Applied Information Technology And Computer Science Vol. 5 No. 1 (2024) 091-109

AITCS

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

TIKAS: Kindergarten Attendance Application with Role-Based Access Control for Tadika Inovasi Kreatif

Nur Hazwani Hanum Samsudin[1], Nurul Hidayah Ab Rahman[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.2024.05.01.006
Received 19 May 2024; Accepted 23 May 2024; Available online 30 August 2024

Abstract: This attendance application (TIKAS) is developed for Tadika Inovasi Kreatif to record the attendance of the students and generate an annual report of attendance. The classified information breach and altered by the unauthorized access motivates the direction of developing TIKAS with automated attendance management. If the student is absent, the parent can easily key in the reason for the absence in the attendance form. Therefore, this study proposed TIKAS to only allow authorized users to enter the application by applying role-based access control (RBAC). Multifactor authentication using One Time Password is also applied to provide secure authentication process to TIKAS. The methodology employed in this application is agile model which consists of six phases: planning, analysis, design, implementation, testing, and deployment. This application was developed using Android Studio as the Integrated Development Environment (IDE), Java programming language, and Firebase as a database. As a result, TIKAS is developed with six modules that are: dashboard, login, register, manage account, parent management, teacher management, and attendance management. The access control to the modules is defined using three role-based that are system admin, teacher, and guardian/parents. Thus, the significance of the study is to prevent a violation of access control to the application and ease the process of recording attendance.

Keyword: Attendance Application, Authentication, Role-Based Access Control

1. Introduction

In the advanced technology era, an organized and efficient attendance application is needed to calculate the average percentage of students present throughout the school year. In spite of this, manually taking attendance has a few disadvantages. For example, a large number of students in a classroom could consume a lot of time as well as affecting the teaching time. Furthermore, it is possible for the manual calculation to be incorrect if it is not done carefully. Therefore, a digital attendance application could facilitate a reduction in time-consuming as the teachers do not have to call out the

*Corresponding author: hidayahar@uthm.edu.my

students' names one by one and they only have to click a few buttons to record attendance. On the other hand, TIKAS can generate a more accurate record of attendance as the report of the total students' present will be counted automatically.

Although an online attendance system improves the process to track students' attendance, some security issues should be considered. For instance, data can only be accessed by individuals whose identities have been verified. According to the Open Web Application Security Project (OWASP), one of the web application security risks is broken access control [1]. Lack of a role for users may violate the principle of least privilege since every user can access all the data. Hence, each user must have their own intended permissions to prevent broken access control such as only the admin can view, update, and delete while the teachers can only view and update. The other weakness of access control is the privileges are elevated when using an account as a user, but acting like an admin once a user logged in. Therefore, role-based access control is important to protect the sensitive data of a system.

Furthermore, Multi-Factor Authentication (MFA) is important to grant access or login to the application after providing two or more information of verification [2]. Significantly, users must go through a few security layers before entering or being authorized by the system. In this proposed application, the admin and teacher will receive a one-time password (OTP) when registering and editing information while the parent or guardian must enter OTP to authenticate themselves. The feature is significant to make the security even stronger and able to protect sensitive data.

Therefore, this study aims to propose an application for kindergarten students' attendance with role-based access control for Tadika Inovasi Kreatif. This kindergarten attendance application will be developed to help the kindergarten students' system from manual to automatic. The teachers will key in the student's attendance automatically and the parents or guardians will get the report of the attendance. The parents or guardians will get a message if their child is absent, and they can submit the reason for absence through the application. Furthermore, the application will generate a report to see a pattern of attendance such as the reason for student absence like fever season or festival season. The objectives of the projects are:

- To design a kindergarten attendance application with role-based access control.
- To develop a kindergarten attendance application with role-based access control.
- To test the functional and security effectiveness and conduct user testing of the kindergarten attendance application with role-based access control.

1.1 Problem Statement

In this study, the problem statements can be broadly divided into four issues that are the likelihood of a manual attendance system, the improvement of multi-factor authentication, classified information breach and alteration by unauthorized access, and the arising of kidnapping cases.

More paperwork is required in the manual attendance system, and it can be inconvenient as it is hard to track back when needed. It also takes more space to store. Traditionally, the teachers need to record the student's attendance by calling out their names and ticking on the paper for each of them. The scenario may be a waste of time as some students are not alert and the teachers must call out their names a few times. Moreover, the students are at the age of being mischievous. Furthermore, a manual attendance system can increase the risk of human error and worsen the employee's productivity as more work is needed. Next, the limitation in generating reports makes it less likely to analyze the attendance trends.

The multi-factor authentication (MFA) allows the user to obtain access to a resource by providing two or more verification factors. Three main types of MFA which are:

- things the user knows, such as a password;

- things the user have, such as One-time Password (OTP)s sent via Short Message Service (SMS); and
- things the user is, such as voice, makes the authentication process stronger.

Users must enter more than just a username and password to identify themselves, thus enhancing the security of the system. According to Forbes, Microsoft has stopped using SMS verification messages in 2020 [3]. The multi-factor authentication via SMS is possible to breach especially when the SMS is not encrypted from the sender to the receiver. Most likely around 50 out of 10,000 Microsoft accounts were hacked back in February 2020 [3]. Hence, this proposed application developed will enhance the security of the application to prevent these threats.

In late 2016, Yahoo announced the largest data breach of all time, as hackers stole information associated with at least 1 billion accounts in 2013 [4]. Recently, a Department for Education, Children and Young People of Tasmania financial data including name, addresses, invoices, and bank account numbers had been compromised in March 2023. The attacker gained access through a third-party file transfer service used by the government [5]. The compromised data can be used to violate the law or to threaten the user. It indicates that data from the educational sector also has become one of the hackers' targets. These problems should be taken seriously as they contain private data for everyone. Secure data protection is needed to ensure that confidential information is not altered or leaked to unauthorized users. This is to safeguard the integrity and confidentiality of the data. In this way, the user will be confident to use the application that has been developed.

The arising of kidnapping cases became a big concern nowadays, even in a public space, anything can happen. According to the news written by Melissa Teo, a man claims that his friend's son was involved in a kidnap attempt by two women at Resorts World Genting [6]. Resorts World Genting is a well-known theme park in Genting Highland and an open space. From 2020 to February 2022, 1,509 children were reported missing and 85 of them are still not located, says Bukit Aman's criminal investigation department (CID). Most kidnapping cases involve girls in general [7]. Thus, the risk can be minimized by implementing the automated attendance application where the parents will be aware of whether their child has been picked up or not based on the report.

2. Related Work

Approaches applied in the proposed application are role-based access control and multi-factor authentication.

