


Research Article

Implementation of One-Time Password and SHA-3 Algorithm on the Lab Inventory Website of the Department of Informatics and Computer Engineering

¹Muhammad Hakim Sadiq, ^{2*}Abdul Wahid , ³Mustari S. Lamada

^{1,2,3} Department of Computer Engineering, Makassar State University, South Sulawesi, Indonesia

* Corresponding Author: wahid@unm.ac.id

Abstract: This study evaluates the effectiveness of the One-Time Password (OTP) system on the Inventory Lab website of the Department of Informatics and Computer Engineering, focusing on OTP and user password security against Brute Force attacks. The objectives include testing OTP validation, analyzing OTP vulnerabilities to Brute Force attacks, and examining the resilience of user passwords under similar attacks. The study contributes to cyber security research by offering insights into implementing OTP and SHA-3 encryption algorithms on websites. Its findings aim to enhance the security measures of the Inventory Lab website. Results indicate that OTP delivery on the website is both successful and secure, with codes encrypted using SHA-3, rendering them unreadable in the database. OTP validation effectively distinguished correct and incorrect codes, including those that expired due to time limits. However, Brute Force trials on OTPs succeeded in some cases due to extended expiration times. Reducing the expiration period to one minute significantly minimized this risk. Similarly, trials on user passwords showed that passwords with complex character combinations resisted attacks more effectively than simpler ones. In summary, the OTP system and SHA-3 encrypted passwords demonstrate robust security but require adjustments to OTP expiration settings and stronger password policies to mitigate the risks of brute-force attacks. These improvements will further safeguard the website's security infrastructure.

Keywords: OTP, sha-3, brute force attack, cybersecurity, security infrastructure



Citation: Sadiq, M. H., Wahid, A., & Lamada, M. S. (2025). Implementation of One-Time Password and SHA-3 Algorithm on the Lab Inventory Website of the Department of Informatics and Computer Engineering. *Iota*, 5(2). <https://doi.org/10.31763/iota.v5i2.914>

Academic Editor: Adi, P.D.P

Received: April 02, 2025

Accepted: April 16, 2025

Published: May 07, 2025

Publisher's Note: ASCEE stays neutral about jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2025 by authors. Licensee ASCEE, Indonesia. This article is an open-access article distributed under the terms and conditions of the Creative Commons Attribution-Share Alike (CC BY SA) license(<https://creativecommons.org/licenses/by-sa/4.0/>)

1. Introduction

Currently, the internet a rapidly evolving technological innovation, facilitates information sharing, commerce, social networking, and entertainment. Many websites require users to create accounts for full access, offering benefits like easier information retrieval and streamlined processes. However, users often reuse simple passwords across platforms, making them vulnerable to hacking. Attackers can steal sensitive data or take over admin accounts, potentially altering websites maliciously. These threats affect all websites, not just those involving financial transactions. Implementing One-Time Passwords (OTP) offers a robust security solution. As single-use, time-limited codes, OTPs enhance website protection, reducing risks and providing users with greater security and confidence.

One-Time Password (OTP) is a security technology designed to protect user accounts from hackers. It is widely used for website logins and financial transactions, offering temporary, single-use passwords valid for a specific time. SHA-3 (Secure Hash Algorithm 3), a highly secure NIST standard, generates unique, tamper-resistant hashes for data protection. Implementing OTP with SHA-3 256 involves generating a 6-digit random code, encrypting it with SHA-3, and storing it in a database. The user receives the OTP, and upon entering it, the server matches the encrypted input with the stored hash. Successful matches grant access, ensuring robust security.

Rizki and Mulyati (2020), in their study titled "Implementasi One Time Password Menggunakan Algoritma SHA-512 Pada Aplikasi Penagihan Hutang PT. XHT", explored implementing OTP with SHA-512 in PT. XHT's debt collection web system. This solution addressed a data breach where hackers exploited weak client passwords to steal usernames and passwords. Another study by Lase and Mufti (2018), titled "Implementasi One Time Password (OTP) Mobile Token Dengan Menggunakan Metode Algoritma MD5 dan SHA", focused on countering replay and masquerade attacks. Their solution involved Android-based OTP Mobile Tokens secured with MD5 and SHA algorithms to enhance user and admin account security.

