: Entity Relation Diagram (ERD)

5. Implementation and Testing

The process of transforming a design or requirement into a functional software system is referred to as implementation. In contrast, testing is the procedure of reviewing a system to guarantee its efficiency, accuracy, and compliance with objectives.

5.1 Implementation

The modules applied in this system will be detailed in the following section. The modules will include registration and login, registration, class subject, schedule, and exam. Each module's partial program code will be detailed in its own section. Admin and teachers will receive an email One-Time Password (OTP) before login, while parents will receive email verification as shown in Figure 8.

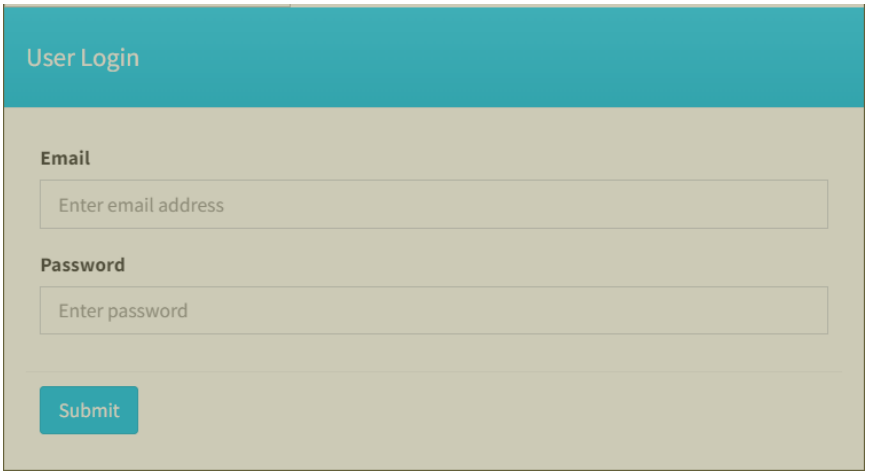

Figure 8: Login Page

```php
function send_otp_code( $to, $subject, $message ) {
    $mail = new PHPMailer(true);

    try {
        //Recipients
        $mail->setFrom('admin@thecityschool.com', 'Admin');
        $mail->addAddress( $to );

        //Content
        $mail->isHTML(true);                                  //Set email format to HTML
        $mail->Subject = $subject;
        $mail->Body    = $message;
        $mail->AltBody = $message;

        $mail->send();
        echo 'An OTP code has been sent to your email. Please check your email now.';
    } catch (Exception $e) {
        echo "Message could not be sent. Mailer Error: {$mail->ErrorInfo}";
    }
}
```

Figure 9: The partial code to send the OTP code in login function

Figure 9 shows the partial code to send One-Time Password (OTP) code in login function, while Figure 10 and Figure 11 show the dashboard of admin and teachers. For administrators and teachers, before they can access the system, they need to fill in the One-Time Password (OTP) that has been sent to their email.