2.1 Role-Based Access Control (RBAC)

Role-based access control (RBAC) principle requires access rights to be assigned to roles rather than to specific users [8]. RBAC has several entities which are subjects, objects, roles, permissions, and actions. A group of permission is assigned to subjects by role. Hence, each subject can only access objects or perform an action based on their role. The role was assigned after an analysis made by an organization. As an example, a school system consists of teachers, students, parents, and so forth. Only an administrator can control the system and assign roles to other users.

RBAC is also used to safeguard the data so that only authorized users can enter. Once the user is authorized, the data can be accessed based on their roles. Hence, RBAC is used in the TIKAS to strengthen the security of the application.

2.2 Multi-Factor Authentication (MFA)

Multi-factor authentication (MFA) is one of the security mechanisms. It is one of the secure ways to log in to an account. Traditionally, using a username and password to log in, but not anymore. MFA or referred to as two-factor authentication (2FA) was implemented for a better user experience. It is to make sure that the right person is using the system or application.

The user needs to provide two or more verification factors to gain access to the information or data. The verification includes something you know (username and password), something you have (OTP, or access card), and something you are (fingerprint, or iris recognition). The verification must come from two different types. For example, username and password are something you know, combined with OTP which is something you have. It would not be 2FA or MFA if you log in using a password and PIN number because both are something you know.

MFA has its own benefit because it enhances an organization's security by identifying themselves with more than just a username and password. It makes users feel more confident and safer to use the system as it has more than one protection layer to secure it. Having MFA could prevent attacks like brute force attacks or password guessing attacks. Also, a MFA system mitigates the risks associated with single-factor logins, including password breaches and unauthorized access to trusted devices [9]. Hence, it will be built into the proposed application to secure the data.

2.3 Review on Existing System

The first existing system is the Attendance System Using Face Recognition. This system was developed to automate the attendance system as using the manual method of passing the attendance paper around and signing it, the students might cheat when using the manual method. Moreover, the system used face recognition to take attendance. The algorithm of face recognition that was used in this system is the feature extraction module. In this system, a picture will be taken by a mobile phone with different conditions such as wearing a spectacle and not wearing a spectacle, a different expression like smiling and not smiling, and different angles like from the front side, left side, and right side. Later, it will be saved in the database, and it will go through pre-processing. The picture will be resized, convert image to grayscale, and median filtering. Afterward, it will go through a feature extraction process. Then, the answer will be given whether the face is recognized or not by the system [10].

The next existing system is the Students Attendance System for Parents. This system was developed for parents to trace whether their children are present or not in school. The parents and teachers could see the student's attendance report via a pie chart or pdf that was recorded. It will also benefit the teachers as they can contact the parents faster. It is because the details of the students were registered in the system. So, the teachers can see the parents' contact in the students' detail. The system used role-based access control (RBAC) as the main security feature. The parents and teachers will be registered by the admin. Later, the admin, parents, and teachers will log in based on their role by entering their email addresses and password. Teacher, parent, and admin could view student's details and the attendance report, but only the admin or teacher could delete students. The information updated was saved in the database. The one who keyed in the student's attendance was the teacher for the subject, and the teacher can only add the list of the students that had been added to their class or subject. Furthermore, this system requires a password policy that must include at least one number, one alphabet either uppercase or lowercase, one special character such as underscore, and a minimum of 10 characters long. To ensure the security of the password, the password was hashed in the database [11].

The last existing system is the Facial Recognition Attendance System using CNN by Tan [12]. The target of this system is to create an attendance system using face recognition for employees of a company or a business. Facial recognition was chosen to decrease the chance of an employee clocking in for another employee. The system also aims to reduce the risk of recording the wrong time when clocking in and clocking out. Due to Covid-19, facial recognition is the best option to avoid physical contact. So, this system was implemented, and the user only needs to scan his or her face with a webcam to record their attendance, and time will be automatically recorded and stored in the database. A company's employer must log in and be authorized by the system to register a new user or employee. Once registered, the new employee's face will be recognized by the system and the employee can easily clock in and clock out. Also, a report on the employees' attendance will be generated daily.

2.4 Comparison Table

A study of existing systems and the proposed application is important to differentiate the output and to analyze it. In this section, Table 1 shows the comparison between TIKAS and existing systems.

Table 1: System comparison

Security Requirements	Attendance System Using Face Recognition	Students Attendance System for Parents	Facial Recognition Attendance System Using CNN	Proposed Application
Role-based access control (RBAC)	No	Yes	No	Yes
One-Time Password (OTP)	No	No	No	Yes
Register/ Login	No	Yes	Yes	Yes
Database	Yes	Yes	Yes	Yes
User	Students	Teachers, parents	User, Admin	Teachers, parents
Project Type	Tool	Web-based	Web-based	Application
Platform	Web-based	Web-based	Web-based	Mobile based

From Table 1, all the systems and applications have a database to store the data. Next, the proposed application has OTP as a security requirement to authenticate before entering the application. Only an Attendance System Using Face Recognition is not required to register or login. Besides, both the Students Attendance System for Parents and the proposed application have an RBAC as a security measure. Moreover, the users of the Attendance System using Face Recognition are students while users of the Attendance System using Face Recognition and the proposed application are teachers and parents, and the Facial Recognition Attendance System Using CNN users are the company employee and employer. Other than that, the project type for the proposed application and the Attendance System using Face Recognition differ while the Students Attendance System for Parents, and Facial Recognition Attendance System Using CNN is the same which is web-based. The project type of Attendance System using Face Recognition is a tool while the proposed application is an application. Lastly, the platform used for the proposed application is mobile-based while others are web-based.

3. Methodology

The software development model that was chosen for the proposed application is agile. Agile is an iterative model. Agile methodology was chosen as the process or phase keeps rotating, therefore the application can be improved from time to time, and a client could give feedback. The feedback from the client is important to determine if functional and user requirements are fulfilled. It is also significant to gain the client's trust that the application is secure.

3.1 Planning Phase

In the planning phase, an objective, problem statement, expected outcome, and project significance is discussed to find the exact solution by proposing the proposal. The stakeholder involved in the proposed application is Tadika Inovasi Kreatif. The idea of implementing the proposed application came as the kindergarten has no automatic attendance system. Hence, there is no security applied. Because of that, the security features of the proposed application are described in the planning phase where RBAC and MFA were chosen.

3.2 Analysis Phase

The analysis phase explains the requirements needed in the proposed application. So, the following is the functional, and non-functional requirement of TIKAS.

Functional requirements explain the functionalities of the proposed application, in which the user can enter input and expect an output after the operation is performed. Table 2 shows the functional requirements.