Lase and Mufti (2018) conducted a study titled "Implementation of One-Time Password (OTP) Mobile Token Using MD5 and SHA Algorithms", which addresses security threats such as replay attacks and masquerade attacks. The research focuses on securing user and administrator accounts by implementing an authentication process through an Android-based OTP Mobile Token utilizing MD5 and SHA algorithms.

Indonesia has experienced numerous data breaches, leading to compromised user accounts. According to cybersecurity firm Surfshark, 1.04 million accounts were breached in Q2 2022, a 143% increase from 430.1 thousand in Q1 2022. Since Q1 2020, the number of data breaches has fluctuated, peaking at 39.6 million accounts in Q2 2020. The figure declined to 669.4 thousand in Q2 2021, then rose again in Q3 2021. While data breaches decreased toward the end of 2021 and early 2022, they surged again in Q2 2022.

Based on the issues outlined in the background above, the researcher proposes a solution with the study titled "Implementation of One-Time Password Using SHA-3 Algorithm on the Laboratory Inventory Website of the Department of Informatics and Computer Engineering." This research focuses on how One-Time Password (OTP) enhances security in the laboratory inventory website.

2. Literature Review

A website consists of interlinked web pages and resources, typically accessed via a URL (Wahid et al., 2021), serving purposes like business, entertainment, and communication, and built using HTML, CSS, and PHP. Tim Berners-Lee introduced the World Wide Web, changing how information is accessed (CNBC Indonesia). Arief (2011) describes the web as an application containing multimedia documents accessed via a browser (Hasugian, 2018).

One-Time Password (OTP) is a security feature used in Two-Factor Authentication (2FA) and Multi-Factor Authentication (MFA), delivered via SMS or email. OTP prevents unauthorized access even if a password is compromised (Andi Rosano et al., 2018), and ensures security by preventing replay attacks (Schneier, 2015; Wibawa et al., 2024).

Before SHA-3, SHA-1 and SHA-2 were used in cryptographic hashing, but emerging vulnerabilities led to the development of SHA-3, based on the Keccak algorithm, to offer enhanced security (Dworkin, 2015). SHA-3's sponge construction provides flexibility and security benefits over SHA-1 and SHA-2.

The internet, originating from ARPANET in 1969, connects millions of devices globally, enabling communication, transactions, and access to services. Email remains a significant communication tool, despite the rise of other platforms (Herring et al., 2013), serving as a reliable method for OTP delivery.

The inventory tracks goods and resources within organizations, while an algorithm is a logical sequence for problem-solving in computing (Maulana, 2017). Data Flow Diagrams (DFD) visually represent data flow in systems, using symbols like entities, processes, and data stores (Li & Chen, 2009).

Entity-relationship diagrams (ERD) help model database structures, showing entities, attributes, and relationships (Afiifah et al., 2022), and are crucial for efficient database design. Visual Studio Code is a versatile, open-source development tool supporting multiple programming languages and extensions (Del Sole, 2021). XAMPP allows local PHP-based website management (Haerulah & Ismiyati, 2017), while flowcharts represent process steps clearly (Tuasamu et al., 2023). Burp Suite is a security tool for web application testing, identifying vulnerabilities through tools like Proxy Intercept, Spider, Scanner, and Intruder (Ni Putu Ana Rainita et al., 2023).

This research builds on previous OTP studies, like Ciputra (2017) and Rizki & Mulyati (2020), by proposing OTP implementation on the Universitas Negeri Makassar website using SHA-3 256, encrypted before being sent to users and expiring in one minute for enhanced security.