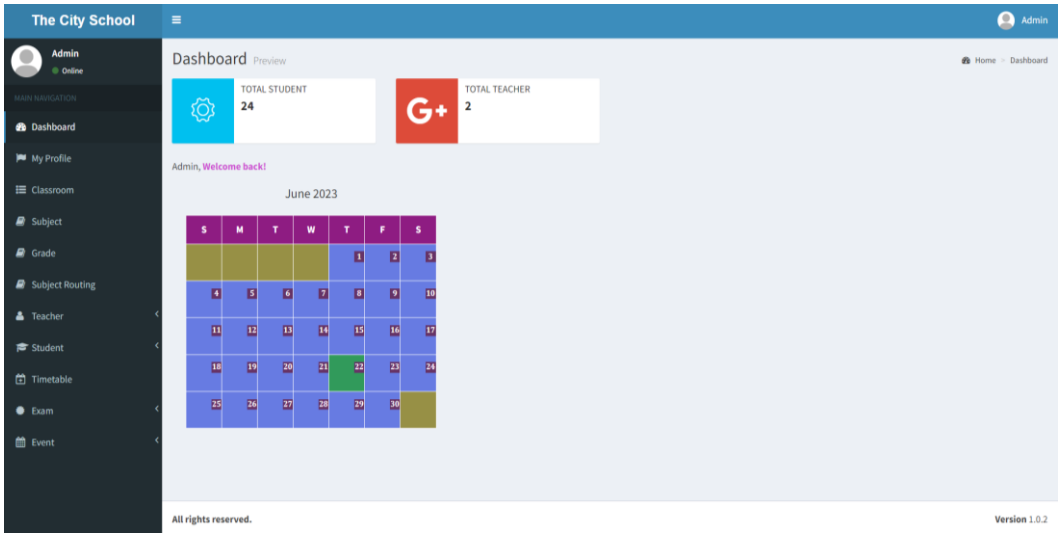

Figure 10: Dashboard of Admin

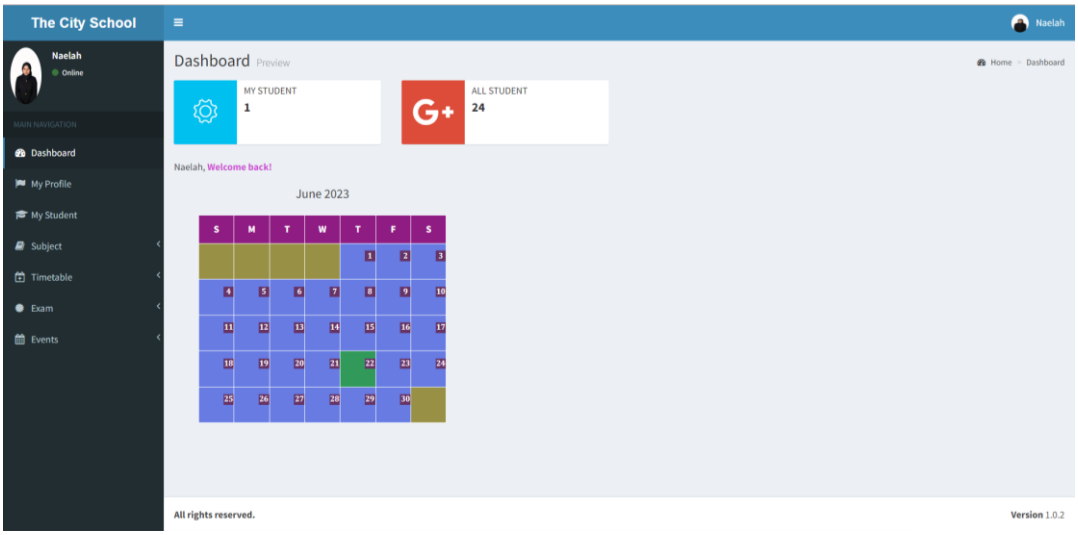

Figure 11: Dashboard of Teacher

Figure 12 shows the interface of One-Time Password (OTP) verification, while Figure 13 shows the email that has been sent for One-Time Password (OTP) code. When admin and teachers try to login to the system, they will receive an email of their One-Time Password (OTP) code, and they need to fill the verification page with the code that has been sent to their email.

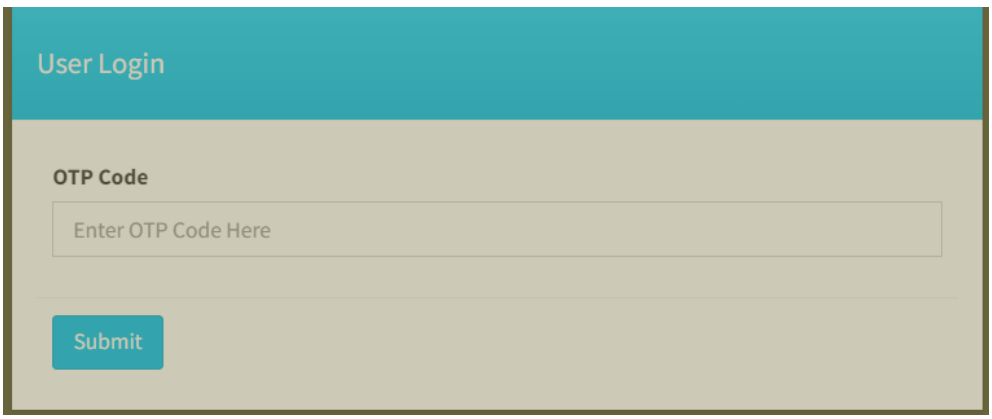

Figure 12: Interface of One-Time Password (OTP) verification.

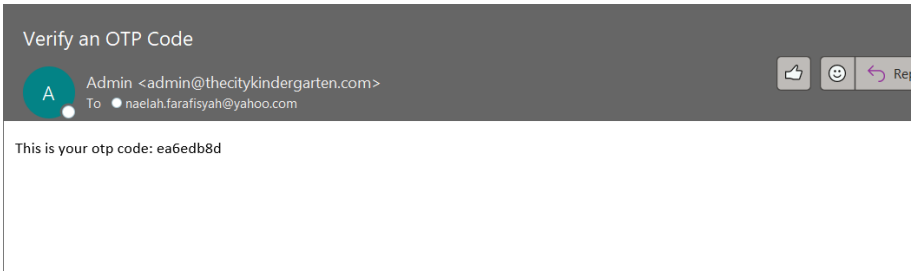

Figure 13: Email that has been sent for One-Time Password (OTP) code.