Table 2: Functional requirements of the proposed application

No.	Functional Requirements
1.	The application should provide a registration form for the users.
2.	The application should provide a login form for the users.
3.	The application should provide OTP verifications for the users.
4.	The application should grant the teachers to record attendance.
5.	The application should allow the teachers to view the dashboard.
6.	The application should grant the parents or guardians to update their child's reason for absence.
7.	The application should allow the parents or guardians to view the report of their child's attendance.
8.	The application should allow the admin to insert, update, and delete parents' account.
9.	The application should allow an admin to insert, update, and delete the teachers and students.
10.	The application should allow all target users to manage an account of their own.

Non-functional requirement refers to the performance of the application. Table 3 shows the non-functional requirements.

Table 3: Non-functional requirements of the proposed application

No.	Non-Functional Requirements
1.	The application should be able to generate the attendance report accurately.
2.	The application should be able to identify the role that logs into the application.
3.	The application should be able to store the password after being encrypted in the database.
4.	The application should be able to protect the users' information.
5.	The application should be able to record the student's attendance.
6.	The application should be able to record the reason for students' absences.
7.	The application should include password complexity.
8.	The application should be able to authorize the user by role.
9.	The application should be able to send an OTP to the user.

3.3 Design Phase

The design phase is where the application design is determined. This is to finalize the user's needs after the analysis phase. The user's need is interpreted in the application and database design. The following figure shows a class diagram of TIKAS.

The class diagram is used to show the application structure such as attributes, methods, and relationships between attributes. The class diagram in Figure 1 shows the relationship between each module. For example, all target users must input all fields required before entering the application, including email, and password. All target users are also required to register if a user does not have an account yet. Other than that, all target users can manage their own account to update information such as by changing their phone number. The teacher has another extra module which is attendance management where the teacher could update the student attendance. In addition, the admin can manage the parent and the teacher like to delete any teacher that no longer works in the kindergarten.

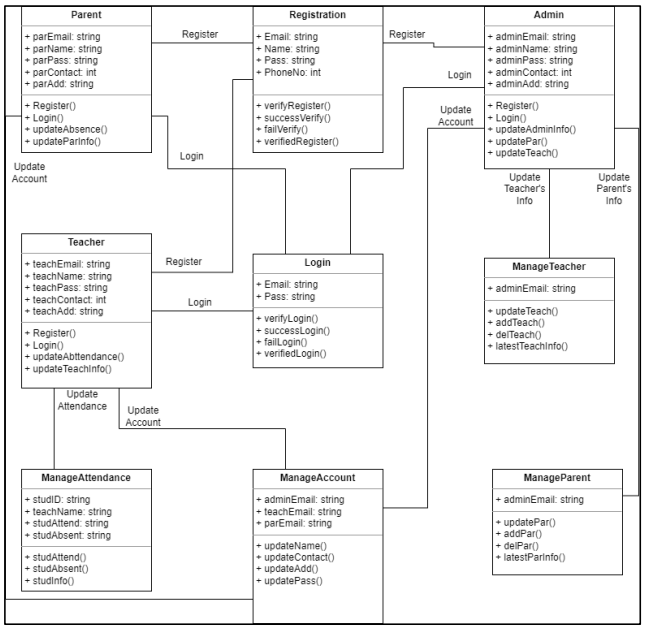


Figure 1: Class diagram of proposed application

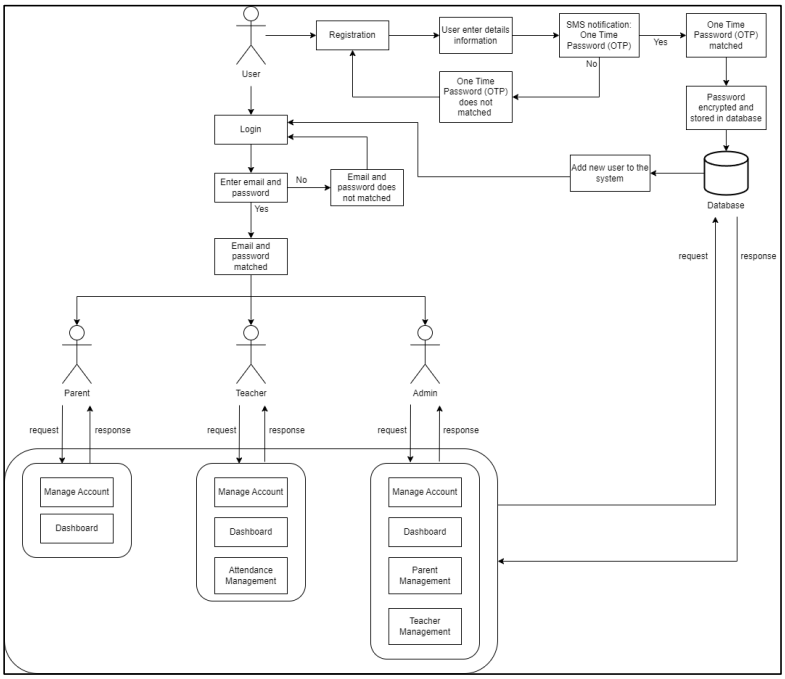


Figure 2: System architecture of TIKAS

System architecture describes how the module relates to one another. Figure 2 shows the architecture of the proposed application. The users are expected to register their account by entering their details like email, phone number, and desired password. Once entered, the user should receive an SMS notification that contains OTP to verify their identity. The OTP that the user receives must match the one that the new user enters. Else, the user must register again. Once the password has been encrypted and the data will be stored in the database, a new user is created. The user can log into the application with an email and password.

Next, the users are expected to log into the application based on their role. The user should enter their email and password. If the email and password are matched with the data being stored in the database, then the user logs in successfully. Else, the login process is unsuccessful, and the user must log in again. After logging in, users can do a process that is related to their role.

The admin role has the most access privileges which are dashboard, manage account, parent management, and teacher management. The teacher can access their dashboard with three main menus. The menus are manage account, student management, and teacher management. If the admin clicks on the manage account, the admin can view or update the account information. If the admin clicks on the parent management, the admin can view, update, insert, or delete the parent's info. Else, the user must log in again with the correct credential data. This is to help the one who is not really into information technology (IT) like the older generation.

Moreover, the teacher has only 3 accesses that the teacher can do which are manage account, dashboard, and attendance management. The teacher can manage accounts by viewing or updating their information under the dashboard menu. The teacher can record the student's attendance in the attendance menu whether the student is present or absent.

Lastly, the parent has a dashboard and manages an account. There are two menus under the dashboard which manage account and attendance. In the Manage Account, the parent can view, or update their account while in the attendance menu, the parent is able to view the report of their child's attendance or update the reason for their child's absence. The next design is Unified Modelling Language (UML) diagrams consisting of use case diagrams and sequence diagrams.

