Applied Information Technology And Computer Science Vol. 4 No. 2 (2023) 316-333

AITCS

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

Dynac Visitor Management System with Facial Recognition

Abdul Hazim Abd Hafiff[1], Shamsul Kamal Ahmad Khalid[2]

[1]Fakulti Sains Komputer dan Teknologi Maklumat,
Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA

DOI: https://doi.org/10.30880/aitcs.2023.04.02.019
Received 08 November 2023; Accepted 08 November 2023; Available online 30 November 2023

Abstract: This paper presents the development of a computerized visitor management system with facial recognition for Dynac Sdn Bhd. The existing manual paper-based registration system has led to some human errors and inefficiencies. The proposed system integrates a facial recognition function to increase efficiency when checking-in and checking-out visitors. It includes digitalizing the registration form, utilizing a web camera for face recognition, and implementing security measures. The web-based system manages to improve the registration efficiency, streamlined visitor management, and enhance security. The system still has some room for improvement such as expanding the visitor classification, implementing additional notification methods and faster facial recognition engines. This system significantly improves visitor management and enhances the overall visitor experience. Several experiments have been conducted with the target users. The system passed all (100%) security checklist planned for the system including user, visitor management and system trail logs.

Keywords: Management System, Web-based, Facial Recognition, TenserFlow, reCaptcha

1. Introduction

The visitor management system is the first line of defense for most organizations and institutions as they help protect against unwanted visitors from entering secure premises without proper authorization [1]. Visitor management systems record vital information such as the incoming and outgoing visitors within the organization [2]. The visitor management system has been around for quite some time now and has been used in many fields of work. However, there are many organizations and institutions that still utilize the manual paper-based visitor log to record their visitor's visitation. This manual method is easy to use, but it takes longer time to process especially when dealing with multiple numbers of visitors within a short period of time [3]. With the current era of digitalization, manual based visitor management could cause a lot of organizational issues [4].

Dynac Sdn Bhd, a company operating for over 5 years, still relies on manual paper-based visitor registration. This outdated process is inefficient and prone to mistakes. Our goal is to develop a computerized visitor management system with facial recognition. The project will take about 40 weeks to complete and will be web-based, designed specifically for Dynac Sdn Bhd's security department. We'll use HTML, CSS, Bootstrap, JavaScript, and PHP, hosted on a cloud server managed by Hostinger.

**Corresponding author: shamsulk@uthm.edu.my*

The rest of the paper is organized as follows: Section 2 provides an overview of related works, Section 3 outlines the project's methodology, and Section 4 presents the overall results and discussion.

2. Related Work

In this section, we'll cover some important terms related to our project. Firstly, we'll delve into the concept of a visitor management system and what it entails. We'll also explore facial recognition technology, including the libraries and algorithms associated with it. Additionally, we'll touch upon security terms such as Captcha and RBAC (Role-Based Access Control). Lastly, we'll provide a comparison of other similar systems for a comprehensive understanding.

2.1 Visitor Management System

A visitor management system is designed to track and monitor visitors' actions within an organization or public institutions like schools or colleges [2]. Our daily activities often involve queries, particularly in office, and service-related settings such as hospitals, businesses, schools, and transportation. To handle this, most places have a reception desk at the entrance where visitors undergo security checks before being assisted by receptionists. [4]. This system serves as the first line of defense for organizations and institutions by preventing unauthorized individuals from entering secure premises [1]. It records essential information regarding visitor arrivals and departures within the organization [2].

2.2 Facial Recognition

Facial recognition is a method used to identify an individual's identity based on their facial features. It remains an active and highly researched field in computer science, constantly uncovering new ways to utilize and optimize its capabilities [5]. Companies like Facebook and Amazon have successfully employed facial recognition in various practical and commercial applications. In addition to facial recognition, other forms of biometric security include voice recognition, fingerprint recognition, and eye retina recognition [6]. Over the past three decades, facial recognition technology has gained significant attention and continues to evolve. Its simplicity and effectiveness in image analysis and pattern recognition applications contribute to its growing popularity[5].

2.3 TenserFlow.js

ThenserFlow.js is an open-source library that allow developers to run machine learning models and perform numerical calculations directly through the browser [7]. TenserFlow.js utilizes the widely known deep learning framework called TenserFlow and it provides a JavaScript-based functionality. One of the key features of TenserFlow.js is its ability to utilizes machine learning models entirely from the browser [7]. This aligns with our project objectives, which is to develop a web-based system. By utilizing TensorFlow.js we can bring machine learning capabilities to the browser, enabling us to perform client-side analysis without relying on server-side computation. This helps increase the speed of our web system and preserves user privacy, as the data is not being sent to external servers for processing.

2.4 Face-api.js

Face-api.js is a JavaScript module that is built on the TensorFlow kernel. TenserFlow.js utilizes convolutional neural networks (CNN) giving it the capability to do face detection, recognition, and face landmark detection [8]. One of the key capabilities that Face-api.js has is the ability to do face-based analysis directly in the browser environment. This aligns with our project goal which is to build a web-based system that has facial recognition capabilities. Additionally, because Face-api.js runs face-based analysis directly in the browser this helps lighten the backend burden by utilizing power from the front end [9]. Face-api.js facial recognition works by detecting and analyzing 68 points of face image tokens. It then compares the taken pictures with face descriptor of the reference data. Then a calculation will

happen between the two-description data to determine the threshold value, and the lower it is the higher the similarity [10].

2.5 Completely Automated Public Turning Test to tell Computer and Humans Apart (CAPTCHA)

CAPTCHA is designed as a security measure to distinguish between humans and automated bots or malicious software [11]. Bot is a type of malicious program that has the capability to run certain automated tasks. CAPTCHA is one of the protections that can be implemented to protect against Bot attacks. CAPTCHA comes in various forms such as text based, image based, audio based, video based, and puzzle based. The CAPTCHA implemented in this proposed system is the ReCaptcha which is a commonly used CAPTCHA created by google for preventing automated bots from conducting nefarious activities [12].

2.6 Role Based Access Control (RBAC)

Access control is the most basic and fundamental requirement to ensure safeguard of information assets within an organization. [13]. Generally, there are two fundamental types of access control which is Discretionary Access Control (DAC) and Mandatory Access Control (MAC). [13]. Role-Based Access Control (RBAC) is an access control that utilizes roles or groups to control user's permission. The user can obtain access permission by becoming a member or a role or group. Each role or group has their own specified permission [14]. This method of access control makes it easier for Admins to manage their users' permissions.