Figure 14 shows the partial code for email verification in login function, while Figure 15 shows the dashboard of parents. When parents try to access the system, they will receive a verification link through their email. They need to click the link to redirect to their dashboard.

```php
function send_verification_link( $to, $subject, $message ) {
    $mail = new PHPMailer(true);

    try {
        //Recipients
        $mail->setFrom('admin@thecityschool.com', 'Admin');
        $mail->addAddress( $to );

        //Content
        $mail->isHTML(true);                                  //Set email format to HTML
        $mail->Subject = $subject;
        $mail->Body    = $message;
        $mail->AltBody = $message;

        $mail->send();
        echo 'A verification link has been sent to your email. Please check your email now.';
    } catch (Exception $e) {
        echo "Message could not be sent. Mailer Error: {$mail->ErrorInfo}";
    }
}
```

Figure 14: Partial code for email verification in login function

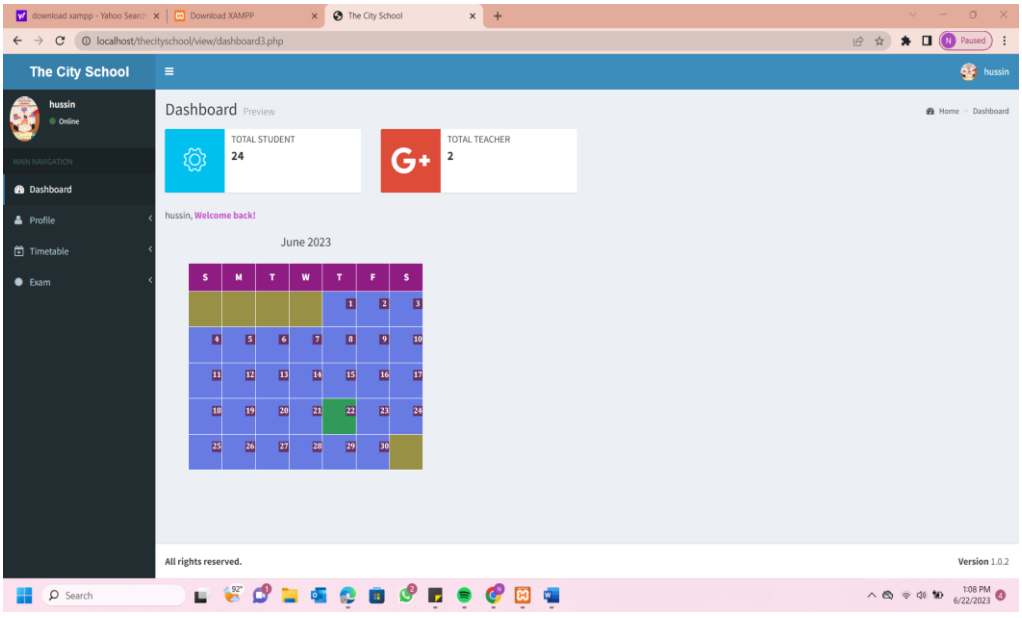

Figure 15: Dashboard of Parents

Figure 16 shows the notifications that prompt parents to check their email for verification link, while Figure 17 shows the email that has been sent for verification link. When parents try to login to the system, they will receive an email of verification link, and they need to click the link to redirect them to the system, which is dashboard page.

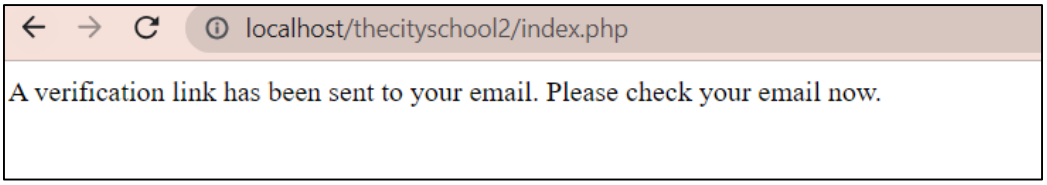

Figure 16: Notifications to prompt parents to check their email

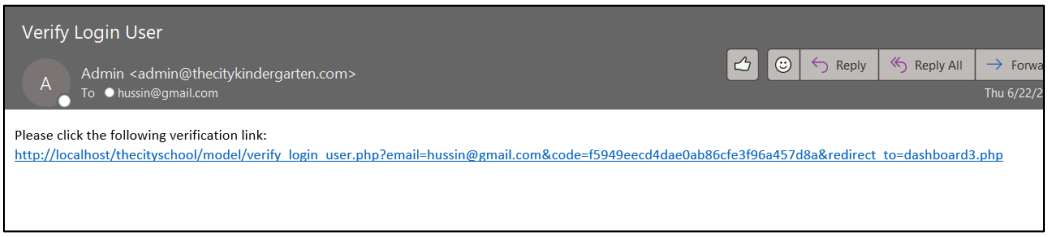

Figure 17: Email that has been sent for verification link

Figure 18 shows the user table with hashed password, while Figure 19 shows the partial code. After the user registers to the system, the password automatically hashes and inserted into the user table.