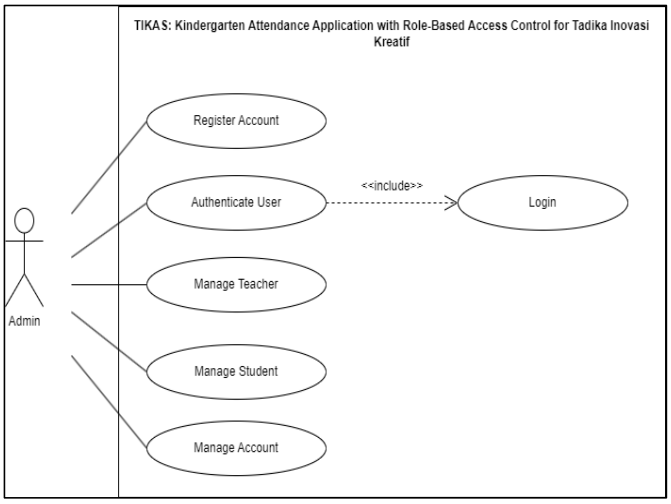


Figure 3: Use case diagram for admin

The use case diagram for admin is shown in Figure 3. The admin can access five modules in total. First, the admin must register as an admin. Second, the admin can log in to the application once successfully registered. Next, the admin can manage the teacher and the student by inserting, updating, or deleting. Also, the admin can manage their own account.

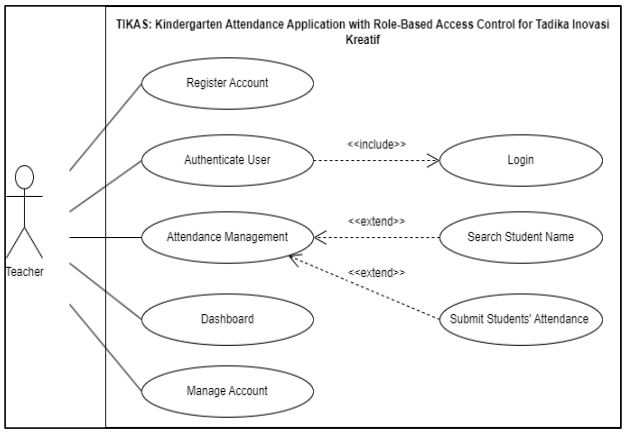


Figure 4: Use case diagram for teacher

The teacher's use case diagram is explained in Figure 4. The teachers are allowed access to five modules in the application. The same goes for the admin, the teacher needs to register as a teacher. Next, the teacher can log in to the application and access the dashboard to manage the students' attendance. Teachers can manage student attendance by searching their names and submitting student attendance. Lastly, teachers can also manage their own accounts.

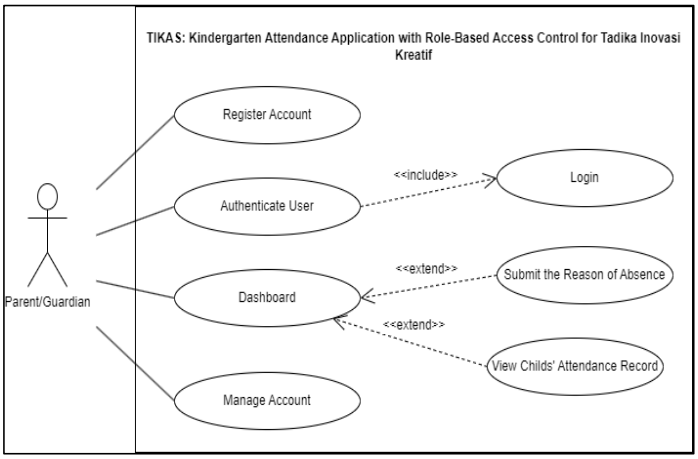


Figure 5: Use case diagram for parent

Figure 5 shows the use case diagram for parents or guardians. The parents or guardians can access four modules, including registering an account, logging in to the application, dashboard, and managing it. All these modules are for parents or guardian aid to submit the reason for the absence of their child to the application, or to view their child's attendance.

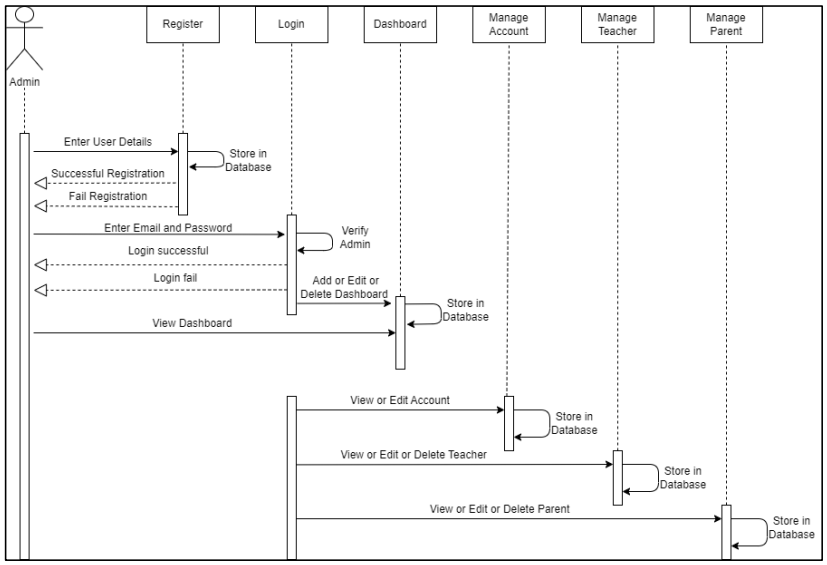


Figure 6: Sequence diagram for admin

The sequence diagram of the proposed application will be shown in Figure 6 for an admin. Firstly, the admin must register as an admin, and log in by entering the email and password. Once login, the admin can either manage the account, the teacher, the parent, or the dashboard.

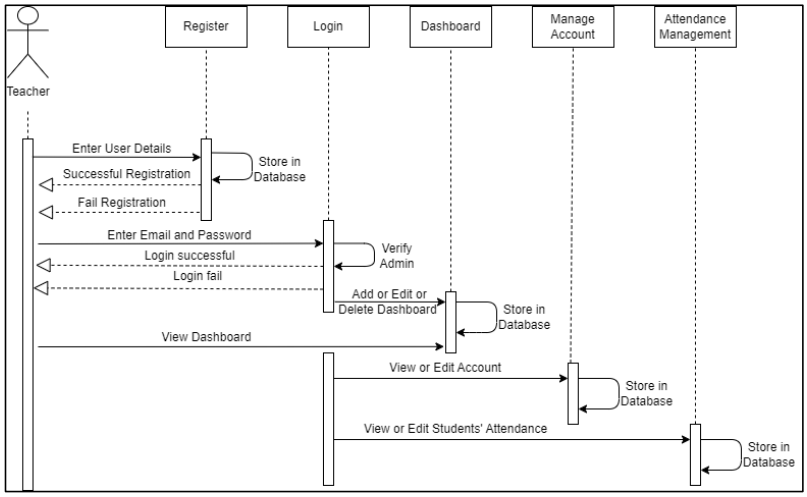


Figure 7: Sequence diagram for teacher

Figure 7 shows the sequence diagram for a teacher. The teacher is required to register as a teacher before logging in to the application. After login in successfully, the teacher can freely do an activity such as going through the dashboard to check information, manage accounts, or manage student attendance. All the data that has been edited by the teacher will be stored in the database.