3. Method

This research employs an experimental approach, where hypotheses are tested through trials and observations to assess the impact of one variable on another. The experiment focuses on evaluating the validity of the OTP system and its security against brute force attacks, as well as assessing the effectiveness of SHA-3 encryption in protecting user data. Additionally, this study falls under applied research as it aims to develop practical solutions to enhance the security of the laboratory inventory system by implementing specific technologies, namely OTP and SHA-3. Figure 1 is the research stage built in this research.

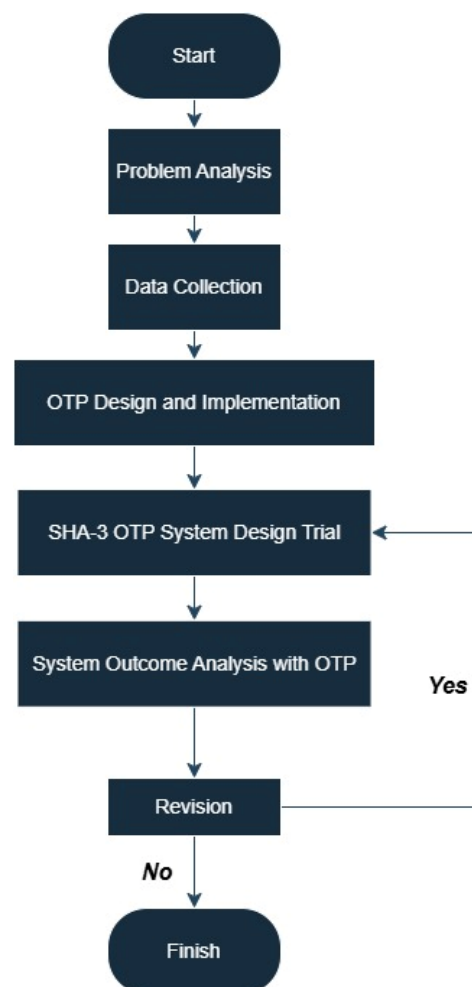


Figure 1. Research Stage

3.1 Problem Analysis

At this stage, the issues outlined in the background have been addressed, and the researcher has explained the underlying problems and formulated them as a foundation for system development.

3.2 Data Collection

At this stage, the researcher gathers data relevant to the study, such as collecting information related to one-time Passwords from previous research. This process helps facilitate the completion of the current research.

3.3 OTP Design and Implementation

In this section, the researcher designs a Data Flow Diagram for the OTP website, modeling the flow of data that occurs during the two-factor authentication process. More complete DFD Level 0 can be seen in Figure 2.



Figure 2. DFD Level 0

The process in DFD Level 0 works as follows: when a user requests an OTP code from the system, the user submits their username and password to the authentication system. The authentication system then generates an OTP code and an OTP expiration time, which are stored in the user table. Afterward, the OTP code and its expiration time are returned to the authentication system. The authentication system then sends the OTP code to the user, while the OTP expiration time is stored within the system. Figure 3 is the Level 1 Data Flow Diagram.