	id	email	password	code	type
Edit Copy Delete	1	naelah.farafisyah@yahoo.com	$2y$10$kUFfveDLPh16t8iqqKdSReSo.ccAJ8rSGomv8g.HMY3...	7ef29e89	Admin
Edit Copy Delete	2	naelahfarafisyah2@gmail.com	$2y$10$munvp8MeDWL8bJ0VP4Dx8ejJvNuylFY8LhREtbrZ5mG...	a025bd02	Teacher
Edit Copy Delete	49	parentz@yahoo.com	$2y$10$ZBNgt2i5P9GVwjks/lVeTOyxE8WeGqPNdeUGMX1qhJM...	6f4b4c7a36d15a14a7b5968e09f68806	Parents

Figure 18: user database with hashed password

```
$password = password_hash( $g_index_number, PASSWORD_DEFAULT );
```

Figure 19: Partial code to hash plain text password

Figure 20 shows the interface of add student, while Figure 21 shows the partial code of add student. For students, if the Identification number and email parents is already in the database, it will prompt an error message “the Index Number and email is duplicated”.

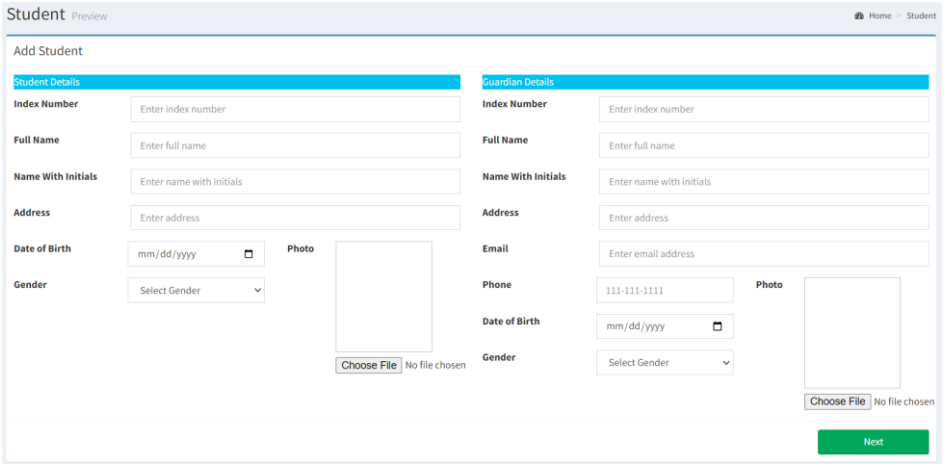

Figure 20: Interface of add student

```php
//Insert Student-------------------------------------------------

$sql1="SELECT * FROM student where index_number='$index_number'";
$result1=mysqli_query($conn,$sql1);
$row1=mysqli_fetch_assoc($result1);
$index_number1=$row1['index_number'];

if($index_number == $index_number1 ){
    //MSK-000143-1 The index number is duplicated.
    $msg+=1;

}else{
    //MSK-000143-4
    if(($extention == "jpg" || $extention == "jpeg" || $extention == "png") && $size < $max){//This line is not needed, bcz we checked it before.
        if(move_uploaded_file($tmpname, $image_path)){
            //MSK-000143-5

            $sql = "INSERT INTO student (index_number,full_name,i_name,gender,address,image_name,reg_year,reg_month,reg_date,b_date)
                VALUES ('".$index_number."','".$full_name."','".$i_name."','".$gender."','".$address."','".$image_path."','"
                .$reg_year."','".$reg_month."','".$reg_date."','".$b_date."')";
```

Figure 21: Partial code of add student

Figure 22 shows the interface to add teachers, while Figure 23 shows the partial code to add teacher. For teacher, if their identification number and email are already stored in the database, it will prompt an error message “the Index Number and Email are duplicated.” If the teachers only duplicate the email, thus it will show “the email is duplicated.”.