2.7 Study on Existing System

This section explores different systems that are like the one being developed. The first system is the currently used system at the company. Currently in use relies on a paper-based visitor management process. Visitors' complete forms at the guard house, providing their details and reason for the visit. These forms are stored physically, and visitor entries and exits are recorded in a logbook. After five years, old records are safely disposed of. Scheduled visits are usually pre-informed to the security department, but unannounced visitors require approval from the human resource department. The security department manages the forms, folders, and disposal of outdated records. For walk-in visitors, the guard informs the HR department, waits for approval, and issues visitor passes.

The second system is the SwipedOn Visitor Management System, developed by a software company based in New Zealand. It aims to simplify and enhance the visitor check-in process through features like self-service kiosks, digital document signing, automated badge printing, guest arrival notifications, pre-registration, and contactless check-in using QR codes.

The third system is the Envoy Visitor Management System by a San Francisco-based software company. Along with streamlining check-ins, it offers workplace technology solutions such as deliveries, room booking, and employee sign-ins. Its features include self-service kiosks, customizable badge printing, host notifications, pre-registration, and touchless check-in using QR codes or the Envoy mobile app.

Table 1 shows the comparison between the existing system, SwipedOn visitor management system, Envoy visitor management system and the proposed visitor management system (Dynac Visitor Management System with Facial Recognition). The current system used by the company has some limitations compared to other systems available. While it serves its purpose, it lacks certain features that could improve its efficiency. On the other hand, the proposed system shares similarities with two commercialized systems, but it has some differences in the check-in process. Unlike those systems, the proposed system doesn't have a self-service kiosk but offers self-registration for visitors. During check-in, an employee will still verify the registration and provide a visitor pass. Another distinction is the contactless check-in method. Systems 2 and 3 allow visitors to check-in using their mobile devices or

by scanning a QR code, whereas the proposed system uses facial recognition for contactless check-in and faster processing. Additionally, Systems 2 and 3 are application-based, while the proposed system is web-based, allowing access from any device with a browser.

Table 1: Comparison of Existing system with proposed system

Function	System 1 (Company system)	System 2 (SwipedOn)	System 3 (Envoy)	Proposed system
Self-service check-in	Yes (manual logbook)	Yes (Kiosk)	Yes (Kios)	No (Only for registration)
User management	No	Yes	Yes	Yes
RBAC	Yes (manually)	Yes	Yes	Yes
Visitor management	Yes (manually)	Yes	Yes	Yes
Visitation notification	No	Yes	Yes	Yes
Pre-registration	No	Yes (Host & admin pre-register)	Yes (Host & admin pre-register)	Yes (visitor pre-register themselves)
Contactless check-in	No	Yes (mobile app & Qr scan)	Yes (mobile app & Qr scan)	Yes (Facial recognition)
System type	Paper based	App based	App based	Web based
Dedicated database	No (manual file & logs)	Yes (Cloud)	Yes (Cloud)	Yes (Cloud)
Integration capabilities	No	Yes	Yes	Yes

3. Methodology/Framework

This section describes all the necessary information about the methodology used for the project development. This section also includes the analysis and design phase of the proposed system. results of each phase in the methodology are also included in this section in table form.

3.1 Prototype Model

The chosen process model for developing the system is the prototyping model. This decision was made to meet the customer's requirements effectively. By utilizing prototyping, we can provide the client with an early preview of the system interface through a draft design. This allows for customer feedback throughout the development process, enabling them to suggest improvements and enhancements if they find any areas lacking in the initial design. The prototyping model will involve five main phases or tasks, as depicted in Figure 1, which need to be accomplished to achieve the project's objectives and successfully deliver the developed system.

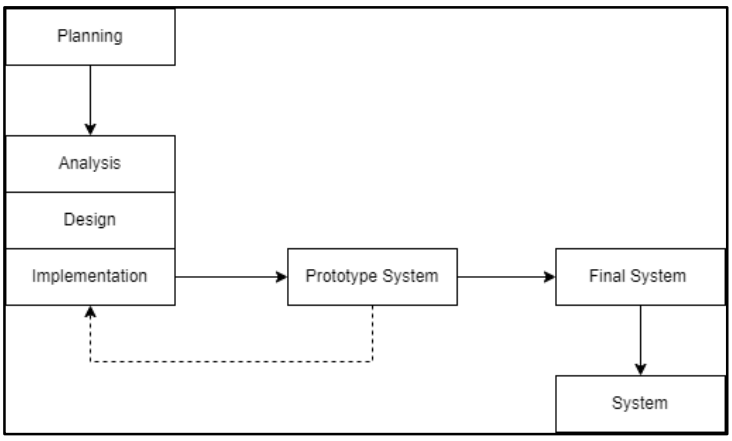

Figure 1: Prototype Model Methodology

Table 2 shows the whole flow of the prototype methodology and the activities done during each phase.

Table 2: Methodology Phases, Task and Outputs

Phases	Task	Output
Planning phase	• A short interview was done with the company's representative. • Determining the scope of the project • Determining the objectives of the project • Understanding issues with the current system • Determining the suitable project timeline • Conduct a site survey at the company premises	• Project proposal • Project scope • Project objectives • Project goals and aims. • Project Gantt chart
Analysis phase	• Gather the system user requirements. • Analyse the type of data taken from the visitors. • Conduct an analysis of existing similar systems. • analysis of existing similar systems. • Understanding the defined user and system requirements	• Identified functional requirements. • Identified non-functional requirements. • Identified user requirements. • Identified stored visitor data. • Identified current system flow. • Identified necessary module and functions
Design phase	• Determining best tools and devices to use • Design the system Data Flow Diagram • Design the system Entity Relationship Diagram • Design the system context diagram. • Design the system UI illustration. • Design the system workflow illustration	• Identified feasible system functionality to implement. • Identified feasible system functionality. • User roles matrix • System GUI illustration design • System ERD design • System DFD design • System workflow illustration • System context diagram illustration design
Prototype system implementation phase	• Develop the system prototype. • Establish a database connection with the system. • Implement the system on a simulation server. • Test the system functionality and usability. • Send the prototype to the company representative to be evaluated. • Modify or change related system parts based on the reviews. • Continue to develop, review, and refine the prototype	• Prototype 1 • Prototype 1 review • Prototype 2 • Prototype 2 review • Prototype 3 • Prototype 3 review
Final system implementation phase	• Develop the finalized system. • Test the finalized system on a functional cloud hosting server. • Test the functionality and usability of the system. • Fix or modify any part of the system that is needed. • Create a system checklist	• Fully functional integrated system • System functionality test review • User acceptance test review

3.2 Analysis Phase

In the system requirement analysis, information such as system functional and non-functional requirement will be discussed. Other requirement analysis information includes the user requirements, hardware, and software requirements. The information gathered during the analysis phase formed the basis for the design phase. Functional and non-functional modules of the system were constructed and defined based on the data collected earlier.