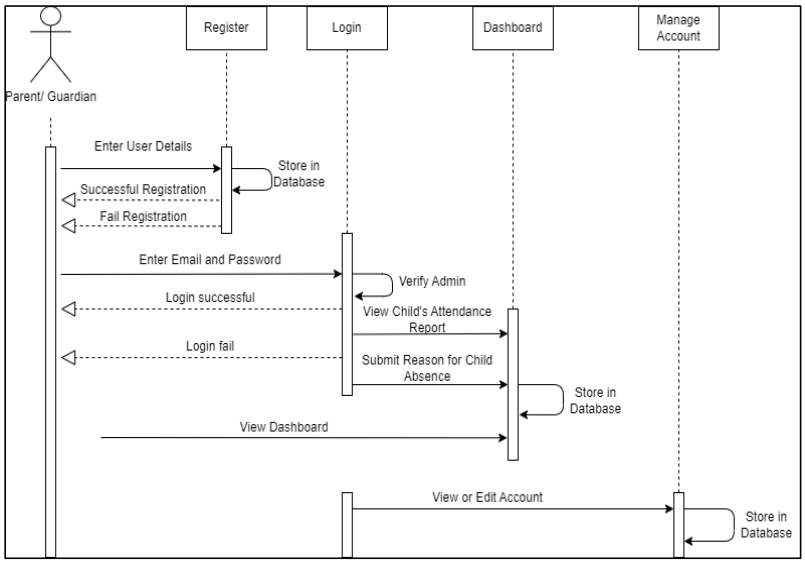


Figure 8: Sequence diagram for parent

The parent or guardian sequence diagram is shown in Figure 8. The first step is that the parent or guardian needs to register as a parent or guardian. Next, parents or guardians are required to enter the application by entering an email and password that matches with data stored in the database. Afterward, the parent or guardian can either manage an account or view, submit a reason for the child's absence, or view the child's attendance report.

The access control matrix describes the access permissions of subjects and objects to the application. Based on the access control matrix table, only activities or operations authorized ensure to be executed [13]. Table 4 shows the access control matrix table of the application.

Table 4: Access control matrix table

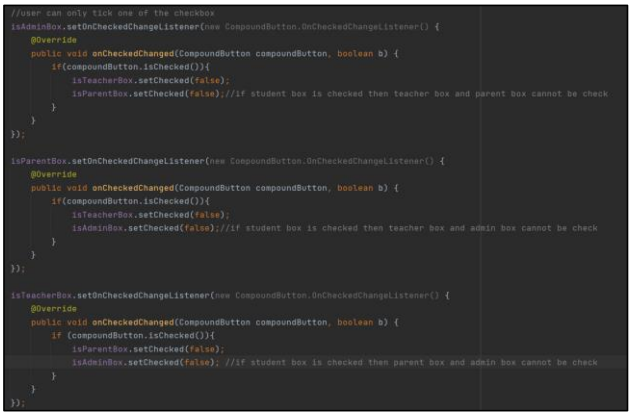
User	Admin	Teacher	Parent
Register	rwx	rwx	rwx
Login	rwx	rwx	rwx
Dashboard	rx	rx	rx
Manage account	rwx	rwx	rwx
Teacher management	rw	--	--
Parent management	rw	--	--
Attendance management	r	rwx	rwx

In Table 4, the row represents the modules, and the column stands for users. *r* means permission to read, *w* for permission to write, and *x* stands for permission to execute. Read means the user can access the module. To modify the modules, users need permission to write. Execution can be done if the user has permission to execute the modules. For instance, all users can read, write, and execute manage account modules as they can view the module, modify the details, and update the modified user details.

4. Results and Discussion

4.1 Implementation of Security Features

In this section, the implementation of security features as a contribution to this application is discussed. RBAC is the main contribution of TIKAS' security feature. In addition to it, TIKAS is developed with MFA, and input validation to minimize potential security risks.



```
//user can only tick one of the checkbox
isAdminBox.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {
    @Override
    public void onCheckedChanged(CompoundButton compoundButton, boolean b) {
        if(compoundButton.isChecked()){
            isTeacherBox.setChecked(false);
            isParentBox.setChecked(false);//if student box is checked then teacher box and parent box cannot be check
        }
    }
});

isParentBox.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {
    @Override
    public void onCheckedChanged(CompoundButton compoundButton, boolean b) {
        if(compoundButton.isChecked()){
            isTeacherBox.setChecked(false);
            isAdminBox.setChecked(false);//if student box is checked then teacher box and admin box cannot be check
        }
    }
});

isTeacherBox.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {
    @Override
    public void onCheckedChanged(CompoundButton compoundButton, boolean b) {
        if (compoundButton.isChecked()){
            isParentBox.setChecked(false);
            isAdminBox.setChecked(false); //if student box is checked then parent box and admin box cannot be check
        }
    }
});
```

Figure 9(a): Code for user to choose a role

Figure 9(b): Registration interface

Figure 9(a) depicts the code to determine the role of the user in the application. Users can only tick one of the checkboxes. Figure 9(b) shows the interface of the registration for users. Users must choose one out of three roles as one requirement to register.

```
//checkbox validation
if(isTeacherBox.isChecked() || isParentBox.isChecked() || isAdminBox.isChecked() ){
    Toast.makeText( context: RegistrationActivity.this,  text: "Select the Account Type", Toast.LENGTH_SHORT).show();
    return;
}
```

Figure 10: Code for checkbox validation

From Figure 10, the checkbox validation code is displayed. The user must tick one of the checkboxes or this code will return "Select the Account Type" if the user did not tick any of the checkboxes.