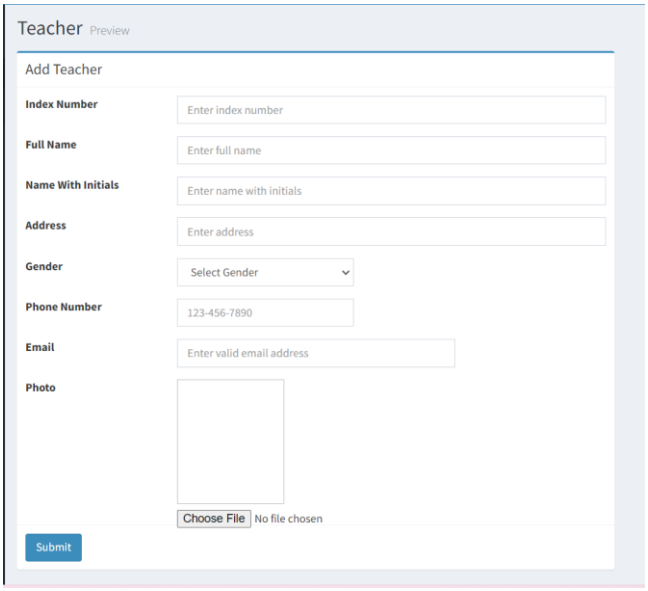

Figure 22: Interface of add teacher

```php
$sql2="SELECT * FROM teacher where email='$email'";
$result2=mysqli_query($conn,$sql2);
$row2=mysqli_fetch_assoc($result2);
$email2=$row2['email'];

$msg=0;//for alerts
$image_path =  $target_dir.$filename.".".$extention;

if($index_number == $index_number1 ){
    //MSK-000143-1 The index number is duplicated.
    $msg+=1;

    if($email == $email2){
        //MSK-000143-2 Both index number and email duplicate.
        $msg+=3;
    }

}else if($email == $email2){

    //MSK-000143-3 Only email address duplicates.
    $msg+=5;

}else{
    //MSK-000143-4
    if(($extention == "jpg" || $extention == "jpeg" || $extention == "png") && $size < $max){//This line is not needed, bcz we checked it before.
        if(move_uploaded_file($tmpname, $image_path)){
            //MSK-000143-5

            $sql = "INSERT INTO teacher (index_number,full_name,i_name,gender,address,phone,email,image_name,reg_date)
                VALUES ('".$index_number."','".$full_name."','".$i_name."','".$gender."','".$address."','".$phone."','".$email."','".$image_path.
                "','".$current_date."')";
```

Figure 23: Partial code of add teacher

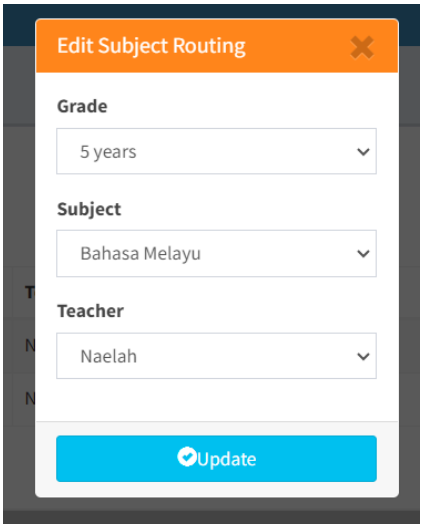

Figure 24: Update subject routing.

```
if($grade_id == $grade_id4 && $subject_id == $subject_id4 && $teacher_id == $teacher_id4){
    if($id == $id4){//MSK-000143-U-1

        if($fee == $fee4){
            $msg+=3;
            //MSK-000143-U-2 You didn't make any of changes.:D

        }else{//MSK-000143-U-3
            $sql5 = "update subject_routing set grade_id='".$grade_id."',subject_id='".$subject_id."',teacher_id='".$teacher_id."' where id='$id'";
            if(mysqli_query($conn,$sql5)){
                $msg+=1;
                //MSK-000143-U-4 The record has been successfully updated in the database

                $sql6="select grade.name as g_name,subject.name as s_name,teacher.i_name as t_name
                        from subject_routing
                        inner join grade
                        on subject_routing.grade_id=grade.id
                        inner join subject
                        on subject_routing.subject_id=subject.id
                        inner join teacher
                        on subject_routing.teacher_id=teacher.id
                        where subject_routing.id='$id'";

                $result6=mysqli_query($conn,$sql6);
                $row6=mysqli_fetch_assoc($result6);//MSK-000143-U-5
```

Figure 25: Partial code of update subject routing.

Figure 24 shows the interface of edit subject routing, while Figure 25 shows the partial code. Subject routing is when the administrator assigns the teachers to teach a subject in accordance with the class plan.