3.3 Functional Requirements Analysis

Table 3 shows the system Functional Requirements for the proposed system. The functional requirements are requirements that are needed for the system to fulfil the aim and objectives of the project.

Table 3: System Functional Requirements

Modules	Descriptions
Login/ Logout	● The system should allow users to login using their username and password. ● The system should allow the users to input a valid username and password to log in as user. ● The system should have an alert if the user input is invalid. ● The user should be redirected to the appropriate homepage based on their user account type. ● The system should create a new session when a user logs in. ● The system should unset and destroy the current user session when the user clicked the logout button. ● The system should redirect user back to the login page when the user successfully logged out
User management	● The system should only allow officer type users to view user list page. ● The system should have an alert if an-authorized user tried to access the view user list page. ● The system should only allow officer type users to create new users. ● The system should have an alert if an-authorized user tried to access the register new user page. ● The system should allow users to edit their own account. ● The system should only allow officer type user to change or edit selected user password. ● The system should have an alert if an authorized user tried to access the change password page. ● The system should only allow officer type users to see the delete user account button. ● The system should not allow an un-authorized user account to see the delete button
Visitor management	● The system should allow only logged-in users to view the visitor list page. ● The system should have an alert if an unknown user tried to access the visitor list page. ● The system should allow logged-in users to register new visitors into the system. ● The system should have an alert if a visitor tried to register again while their data is in the system database. ● The system should allow logged in users to search for visitor information manually using the visitor ic/passport number. ● The system should allow logged in users to view detailed visitor information. ● The system should allow logged-in users to view the visitor visitation logs. ● The system should allow logged in users to insert purpose and select visitor attendee to check-in visitors. ● The system should allow logged in users to insert purpose and select visitor attendee to check-in visitors. ● The system should send out an email notification to selected attendees when a visitor checked in. ● The system should allow logged-in users to check-out visitors. ● The system should only allow officer type user to see the delete visitor visitation logs button
Face recognition	● The system should only allow logged-in users to utilize the face recognition function. ● The system should have an alert if an unknown user tried to utilize the face recognition function. ● The system should be able to only detect human faces. ● The system should be able to detect known and unknown faces. ● The system should be able to redirect the user to the correct known visitor information page. ● The system should have an alert if an unknown visitor is detected.

Table 3: (Cont.)

Modules	Descriptions
System trail log	● The system should record failed login attempts into the system trail logs. ● The system should record successful login attempt into the system trail logs. ● The system should record user account creation actions into the system trail logs. ● The system should record user account update actions into the system trail logs ● The system should record change password actions into the system trail logs. ● The system should record deleted user account actions into the system trail logs. ● The system should record new visitor registration actions into the system trail logs. ● The system should record visitor check-ins actions into the system trail logs. ● The system should record visitor check-out actions into the system trail logs.
System trail log	● The system should record deleted visitor visitation actions into the system trail logs. ● The system should record delete visitor information actions into the system trail logs.

3.4 Non-Functional Requirements Analysis

Table 4 shows the non-functional requirements of the proposed system. The accessibility and compatibility requirements focus on ensuring the accessibility of the system across different platforms and devices. The usability and user experience requirements focus on how user-friendly the system is. The availability and reliability of the system requirements focus on the operational time of the system. The performance requirements focus on the systems performance in terms of accuracy and speed of the system response time. The security requirements focus on the security aspect of the system.

Table 4: Non-Functional Requirements

Requirements	Description
Accessibility and Compatibility	● The system should be able to function on any web browser. ● The system should be able to function on any computer with a web camera. ● The system should be able to function on any supported mobile devices.
Usability and User Experience	● The system should be user-friendly and easy to navigate. ● The system should be able to send out an informative email notification.
Availability and Reliability	● The system should be available at any time. ● The system should be able to utilize a web camera.
Performance	● The system should be able to detect human faces in less than 1 minute. ● The system should be able to recognize human faces.
Security	● The system should only be usable by logged-in users. ● The system should utilize a strong password scheme. ● The system should hash and salt users' account passwords. ● The system should have functional input data validation. ● The system should utilize data sanitization. ● The system should utilize role-based access control. ● The system should have HTTPS encryption.

3.5 User Requirements Analysis

Table 5 shows the system users requirements for the proposed system. The user requirements are the expectation, specific needs and features that users expect to have on the proposed system.

Table 5: User Requirements

No.	User Requirements
1	Users should be able to log into the system using their username and password given by the officer(admin).
2	The system should provide a user-friendly interface that is easy to navigate.
3	User should be able to update their profile information except for their account password.
4	Officer type user should be able to view list of users within the system.
5	Officer type user should be able to edit other users' password in the system.
6	Officer type user should be able to delete unused user accounts.
7	The system should allow logged in user to search for specific visitor information within the system.
8	The system should allow visitors to pre-register their information.
9	The system should allow logged in users to utilize the facial recognition function for a more efficient visitor check-ins and check-out process.
10	The system should allow logged in users to check-in and check-out visitors.
11	The system should provide an informative email notification notifying attendee that a visitor is awaiting their presence.
12	The system should generate visitor visitation logs and records for security and compliance purposes.
13	The system should be able to track the users' activities within the system.
14	The system should provide a secure and reliable environment to protect user data and ensure confidentiality.
15	The system should comply with the PDPA security act.

3.6 Hardware and Software Requirements Analysis

Table 6 and 7 shows the hardware and software of the proposed system. These requirements were based on the device used to develop the proposed system.