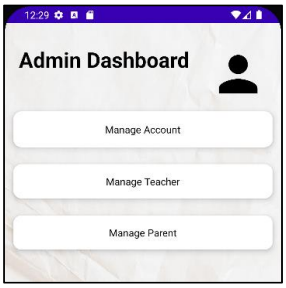


Figure 11(a): Admin dashboard interface

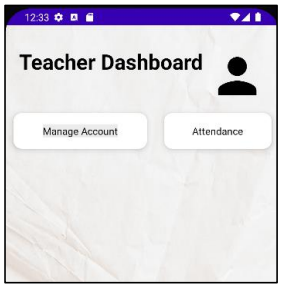


Figure 11(b): Teacher dashboard interface

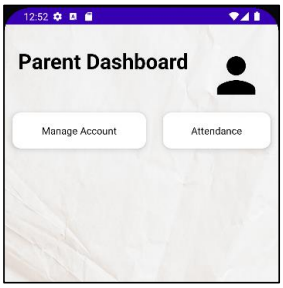


Figure 11(c): Admin dashboard interface

Other than registering and login based on role, each user can access their section by role. Figure 11(a) shows the admin dashboard interface in which the admin can manage account, manage teacher and manage parent. Figure 11(b) and Figure 11(c) display the teacher and parent dashboard interface which they can only access manage account and attendance module. A similar approach was applied in previous studies such as those of [14] that provide secure and trusted framework for a mobile cloud system, and those of [15] that provide secure access control for a mobile health application. As in-line with the OWASP recommendations [16], TIKAS can minimize the risk of broken access control, and becomes a trusted application to TIKAS's users.

The other security feature that was implemented in the application is MFA. Users must enter their password and One-Time Password (OTP) to complete the registration.

```
if (isAdminBox.isChecked()){ //user choose role as admin
    startActivity(new Intent(getApplicationContext(), SendOTPActivity.class)); //user will go to send OTP page
    finish(); //user cannot go back to register
}

if (isTeacherBox.isChecked()){ //user choose role as teacher
    startActivity(new Intent(getApplicationContext(), SendOTPActivity.class));
    finish(); //user cannot go back to register
}

if (isParentBox.isChecked()){ //user choose role as parent
    startActivity(new Intent(getApplicationContext(), SendOTPActivity.class));
    finish(); //user cannot go back to register
}
```

Figure 12: Code for register button

The code from Figure 12 shows that once the user clicks the register button with complete user details and chosen role, user will go to OTP page.

```
getOTPbtn.setOnClickListener(new View.OnClickListener() {  //code for user to get OTP
    @Override
    public void onClick(View view) {
        if (inputPhone.getText().toString().trim().isEmpty()) {
            Toast.makeText( context: SendOTPActivity.this,  text: "Enter Phone Number", Toast.LENGTH_SHORT).show(
            return;
        }
        progressBar.setVisibility(view.VISIBLE);
        getOTPbtn.setVisibility(view.INVISIBLE);

        PhoneAuthProvider.getInstance().verifyPhoneNumber(
                phoneNumber: "+60" + inputPhone.getText().toString(),
                timeout: 60,
                TimeUnit.SECONDS, //once sent, user can't get new code for the next 60 seconds
                SendOTPActivity.this,
                new PhoneAuthProvider.OnVerificationStateChangedCallbacks(){
```

Figure 13(a): Code for user to receive OTP

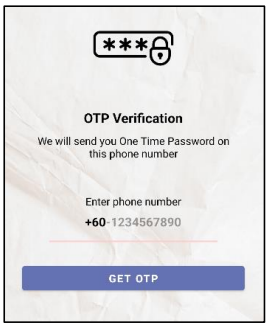


Figure 13(b): Interface to enter phone number to get OTP code

In Figure 13(a), it shows the code for user to receive OTP. Users are required to enter their phone number. Once the user clicks the get OTP button, the user will receive a six-digit code through a message at the phone number entered earlier and then if the user wants to resend the code, the user must wait for a minute as user cannot get a new code for the next 60 seconds. Figure 13(b) shows the interface of user to get OTP code after entering their phone number.

Commented [NR1]: please also add these sentences from thesis:
A similar approach was applied in previous studies such as those of [25] that provide secure and trusted framework for a mobile cloud system, and those of [26] that provide secure access control for a mobile health application. As in-line with the OWASP recommendations [27], TIKAS can minimize the risk of broken access control, and becomes a trusted application to TIKAS's users.

```
buttonVerify.setOnClickListener(new View.OnClickListener() {  //code for user to validate OTP received
    @Override
    public void onClick(View view) {

        if (inputCode1.getText().toString().trim().isEmpty()
            || inputCode2.getText().toString().trim().isEmpty()
            || inputCode3.getText().toString().trim().isEmpty()
            || inputCode4.getText().toString().trim().isEmpty()
            || inputCode5.getText().toString().trim().isEmpty()
            || inputCode6.getText().toString().trim().isEmpty()) {
            Toast.makeText( context: VerifyOTPActivity.this, text: "Please enter valid code", Toast.LENGTH_SHORT).show()
            return;
        }
        String code=
                inputCode1.getText().toString() +
                        inputCode2.getText().toString() +
                        inputCode3.getText().toString() +
                        inputCode4.getText().toString() +
                        inputCode5.getText().toString() +
                        inputCode6.getText().toString();
```

Figure 14(a): Code for user to enter and verify OTP received

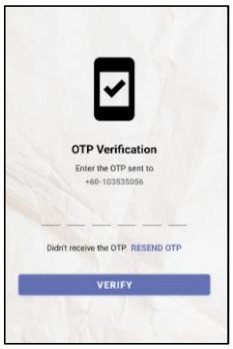


Figure 14(b): Interface to verify the OTP received

Figure 14(a) explains the code for the user to enter the code received and click on the verify button. If the code is invalid, a message “Please enter valid code” will pop up. Figure 14(b) shows the interface for user to enter the OTP received and verify the code.

```
if(verificationId != null) {
    progressBar.setVisibility(View.VISIBLE);
    buttonVerify.setVisibility(View.INVISIBLE);
    PhoneAuthCredential phoneAuthCredential = PhoneAuthProvider.getCredential(
            verificationId,
            code
    );
    FirebaseAuth.getInstance().signInWithCredential(phoneAuthCredential) //code for OTP verified
            .addOnCompleteListener(new OnCompleteListener<AuthResult>() {
                @Override
                public void onComplete(@NonNull Task<AuthResult> task) {
                    progressBar.setVisibility(View.GONE);
                    buttonVerify.setVisibility(View.VISIBLE);
                    if (task.isSuccessful()) {
                        Intent intent = new Intent(getApplicationContext(), LoginActivity.class);
                        intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK | Intent.FLAG_ACTIVITY_CLEAR_TASK);
                        startActivity(intent);
                    } else {
                        Toast.makeText( context: VerifyOTPActivity.this, text: "The verification code entered was invalid", Toast.LENGTH_SHORT).show();
                    }
                }
            });
}
```

Figure 15: Code for OTP verification

Figure 15 describes the code for OTP verification. Line 14 shows that the `if` function to go to login module once user clicks the verify button and the code is successfully verified, but if user enter the wrong code the `else` function will run, and the “The verification code entered is invalid” message will pop up.