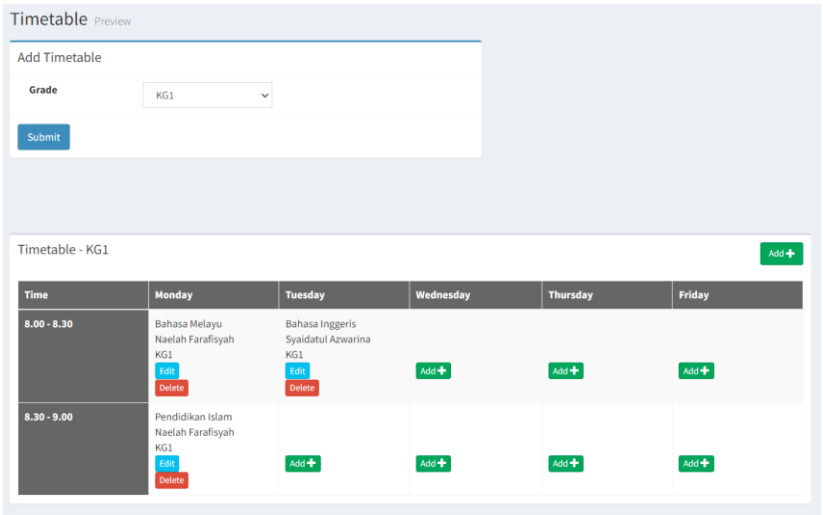

Figure 26: Interface of add timetable.

```
$sql1="SELECT * FROM timetable WHERE day='$day' and classroom_id=$classroom_id and end_time > $start_time and (start_time <= $start_time or start_time<$end_time)";

$result1=mysqli_query($conn,$sql1);

if(mysqli_num_rows($result1) > 0){//MSK-000143-1 At this time there already have class, in that classroom.
    $msg+=1;

}else{//MSK-000143-2

    $sql="INSERT INTO timetable (grade_id, day, subject_id,teacher_id,classroom_id,start_time,end_time)
  VALUES ( '".$grade_id."','".$day."','".$subject_id."','".$teacher_id."','".$classroom_id."','".$start_time."','".$end_time."')";

    if(mysqli_query($conn,$sql)){
        $msg+=2;
        //MSK-000143-3 The record has been successfully inserted into the database.
```

Figure 27: Partial code of add timetable.

Figure 26 shows the interface of add timetable, will Figure 27 shows the partial code. Based on the code, the admin cannot add more than one class at the same time. This is to avoid overlap between classes. If the admin adds a class at the same time, it will issue an error message stating that the time already have class. Figure 28 shows the interface of update events, while Figure 29 shows the partial code. Only teachers can add and update the events, while administrators and parents can view the events of the school.

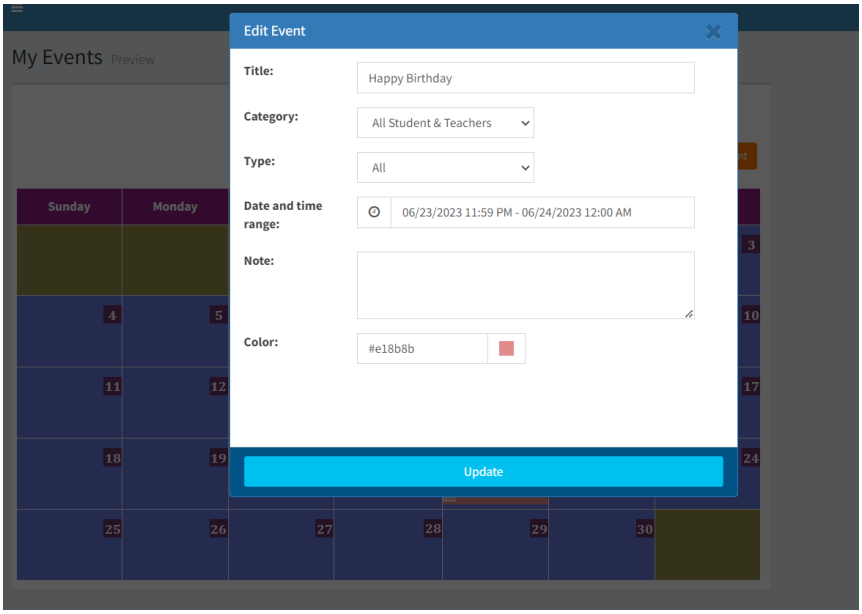

Figure 28: Interface of update events

```
$sql="SELECT * FROM events WHERE id='$id'";
$result=mysqli_query($conn,$sql);
$row=mysqli_fetch_assoc($result);

$s_date = $row['start_date_time'];
$e_date = $row['end_date_time'];
$date_time_range2=$s_date." - ".$e_date;

$prefix="";
$grade_id="";
$msg=0;

if(isset($_POST["checkbox1"])){
    for($i=0;$i<count($_POST["checkbox1"]);$i++){

        $grade=$_POST["checkbox1"][$i];

        $grade_id.=$prefix.$grade;
        $prefix=',';

    }
}

if($date_time_range == $date_time_range2){
    $sql1 = "update events set title='".$title."',note='".$note."',color='".$color."',category_id='".$category_id."',grade_id='".$grade_id."' where id='$id'";

    if(mysqli_query($conn,$sql1)){

        $msg+=1;
        //MSK-000143-U-3 The record has been successfully updated in the database
```

Figure 29: Partial code of update events.

5.2 Testing

Functional testing is done to ensure that the system is operating successfully and satisfying the functional requirements. It is carried out to guarantee that the system fulfils its intended purpose and produces the desired result. Table 4, 5 and 6 show the test plan for Admin, Teacher and Parent respectively.