Table 6: System Hardware Requirements

Devices	Description
Computer/ Laptop	• Operating System: Windows 7/8/8.1/10 • Memory (RAM): 2 GB of RAM required. • Hard Disk Space: 250 MB of free space required for full installation. • Processor: Intel Pentium 4 Dual Core GHz or higher. • Stable internet connection
Camera	Camera that can transfer data through any form of media (usb, network line)

Table 7: System Software Requirements

Type	Software	Functions
Operating system	• Windows 10 Home 64-bit	Operating system used for developing the system
Programming editor/ compiler	• Sublime	Sublime is used for developing the system that uses web language
Database	• MySQL	Run and build the database
Design tool	• Draw.io	Designing the system database and flow chart
Programming Language	• PHP • HTML • Bootstrap • Java • CSS	Programming language used to build the system
Server application	• Hostinger	Web server used to host the system. Hostinger is an online web hosting service

3.7 User Role Matrix Analysis

Table 8 shows the system user role matrix. The system has two types of users and one unofficial user. Each user type has their own designated roles and capabilities. The table is divided into categories based on the system requirements modules.

Table 8: System User Role Matrix

Modules	Functions	Security officer (Admin)	Employees (Hr/Security Guards)	visitors
Login/Logout	Login	√	√	X
	Logout	√	√	X
User Management	Create new users	√	X	X
	View user lists	√	X	X
	Edit selected user password	√	X	X
	Edit personal profile	√	√	X
	Delete user account	√	X	X
Visitor Management	Register visitors	√	√	√
	View visitor lists	√	√	X
	View visitor visitation logs	√	√	X
	Check-in visitors	√	√	X
	Check-out visitors	√	√	X
	Delete visitor information	√	X	X
	Delete visitor visitation logs	√	X	X
Face Recognition	Scan for visitor information	√	√	X
System Trail Logs	View system trail logs	√	X	X
	Download system trail logs	√	X	X

3.8 System Design Phase

In the design phase, various components were developed, including the user interface, system database, and system flow diagram. Output from this phase includes the system's Data Flow Diagram (DFD) and Entity Relationship Diagram (ERD), which were created based on the information obtained during the analysis phase. The context diagram illustrates the interaction between the system and external entities, while the ERD provides a detailed description of the entities and data types implemented in the proposed system.

3.9 System Context Diagram

The context diagram gives an overview of the Dynac Visitor Management System with facial recognition. It shows how different entities interact with the system. There are three entities: the security officer (admin), employees (like HR staff and security guards), and visitors. The security officer can manage user accounts, view/edit user info, and handle visitor tasks. Employees can access/edit their own accounts and perform visitor tasks. Visitors can pre-register for faster check-in/check-out. The diagram helps understand the system's flow. See Figure 2 for the context diagram.

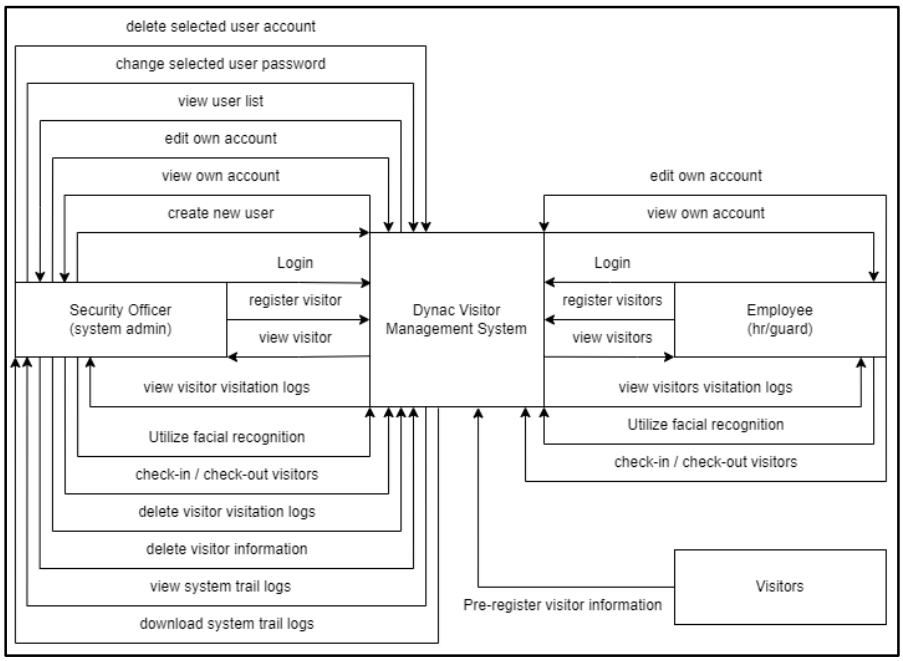

Figure 2: System Context Diagram

3.10 System Context Diagram

Figure 3 shows The Data Flow Diagram Level 0 breaks down the components of the Context Level Diagram in more detail. It shows the main functions of the system as sub-processes. The DFD Level 0 illustrates how entities interact with system processes and data storage.