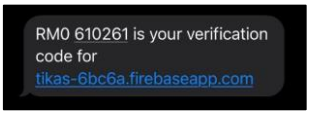


Figure 16: Verification code received by user

In Figure 16, the user receives the OTP verification code once the user enters the correct format of a phone number and enters click get OTP button. It indicates that TIKAS has successfully implemented MFA that are: (i) something you know and (ii) something you have, factors.

```
findViewById(R.id.resendOTP).setOnClickListener(new View.OnClickListener() { //code for user to get a new code
    @Override
    public void onClick(View view) {
        PhoneAuthProvider.getInstance().verifyPhoneNumber(
                phoneNumber: "+60" + getIntent().getStringExtra( name: "phone"),
                timeout: 60,
                TimeUnit.SECONDS, //once sent, user can't get new code for the next 60 seconds
                VerifyOTPActivity.this,
                new PhoneAuthProvider.OnVerificationStateChangedCallbacks(){

                    @Override
                    public void onVerificationCompleted(@NonNull PhoneAuthCredential phoneAuthCredential) {

                    }

                    @Override
                    public void onVerificationFailed(@NonNull FirebaseException e) {

                        Toast.makeText( context: VerifyOTPActivity.this, e.getMessage(), Toast.LENGTH_SHORT).show();
                    }

                    @Override
                    public void onCodeSent(@NonNull String newverificationId, @NonNull PhoneAuthProvider.ForceResendingToken forceResendingToken) {
                        verificationId= newverificationId;
                        Toast.makeText( context: VerifyOTPActivity.this, text: "OTP Sent", Toast.LENGTH_SHORT).show();
                    }
                }
        );
```

Figure 17: Code for user to get new OTP code

From Figure 17, a user will receive an OTP code after clicking on the resend button. The new code is the same as the previous code as the code is declared as `verificationId=newverificationId`.

Input validation is applied to sanitize only the correct input with the correct format. The data that follows the format will be validated. For instance, if the user enters the wrong email address, an error message will appear as in Figure 18.

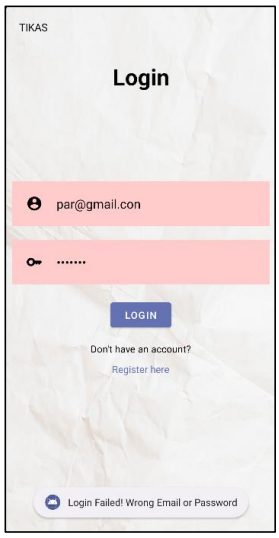


Figure 18: Pop-up message for wrong address format

```
public boolean checkField(EditText textField) {
    if (textField.getText().toString().isEmpty()) {
        textField.setError("Error");
        valid = false;
    } else {
        valid = true;
    }

    return valid;
}
```

Figure 19(a): Code to validate the text field

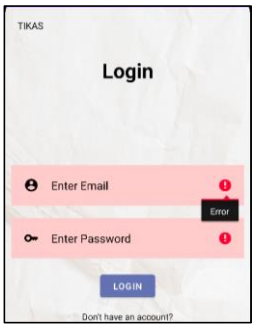


Figure 19(b): Pop-up message for input validation on the login page

Figure 19(a) depicts the code to validate whether the user fills in the text field or not. If the `checkField` returns a true value, then no error message will appear. Figure 19(b) describes a pop-up message that will appear if the user does not enter the correctly formatted data to log into the application.

4.2 Test Plan Table

Table 5 shows the result of the security test which examined the system security function, and the result of functional testing is shown in Table 6. The result turns out to be passed for all the tests.

Table 5: Result of security test plan

No.	Checklist	Actual Result
1.	Users should be able to register based on their role.	Pass
2.	Users should be able to verify their identity by receiving OTP with six digits.	Pass
3.	Users should not be able to register if the OTP received does not match the input entered by the user.	Pass
4.	Users should be able to log in based on their role after successfully registering.	Pass
5.	Users should be able to log in with a valid email and password.	Pass
6.	Users should not be able to log in with an invalid email and password.	Pass
7.	Teachers can access their section based on their role.	Pass
8.	Admin can access their section based on their role.	Pass
9.	Guardians can access their section based on their role.	Pass

Table 6: Result of functional test plan

No.	Checklist	Actual Result
1.	Validate all required input fields.	Pass
2.	All application links are working without any problem.	Pass
3.	All buttons in the application should be working and easy to view by the users.	Pass
4.	Users should be able to go through the verification process when registering.	Pass
5.	Users should be able to register.	Pass
6.	Users should be able to log in.	Pass
7.	Users should be able to log out.	Pass
8.	Users should be able to access the dashboard.	Pass
9.	Users should be able to manage their account.	Pass
10.	The teacher should be able to record student attendance for students that attend on the day.	Pass
11.	Teacher should be able to record student attendance for student that absent on the day.	Pass
12.	Parents should be able to upload images as evidence of absence such as Medical Certificate (MC).	Pass

4.3 User Testing

User testing has been done with one volunteer of each role. The testing was conducted online. The testing was done by distributing an apk file to the principal of the kindergarten and give out a Google Form to answer the question. As a result, Figure 20 displays that all the users can register based on their role. Next, the users can log in based on their role after successfully registering. Moreover, users can only log in with a valid email and password. Additionally, each role which is admin, teacher, and guardian can access their section based on their role. Figure 21 described application functional testing. From the observation, all tests are passed. All input fields could be validated. All users can register, log in, and log out.