Table 4: Test Plan for Admin

No.	Test case	Pass	Fail
1.	System can be executed from start to end.	Yes	
2.	Admin able to login and logout to the system.	Yes	
3.	Admin able to register teachers, students, and parents.	Yes	
4.	Admin able to enter the system with OTP code	Yes	
5.	Admin able to delete teachers, students, and parents	Yes	
6.	Admin able to add classroom, grade, and examination name	Yes	
7.	Admin able to generate timetable for students and teachers	Yes	

Table 5: Test Plan for Teacher

No.	Test case	Pass	Fail
1.	System can be executed from start to end.	Yes	
2.	Teacher able to login and logout to the system.	Yes	
3.	Teacher able to add events and activities	Yes	
4.	Admin able to enter the system with OTP code	Yes	
5.	Teacher able to add grade for students	Yes	
6.	Teacher able to view list of class and students	Yes	
7.	Teacher able to view timetable	Yes	

Table 6: Test Plan for Parents

No.	Test case	Pass	Fail
1.	System can be executed from start to end.	Yes	
2.	Parents able to login and logout to the system.	Yes	
3.	Parents able to view events & activities	Yes	
4.	Parents able to view their child’s timetable	Yes	
5.	Parents able to view their child’s examination result	Yes	

5.3 User Acceptance Test

The user acceptance test result is collected using Google Form. The google form consists of two sections, which is A and B. Section A is about the interface of the system, while section B is about the functionality of the system. These sections have scales from 1 (Very Unsatisfied / Very Disagree) to 5 (Strongly Unsatisfied / Strongly Agree). This user testing was created to make sure the system was developed according to the user's preference and requirements. Other than that, user testing also can identify the feedback from the user. This allows the user to assist the developer in identifying current issues and providing a summary of the system's availability. There are 10 users who were tested this system, and the roles are admin, teachers, and parents.

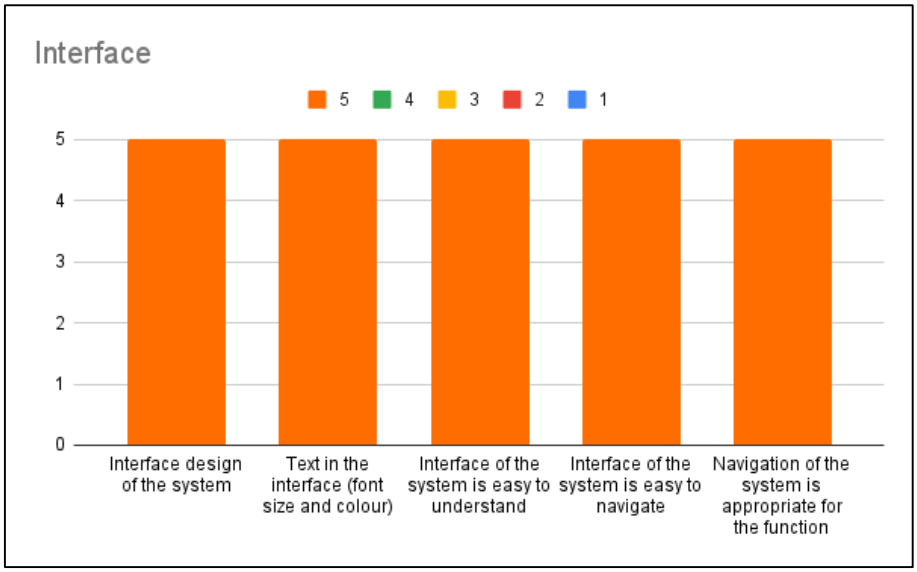

Figure 30: Google form user acceptance result for Section A: Interface

Figure 30 shows the google form user acceptance result for Section A which is for interface of system. The question for this section is interface design of the system, text in the interface, interface of the system is easy to understand, system is easy to navigate, and navigation of the system is appropriate for the function. The respondent rate 5 to all questions, which indicates that users are strongly satisfied with the interface design.