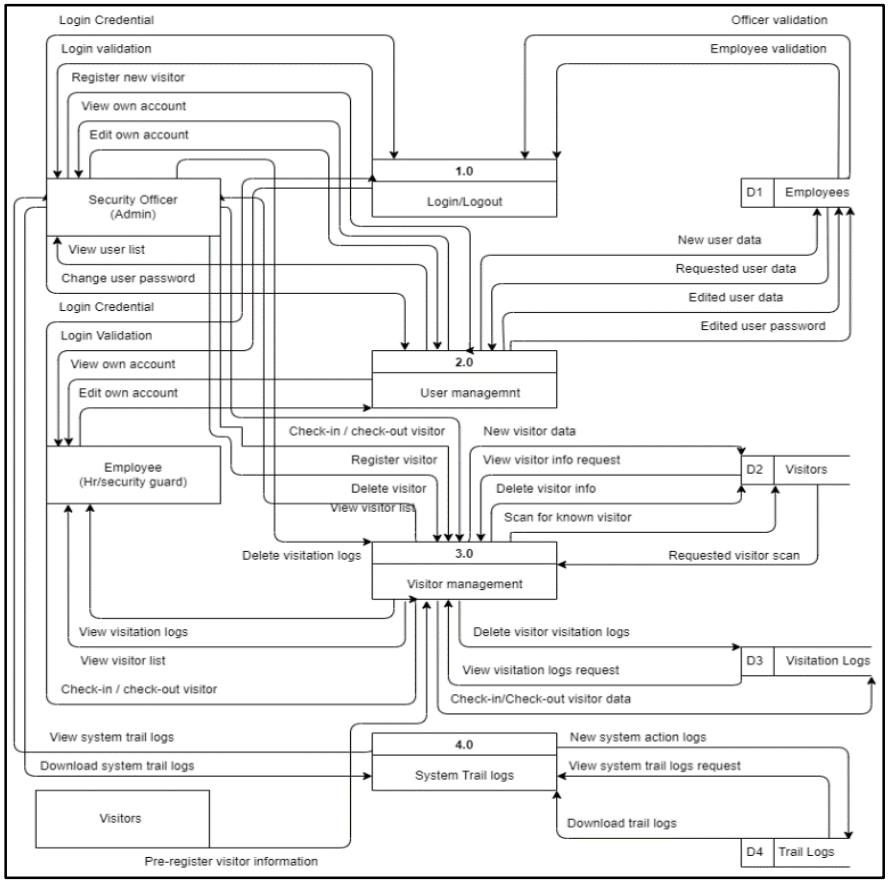

Figure 3: System Data Flow Diagram Level 0

3.11 System Entity Relationship Diagram

The Entity Relationship Diagrams (ERDs) help visualize and define the data and its relationships in the proposed Dynac Visitor Management System. It illustrates how entities are connected and the data that links them together. Figure 4 presents the ERD for the system, showcasing five entities: Employees, Visitors, Visitation Logs, Attendees, and Logs. Among them, Employees and Visitors are the parent entities. Figure 4 depicts the system's entity relationship diagram.

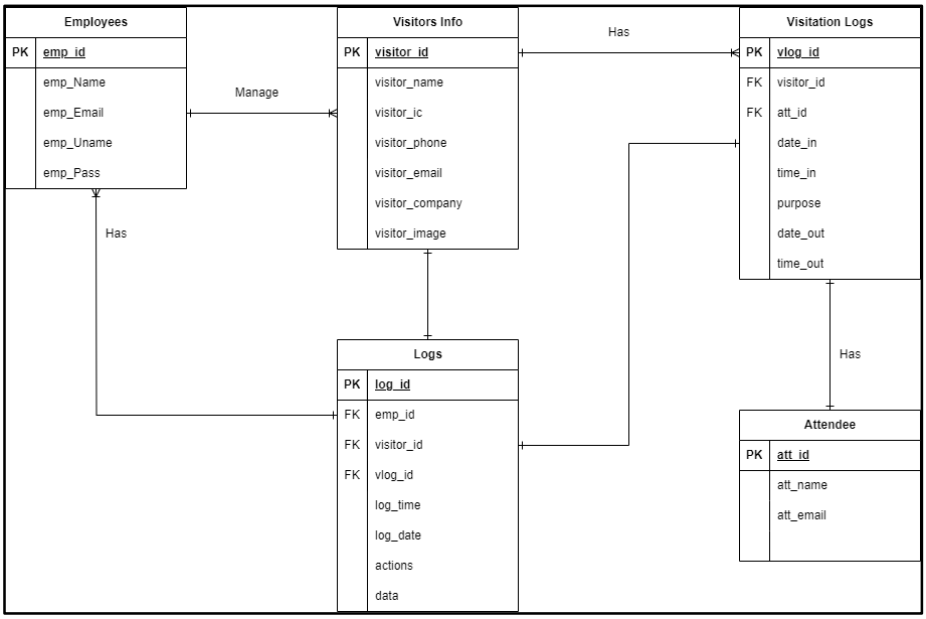

Figure 4: Entity Relationship Diagram

4. Result and Discussion

This section will show the overall result of the system, this includes the finalized and implemented system design. The implemented system design will be divided into two sections, first is the system security implementation and the second section is the system module implementation. Security implementation will focus on the security aspect of the system. The system module implementation will focus on the implemented functional module of the system and the overall finalized look of the system.

4.1 Security Implementation

There are eight total security modules implemented into the system to increase security. The security implementation includes form input validation, data sanitization, bind parameterized SQL query, Role-Base access control, use of strong password, password hashing, limited login attempt and Captcha.

4.2 Secure Coding Practice

Secure coding practices, including input validation, data sanitization, parameterized SQL queries, password hashing, strong password enforcement, and limited login attempts, are crucial for web application security. However, they are often overlooked or underestimated in web development. Many existing web applications are vulnerable to attacks due to poor coding practices and ad hoc development. To address this issue, developers can use defensive programming techniques to create more secure applications [15].

In our system, we have implemented input validation and error message handling for the login section. Input validation is applied throughout the system for modules and functions that require user data input, such as visitor registration, user account creation, and profile updates. Figure 5 shows a code

snippet that demonstrates how we validate user login data and enforce strong password requirements. These measures help enhance the security of our system.

```php
if (!preg_match("/^[a-z0-9A-Z ]*$/", $username)) {
    $username_error = "Only letters, numbers and white space allowed";
    $error_count++;
}
if (strlen($password) < 8 || !preg_match("/[A-Z]/", $password) || !preg_match("/[!@#$%^&*()_+]/", $password)) {
    $password_error = "Password must be at least 8 characters and contain at least one capital letter and one symbol.";
    $error_count++;
}

if ($error_count >= 1) {
    echo "<script type='text/javascript'>alert('Error, Try again');</script>";
}
```

Figure 5: Login input validation code snippet

Figure 6 shows the output error message for the login input text box. Additionally, when user input wrong credential or wrong username or password the error prompt will be the same in both case which is incorrect password or username. Without specifying which credential is wrong, this makes it harder for attacker to guess and bruit force the login function.

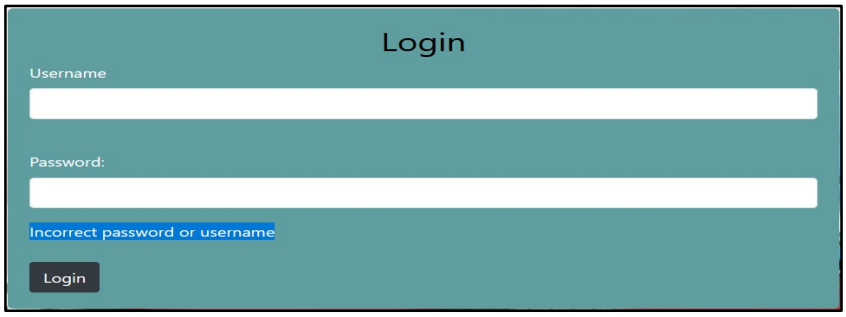

Figure 6: Error message for wrong login credentials

Figure 7 and figure 8 show the secure coding for the data input sanitization and parameterized SQL query code. As shown in figure 7, The code snippet “mysqli_real_escape_string” is used for data sanitization. This is used throughout the system where the user’s send in data to the server. This includes data for user registrations, personal account data update, visitor registrations and the system logs entry.

```php
$username = mysqli_real_escape_string($db, trim($_POST['username']));
$password = mysqli_real_escape_string($db, trim($_POST['password']));
```