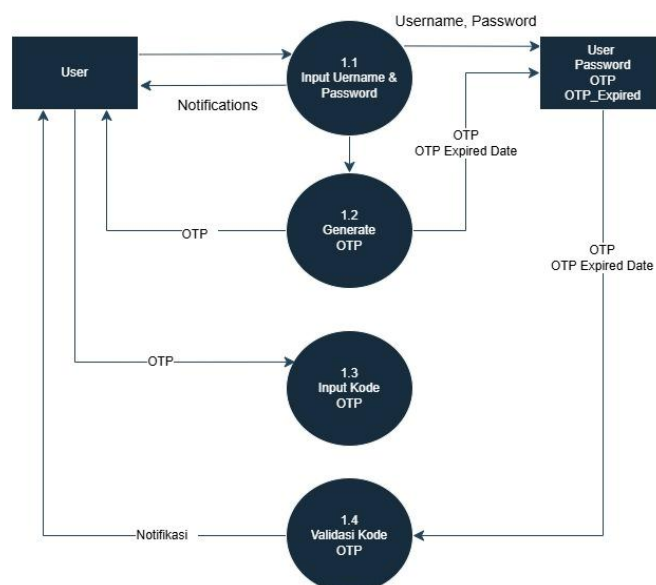


Figure 3. Data Flow Diagram Level 1

The following explains the workflow of DFD Level 1: First, the user inputs their username and password, which are then checked against the user table. If a matching user record is found, the system generates an OTP code along with an expiration time. If no matching data is found, the system notifies the user. The generated OTP and its expiration time are stored in the user table and also sent to the user. The OTP and its expiration time are saved in the user table to enable OTP validation. The user inputs the received OTP, which the system validates by checking both the OTP code and whether it has expired. If the OTP is correct and still valid, the user is granted access to the website. If the OTP is incorrect or expired, the user will receive a notification.

3.4 Flowchart

At this stage, the researcher designs a flowchart that explains the system's workflow, specifically for the "Implementation of One Time Password with the SHA-3 Algorithm on the inventory website of Universitas Negeri Makassar." Figure 4 is the Login Menu Flowchart.

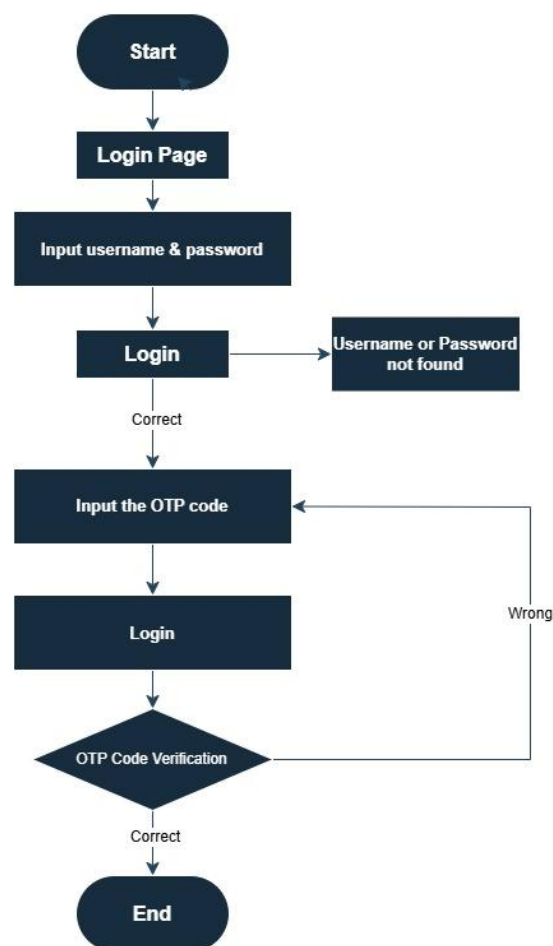


Figure 1. Flowchart Login Menu

The flowchart above illustrates the login process on the website. To access the main page, users must enter their username and password. The system then verifies whether the provided credentials match the data stored in the database. If the information is correct, the user is directed to the OTP code submission page. However, if the credentials are incorrect, an error message is displayed, and the user is redirected back to the login page. Once the correct OTP code is entered, the user successfully gains access to the

website. If the OTP code is incorrect, the user must re-enter the code. Figure 5 is the OTP Code Generation Flowchart.

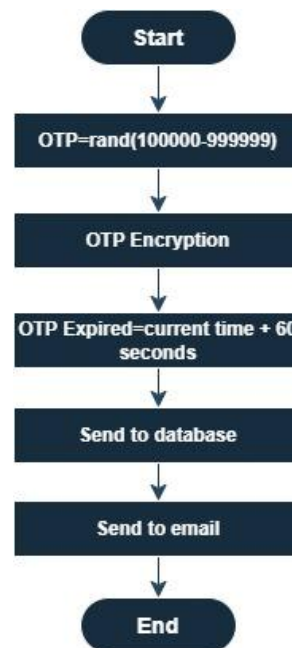


Figure 2. OTP Code Generation Flowchart

The flowchart above illustrates the process of generating the OTP code. During the OTP creation, the system generates a random number between 100000 and 999999. This random number is then encrypted using the SHA-3 algorithm. To determine the OTP expiration time, the system checks the current time and adds 60 seconds. Both the OTP code and its expiration time are then stored in the database and sent to the user's email. This explanation describes the process of creating the OTP code. Figure 6 is the OTP Verification Flowchart.