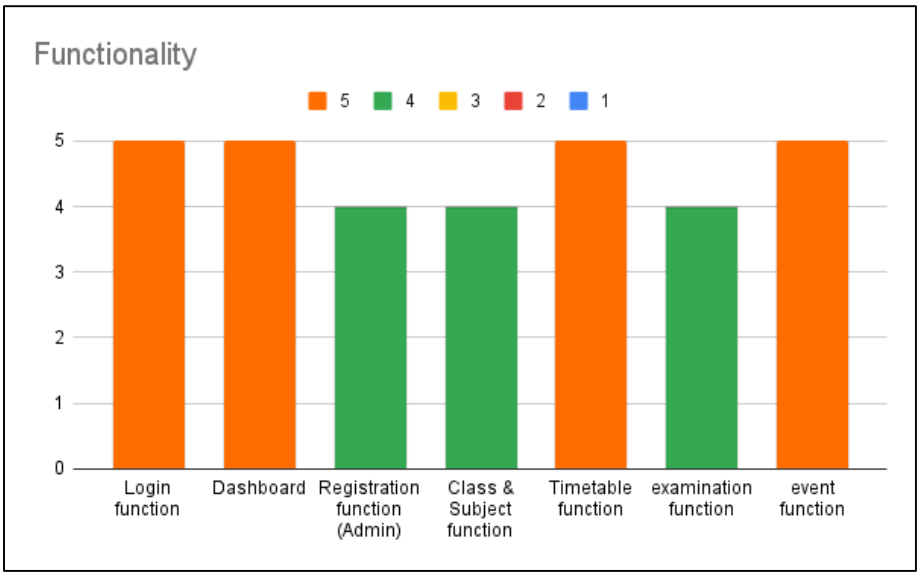

Figure 31: Google form user acceptance result for Section B: Functionality

Figure 31 shows the google form user acceptance result for section B, which is for functionality of the system. The questions for this section are login function, dashboard, registration function (for administrator only), class and subject function, timetable function, examination function, and event function. For this section, the users' rates 4 for registration function, class and subject function, and examination function, while registration function, class and subject function, and event function are rated 5 which indicates user are strongly satisfied with the functionality.

Respondents also leave a crucial comment for the feature that the developer may overlook. Firstly, registration students, the system requires to register each student for each parent, it means this system

cannot register more than one child. In the end, the registration of students could be better if the administrators could register more than one child.

6. Conclusion

In conclusion, this online kindergarten management system with dual authentication will assist administrators, teachers, and parents in smoothly managing the school and storing data. It is because this system provides registration, examination, timetable, and events, making it easier for management to maintain them all in one location. The kindergarten school will benefit greatly from this because they will no longer need to manually save the data.

This system has been built effectively in order to achieve the objectives. The goal of this system is to ensure that this kindergarten stores and secures critical data with dual authentication, such as OTP code and verification link. This is because only verified users can enter the system with the code and link that has been provided through their emails. Parents may also see their son's schedule and exam results from anywhere.

Even if this system meets the objectives, it will have certain limitations that may be utilized as future references to improve the system. This system can currently only add one student for each parent. For example, if the parents have many children, the administrator can only add one student with a parent. Aside from that, teachers are unable to upload exam papers and leave remarks about the development of their pupils.

Develop new features which enable administrators to register their children under one parent as suggestions for improving the system to allow adding more than one kid. Furthermore, teachers can submit test papers and include comments on student development and successes so that parents are informed of their children's progress.

Acknowledgement

The authors would also like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support.

References

[1] I. Blau and O. Presser, “e-L eadership of school principals: Increasing school effectiveness by a school data management system,” *British Journal of Educational Technology*, vol. 44, no. 6, pp. 1000–1011, Jul. 2013, doi: https://doi.org/10.1111/bjet.12088.

[2] I. Blau and M. Hameiri, “Teacher–families online interactions and gender differences in parental involvement through school data system: Do mothers want to know more than fathers about their children?,” *Computers & Education*, vol. 59, no. 2, pp. 701–709, Sep. 2012, doi: https://doi.org/10.1016/j.compedu.2012.03.012.

[3] L. Jean Camp and S. Lewis, *Economics of Information Security*. Springer Science & Business Media, 2006. Accessed: Sep. 27, 2023. [Online]. Available: https://www.usenix.org/conference/soups2019/presentation/reese

[4] H. Parmar, N. Nainan, and S. Thaseen, “Generation Of Secure One -Time Password Based On Image Authentication,” *Semantic Scholar*, 2012. https://www.semanticscholar.org/paper/GENERATION-OF-SECURE-ONE-TIME-PASSWORD-BASED-ON-Parmar-Nainan/1e752498160daa941a76028ee7b0e5aeae446992

[5] R. Sandhu, “Role-based Access Control,” *Elsevier eBooks*, pp. 237–286, Jan. 1998, doi: https://doi.org/10.1016/s0065-2458(08)60206-5.

[6] "SAPS," *Moe.gov.my*, 2022. https://sapsnkra.moe.gov.my/ (accessed Jan. 10, 2023).

[7] lianasahperi, "MODUL SMPK," *Slideshare.net*, 2018. https://www.slideshare.net/lianasahperi/modul-smpk?next_slideshow=true (accessed Nov. 21, 2022).

[8] "Sistem Maklumat Prasekolah Kebangsaan," *Moe.gov.my*, 2022. https://smpk.moe.gov.my/ (accessed Oct. 27, 2022).

[9] "Fig. 1: Prototyping Methodology," *ResearchGate*, 2022. https://www.researchgate.net/figure/Prototyping-Methodology_fig1_303016579 (accessed Jan. 08, 2023).

[10] K. Rana, "Prototype Model - Phases, Types, Advantages & Disadvantages," *ArtOfTesting*, Feb. 07, 2020. https://artoftesting.com/prototype-model (accessed Jan. 01, 2023).

[11] "Entity Relationship Diagram (ERD) - What is an ER Diagram?," *Smartdraw.com*, 2023. https://www.smartdraw.com/entity-relationship-diagram/ (accessed Jan. 03, 2023).