Figure 7: Data Input sanitization code snippet

Figure 8 shows parameterized SQL query code for the login function. Parameterized SQL Query was used to reduce the chances of SQL Injection occurring to the system. This is done throughout the system, specifically parts that incorporate data submission to the server. This includes user login, user registrations, visitor registration, the visitor check-ins, and check-outs and the system logs module.

```php
// Check if the user is timed out
if (isset($_SESSION['last_attempt_time']) && time() - $_SESSION['last_attempt_time'] < 300) {
    $error = "Too many failed login attempts. Please try again later.";
} else {
    // Check if the email exists in the database
    $query = "SELECT * FROM employees WHERE emp_Uname = ?";
    $stmt = $db->prepare($query);
    $stmt->bind_param("s", $username);
    $stmt->execute();
    $result = $stmt->get_result();
    $user = $result->fetch_assoc();
    $stmt->close();

    if ($user) {
        // Check if the password is correct
        if (password_verify($password, $user['emp_pass'])) {
            if ($user['emp_type'] == 'officer') {
```

Figure 8: Parameterized SQL Query code snippet

Figure 9 displays the code snippet for the limited login attempt function in our system's user login section. It helps prevent brute force attacks by monitoring failed login attempts. If there are more than three failed attempts, the input box is locked for 5 minutes, denying further login attempts. Successful logins reset the attempt counter to 0. This feature helps enhance the system's security.

```
// Check if the number of failed attempts is stored in the session
    if (!isset($_SESSION['login_attempts'])) {
        $_SESSION['login_attempts'] = 0;
    }

// Check if the user is timed out
    if (isset($_SESSION['last_attempt_time']) && time() - $_SESSION['last_attempt_time'] < 300) {
        $error = "Too many failed login attempts. Please try again later.";
    } else {
        // Check if the email exists in the database
```

Figure 9: Limited login attempt code snippet

4.3 Password Hashing using (Bcrypt)

Bcrypt algorithm hashing was used to hash the user account password. This algorithm was chosen because it automatically incorporates salt into its hashing mechanism making it more random and unique. Figure 10 shows a code snippet used in user registration that incorporate Bcrypt hashing for the user account password.

```
if ($result->num_rows > 0) {
    $error = 'Username already exists';
    echo "<script type='text/javascript'>alert('Username already exists');</script>";
} else {
    // Hash the password
    $password = password_hash($password, PASSWORD_BCRYPT);

    // Prepare the SQL statement
    $query = "INSERT INTO employees (emp_Name, emp_Email, emp_Uname, emp_pass, emp_type) VALUES (?, ?, ?, ?, ?)";
    $statement = $db->prepare($query);
```

Figure 10: Password hashing code snippet

4.4 ReCAPTCHA

ReCaptcha was used for the visitor pre-registration function. Captcha was implemented for added security specifically to hinder bot-based attacks or spam attacks from happening on the visitor pre-registration page. Figure 11 shows the code snippet for the implemented reCaptcha and figure 12 shows the implemented captcha on the visitor pre-registration form page.

```
// reCAPTCHA configuration
$recaptcha_site_key = '6Lf-[redacted]';
$recaptcha_secret_key = '6Lf-[redacted]';

// Function to validate reCAPTCHA response
function validate_recaptcha($recaptcha_response, $recaptcha_secret_key)
{
    $url = 'https://www.google.com/recaptcha/api/siteverify';
    $data = array(
        'secret' => $recaptcha_secret_key,
        'response' => $recaptcha_response
    );
    $options = array(
        'http' => array(
            'header' => "Content-type: application/x-www-form-urlencoded\r\n",
            'method' => 'POST',
            'content' => http_build_query($data)
        )
    );
    $context = stream_context_create($options);
    $result = file_get_contents($url, false, $context);
    $response = json_decode($result);
    return $response->success;
}
```

Figure 11: Code snippet for implemented captcha

Figure 12: reCAPTCHA in visitor pre-registration page

4.5 System module implementation (User Management page)

Figure 13 shows the user management page of the system. The user management module includes viewing list of user accounts in the system, creating new user account, updating user personal information, changing user account password, and deleting user accounts. This page is only accessible by admin/officer type user account. Here admin/officer will have the options to register new users, edit each user's password or delete any unused user accounts.

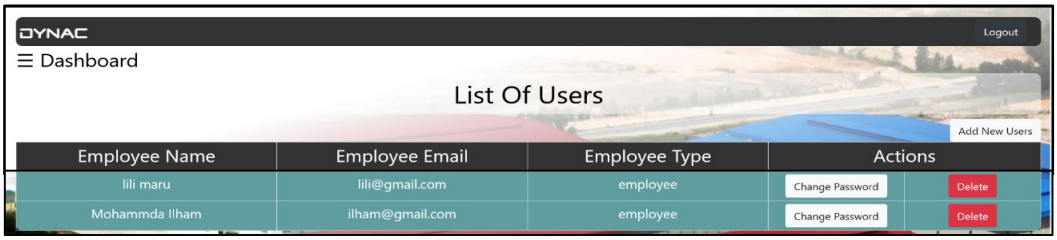

Figure 13: User Management Page

4.6 System module implementation (Visitor Management page)

In the visitor management page. The visitor management module includes viewing list of visitors in the system, registering new visitors, search for visitor information manually, view detailed visitor information and deleting visitor information. This page is accessible to both admin/officer and employees type accounts with sight difference being that the employee class will not be able to delete the visitor information.

In the visitor visitation logs page. Both admin/officer and employee user accounts can use the check-in and check-out function. However, only admin/officer accounts can see the delete visitor visitation log button. It's important to note that the button won't appear for other user types. Additionally, an email notification is sent to the attendee's email when a visitor checks-in to the premises. This is shown in figure 14.