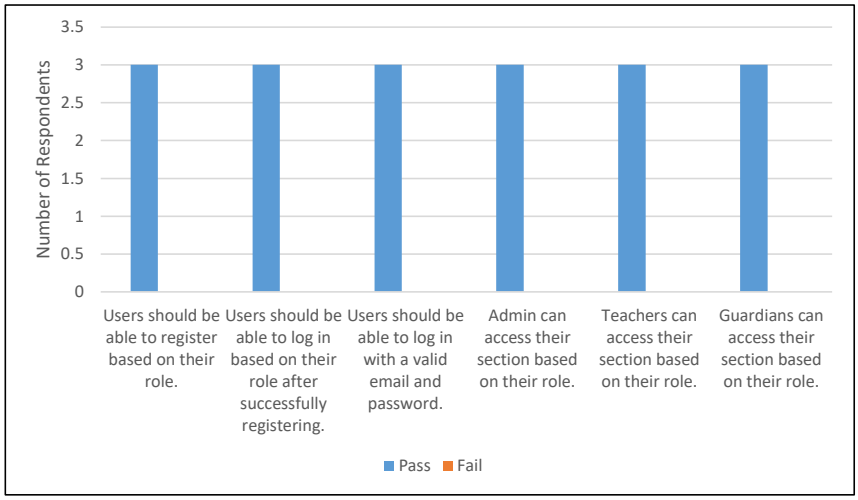


Figure 20: Application security testing result

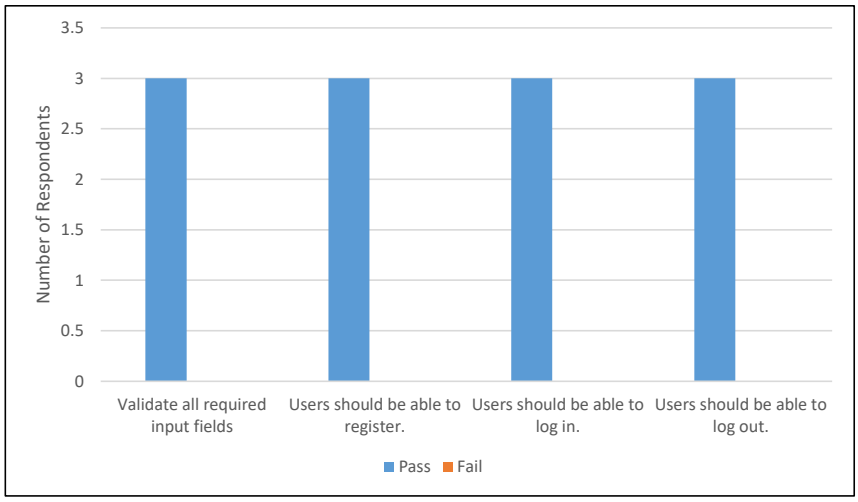


Figure 21: Application functional testing result

From the application security testing result and security test plan result, all the security features which are RBAC, OTP, and input validation are successfully developed in the application.

5. Conclusion

In conclusion, TIKAS: Kindergarten Attendance Application with Role-Based Access Control for Tadika Inovasi Kreatif is a mobile application that is very accommodating for teachers and parents. The main security feature of the application is role-based access control to prevent any alteration from unauthorized users. In the future, more security should be added to strengthen the integrity and confidentiality of the data and add more features to make it more user-friendly. For instance, users should be able to recover their password to make sure that they are able to recover their account back in case they forgot the password. Moreover, a security log should be added to the application to monitor the activity of the user. Lastly, the application should be able to be used by all types of mobile operating systems such as Android Operating System, Harmony operating system, and iPhone Operating System (iOS).

Acknowledgment

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

References

[1] OWASP, "OWASP Top Ten." 2021. Accessed: Nov. 13, 2023. [Online]. Available: https://owasp.org/www-project-top-ten/

[2] M. E. Shacklett, "What is multifactor authentication and how does it work?," Nov. 03, 2021. Accessed: Nov. 11, 2022. [Online]. Available: https://www.techtarget.com/searchsecurity/definition/multifactor-authentication-MFA

[3] Z. Doffman, "Microsoft Wants You To Stop Using SMS Verification Messages: Here's Why You Should Be Concerned," *Forbes*, Nov. 17, 2020. Accessed: Nov. 12, 2022. [Online]. Available: https://www.forbes.com/sites/zakdoffman/2020/11/17/microsoft-warning-about-sms-security-codes-sent-to-apple-iphone-and-google-android-phones/?sh=7589f33f242f

[4] S. Thielman and N. York, "Yahoo hack: 1bn accounts compromised by biggest data breach in history," Dec. 2016. Accessed: Nov. 12, 2022. [Online]. Available: https://archive.comsuregroup.com/wp-content/uploads/2018/01/Yahoo-hack_-1bn-accounts-compromised-by-biggest-data-breach-in-history-_-Technology-_-The-Guardian.pdf

[5] T. Cosoleto, "Bank details might be accessed in Tas govt cyber hack," *The Canberra Times*, Apr. 05, 2023. Accessed: May 18, 2023. [Online]. Available: https://www.canberratimes.com.au/story/8149785/bank-details-might-be-accessed-in-tas-govt-cyber-hack/

[6] M. Teo, "'Be careful where you bring your children': Man claims 2 women tried to kidnap his friend's son at Resorts World Genting," Sep. 22, 2022. Accessed: Nov. 13, 2022. [Online]. Available: https://www.asiaone.com/malaysia/be-careful-where-you-bring-your-children-man-claims-2-women-tried-kidnap-his-friends-son

[7] FMT. Reporters, "1,509 children have gone missing since 2020, says Bukit Aman," Apr. 15, 2022. Accessed: Nov. 13, 2022. [Online]. Available: https://www.freemalaysiatoday.com/category/nation/2022/04/15/1509-children-have-gone-missing-since-2020-says-bukit-aman/

[8] R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, "Computer role-based access control models," *Computer*, vol. 29, no. 2. pp. 38–47, Feb. 1996. doi: 10.1109/2.485845.

[9] M.- Shiang, C.-C. Lee, and Y.- Liang, "A Simple Remote User Authentication Scheme," 2002. Accessed: Dec. 11, 2022. [Online]. Available: www.elsevier.com/locate/mcm

[10] N. A. Alias, "Attendance System Using Face Recognition," Universiti Tun Hussein Onn Malaysia, Batu Pahat, 2020. Accessed: Dec. 12, 2022. [Online]. Available: http://archive.uthm.edu.my.ezproxy.uthm.edu.my/bitstream/123456789/3883/1/FKEE_2020_NUR%20AMIRA%20ALIAS.pdf

[11] N. Q. Shafee, "Student Attendance System For Parents," Universiti Tun Hussein Onn Malaysia, Batu Pahat, 2021. Accessed: Dec. 12, 2022. [Online]. Available: http://archive.uthm.edu.my.ezproxy.uthm.edu.my/bitstream/123456789/7027/1/FSKTM_2021_NUR%20QAMARINA%20BINTI%20MOHD%20SHAFEE_AI180057.pdf

[12] T. W. Yee, "Facial Recognition Attendance System Using CNN," Universiti Tun Hussein Onn Malaysia, Batu Pahat, 2021. Accessed: Dec. 12, 2022. [Online]. Available: http://archive.uthm.edu.my.ezproxy.uthm.edu.my/bitstream/123456789/7023/1/FSKTM_2021_Tan%20Wa%20iYee_Ai180234.pdf

[13] R. S. Sandhu and P. Samarati, "Access control: principle and practice," *IEEE Communications Magazine*, vol. 32, no. 9, pp. 40–48, Sep. 1994, doi: 10.1109/35.312842.

[14] A. M. Abdul *et al.*, "Enhancing Security of Mobile Cloud Computing by Trust- and Role-Based Access Control," *Sci Program*, vol. 2022, 2022, doi: 10.1155/2022/9995023.

[15] S. A. Butt, T. Jamal, M. A. Azad, A. Ali, and N. S. Safa, "A multivariant secure framework for smart mobile health application," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 8, p. e3684, 2022.

[16] OWASP, "Access Control," 2021. https://owasp.org/www-community/Access_Control (accessed Jun. 21, 2023).