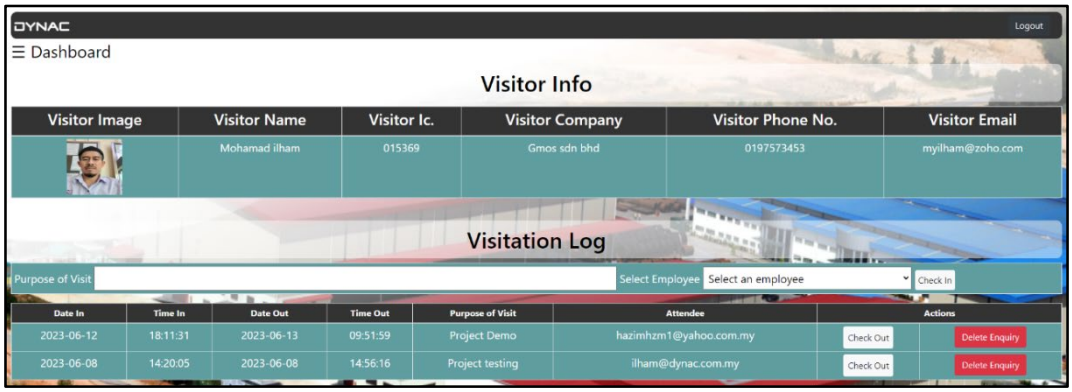

Figure 14: Visitor visitation logs page

4.7 System Module Implementation (System Logs page)

Figure 15 shows the system trail logs page. The system trail logs function is implemented throughout the whole system to monitor all action done by the system users. Action such as user login, user registration, user account update, visitor check-ins and check-outs. Any delete action such as delete user accounts and delete visitor information are also recorded in the trail log. Admins will also have the options to download the system trail logs into an excel file.

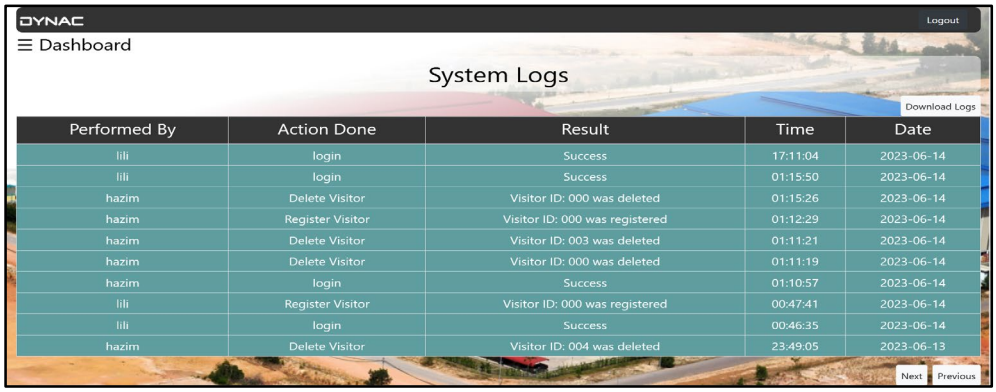

Figure 15: System Trail Logs page

4.8 System Functionality Testing

The functionality testing is done to ensure that the system works as intended and all modules are functional. Table 9 shows the testing result for the login and log out module. Table 10 shows the testing result for the user management module and all its functions. Table 11 shows the testing result for the visitor management module and including all its functions. Table 12 shows the system trail logs module testing result.

Table 9: Login and Logout module testing

	Test Cases	Expected Result	Actual Result
T1-1	Login with correct officer type email account and correct password	Login successful and redirect to officer home page	As expected
T1-2	Login with correct officer type email account and wrong password	An error message pop up informing wrong password or email	As expected
T1-3	Login with wrong officer type email account and correct password	An error message pop up informing wrong password or email	As expected
T1-4	Login with wrong officer type email account and wrong password	An error message pop up informing wrong password or email	As expected
T1-5	Login with correct employee type email account and correct password	Login successful and redirect to officer home page	As expected
T1-6	Login with correct employee type email account and wrong password	An error message pop up informing wrong password or email	As expected
T1-7	Login with wrong employee type email account and correct password	An error message pop up informing wrong password or email	As expected
T1-8	Login with wrong employee type email account and wrong password	An error message pop up informing wrong password or email	As expected
T1-10	User attempt to login with wrong credential for three times or more	The input text box for login will be lock and a message will show up saying too many failed attempt	As expected
T1-11	Users log out by pressing the logout button	User session will be destroyed and be redirected back to login page	As expected

Table 10: User Management Module Test Result

	Test Cases	Expected Result	Actual Result
T2-1	Try to view page while logged in as officer type user	Able to view page	As expected
T2-2	Try to view page while logged in as employee type user	An error message show saying "Not Authorized"	As expected
T2-3	Register new user account with correct data input	Registration successful and pop show up saying "new user created"	As expected
T2-4	Register new user account with incorrect data input	Error messages show up at the bottom of each related input text box notifying the correct data input	As expected
T2-5	Try to view edit account page	Can view edit account page and only own account data are shown	As expected
T2-6	Try to view edit password page while logged in as an employee type user	An error message show up saying "Not Authorized"	As expected
T2-7	Try to view edit password page for selected user while logged in as officer type user	Able to view edit password page of the selected user	As expected
T2-8	Try to change selected user password using correct data input while logged in as officer type user	Able to change selected user password and new password will be re-hashed	As expected
T2-9	Try to change selected user password using wrong data input while logged in as officer type user	Error message will show up at the bottom of input text box saying the requirement for the user password	As expected
T2-10	Delete button in the view user list page should only be visible to user type officer	Logged in user type officer can see and use the delete user account button	As expected
T2-11	Try to delete the selected user account	A pop up will show up asking for delete action confirmation	As expected

Table 11: Visitor Management module test result

	Test Cases	Expected Result	Actual Result
T3-1	Try to view visitor list page while logged in	Able to view visitor list page	As expected
T3-2	Try to view visitor list page while not logged in	An error message will pop up saying "please logged in first" and will be redirected to the login page	As expected
T3-3	Try to register visitor with correct data inputs	Able to register new visitor and pop message will show "New Visitor created; do you want to register another visitor?"	As expected
T3-4	Try to register visitor with incorrect data inputs	Error messages show up at the bottom of each related input text box notifying the correct data input	As expected
T3-5	Try to type in unknown visitor ic/passport number in the manual search box	A message will show up saying "no visitor with this ic existed"	As expected
T3-6	Try to type known visitor ic/passport number in the manual search box	The searched for visitor information will show up	As expected
T3-7	Try to view detailed visitor information page while logged in	Able to view the detailed visitor information page	As expected
T3-8	Try to view detailed visitor information page while no logged in	An error message will pop up saying "please logged in first" and will be redirected to the login page	As expected
T3-9	Delete button in the detailed visitor information page should only be visible to user type officer	Officer type account user should be able to see and use the delete button	As expected
T3-10	Try to delete the selected visitor visitation log	A pop up will show up asking for delete action confirmation	As expected

Table 11: Visitor Management module test result

T3-11	Delete button in the view visitor list page should only be visible to user type officer	Officer type account user should be able to see and use the delete button	As expected
T3-12	Try to delete the selected visitor	A pop up will show up asking for delete action confirmation	As expected
T3-13	Try to use face recognition while webcam is pointed at a non-human face	Face recognition module will not give any respond and keep searching for faces	As expected
T3-14	Try to use face recognition while webcam is pointed at a human face	Face recognition module can detect the face and draw a blue box around it	As expected
T3-15	Try to scan unregistered visitor using the webcam	Face recognition module will label the face as unknown, and a pop will show asking to try again or register new visitor	As expected
T3-16	Try to scan registered visitor using web cam	Face recognition module will recognise the face and redirect user to the appropriate visitor information page	As expected

Table 12: System Trail Logs test results

	Test Cases	Expected Result	Actual Result
T4-1	Try to view system trail logs page while logged in as officer	Able to view the system trail logs page	As expected
T4-2	Try to view system trail logs page while logged in as employee	An error message show up saying "Not Authorized"	As expected
T4-3	Try to utilise the download system trail log function button	Trail logs will be downloaded into an excel file	As expected

5. Conclusion

Overall, the system had achieved the objectives and requirements that had been set during the planning stage. The system was developed in accordance with the scheme set up during the design stage of the project. All the planned functions and modules have been implemented successfully and are working as intended. The system successfully improved the efficiency and effectiveness of the current visitor management by implementing a computerized visitor management system. Although the system has many advantages, it does still come with some limitations. Additional improvements can still be made to counter some of the system's limitations such as the slow facial recognition function, the limited visitation notification, and the lack of a dynamic visitor pass generator. Overall, by utilizing this system Dynac will no longer need to manually manage their visitors. Visitors will have a much easier time visiting and employees will have a much easier time managing their visitors.

Acknowledgements

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

References

[1] J. N. D. Rivera, "VMS Support: A mobile-based support to computerized visitor management system," *Software Impacts*, vol. 8, p. 100056, 2021, doi: https://doi.org/10.1016/j.simpa.2021.100056.

[2] D. Alkhodary, I. A. Abu-Alsondos, B. J. A. Ali, M. Shehadeh, and H. A. Salhab, "Visitor Management System Design and Implementation during the Covid-19 Pandemic," *Information Sciences Letters*, vol. 11, no. 4, pp. 1059–1067, Jul. 2022, doi: 10.18576/isl/110406.

[3] M. Oktaviandri and F. Kah Keat, "Design and Development of Visitor Management System," *mekatronika-Journal of Intelligent Manufacturing & Mechatronics*, vol. 01, pp. 73–79, doi: 10.15282/mekatronika.v1i1.152.

[4] I. A. B K and Akshay. P, "A Low Cost implementation of Visitor management system for small organisation or enterprise," in *2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC)*, 2021, pp. 502–506. doi: 10.1109/ICESC51422.2021.9533019.

[5] I. Adjabi, A. Ouahabi, A. Benzaoui, and A. Taleb-Ahmed, "Past, Present, and Future of Face Recognition: A Review," *Electronics (Basel)*, vol. 9, no. 8, 2020, doi: 10.3390/electronics9081188.

[6] Vandana and N. Kaur, "A Study of Biometric Identification and Verification System," in *2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, 2021, pp. 60–64. doi: 10.1109/ICACITE51222.2021.9404735.

[7] C. Gerard, "TensorFlow.js," in *Practical Machine Learning in JavaScript: TensorFlow.js for Web Developers*, Berkeley, CA: Apress, 2021, pp. 25–43. doi: 10.1007/978-1-4842-6418-8_2.

[8] H. Klym and I. Vasylchyshyn, "Face Detection Using an Implementation Running in a Web Browser," in *2020 IEEE 21st International Conference on Computational Problems of Electrical Engineering (CPEE)*, 2020, pp. 1–4. doi: 10.1109/CPEE50798.2020.9238754.

[9] X. Zhou, W. Hu, G.-P. Liu, and Z. Pang, "Face Recognition System Based on NCSLab for Online Experimentation in Engineering Education," in *2022 41st Chinese Control Conference (CCC)*, 2022, pp. 4390–4394. doi: 10.23919/CCC55666.2022.9902741.

[10] M. Alizadeh, K. Andersson, and O. Schelén, "DHT- and blockchain-based smart identification for video conferencing," *Blockchain: Research and Applications*, vol. 3, no. 2, p. 100066, 2022, doi: https://doi.org/10.1016/j.bcra.2022.100066.

[11] N. T. Dinh and V. T. Hoang, "Recent advances of Captcha security analysis: a short literature review," *Procedia Comput Sci*, vol. 218, pp. 2550–2562, 2023, doi: https://doi.org/10.1016/j.procs.2023.01.229.

[12] D. Wang, M. Moh, and T.-S. Moh, "Using Deep Learning to Solve Google reCAPTCHA v2's Image Challenges," in *2020 14th International Conference on Ubiquitous Information Management and Communication (IMCOM)*, 2020, pp. 1–5. doi: 10.1109/IMCOM48794.2020.9001774.

[13] M. Uddin, S. Islam, and A. Al-Nemrat, "A Dynamic Access Control Model Using Authorising Workflow and Task-Role-Based Access Control," *IEEE Access*, vol. 7, pp. 166676–166689, 2019, doi: 10.1109/ACCESS.2019.2947377.

[14] B. Cui, Z. Lan, and X. Bai, "Research on Role-based Access Control in IPv6 Smart Home," in *2019 IEEE 9th International Conference on Electronics Information and Emergency Communication (ICEIEC)*, 2019, pp. 205–208. doi: 10.1109/ICEIEC.2019.8784596.

[15] S. B. Rahayu, A. Ahmad, and S. H. Z. Rashid, "Defensive programming: Developing a web application with a secure coding practices," *AIP Conf Proc*, vol. 2617, no. 1, p. 050006, Nov. 2022, doi: 10.1063/5.0119726.