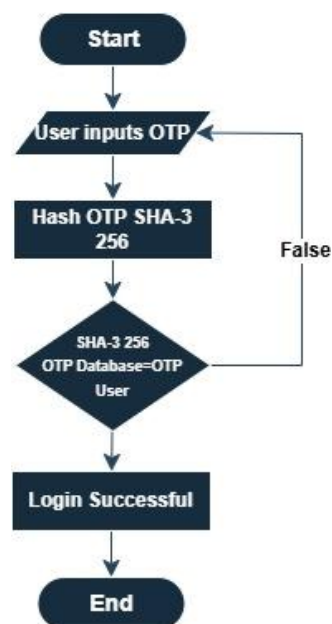


Figure 3. OTP Verification Flowchart

The workflow described in the flowchart is as follows: First, the user enters the OTP code received in their email. The entered code is then hashed using the SHA-3 256

method. This hashed code is compared with the OTP hash stored in the user database. If the hashed code matches the one in the database, the OTP is correct, and the user will successfully log in. However, if the hashes do not match, the code is incorrect, and the user will receive a notification.

3.5 Database Design

Moreover, the researcher created a table which is a design of the database that will be used.

```
table_user
-----
Id int (11)
username varchar (256)
password varchar (256)
Email varchar (256)
OTP varchar (256)
OTP Expired datetime
```

3.6 User Interface

At this stage, the researcher designs the User Interface, which involves creating the layout and visual design of the website to ensure users can easily access the site and use the One Time Password feature. Figure 7 is the design of the Login Menu. While Figure 8 is the OTP Code Delivery Screen Design.

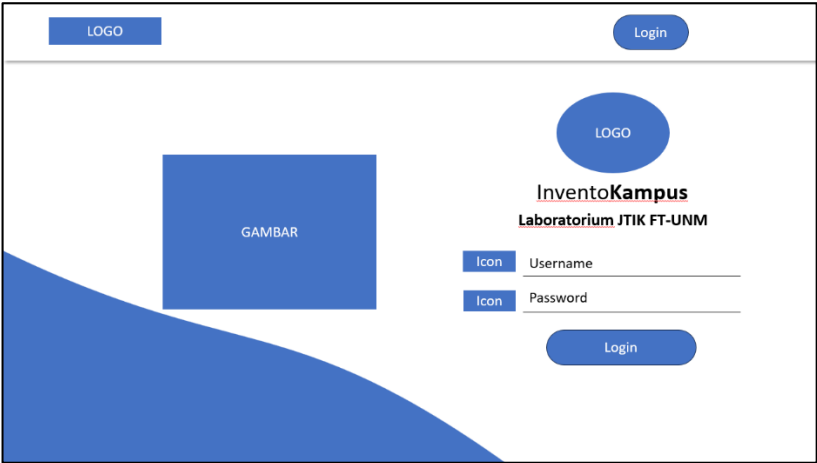


Figure 7. Login Menu Screen Design

Figure 8. OTP Code Delivery Screen Design

The system interface consists of several key fields and buttons. The Username field is used to input the user's stored username, while the Password field allows users to enter their saved password. The OTP Code field is provided for users to input the received One-Time Password, ensuring an additional layer of security. The Send button enables the system to dispatch the OTP code to the user's registered phone number via SMS. Finally, the Login button verifies the username, password, and OTP; if all credentials are correct, the user is granted access to the system, otherwise, a login failure notification appears.

3.7 Testing the Design Results with OTP Implementation

Table 1. System Validation

No	Received OTP code	OTP code entered	Validasi status	Description
1	OTP code received1	OTP code entered1	Success/Failure	Description1
2	OTP code received2	OTP code entered2	Success/Failure	Description2
3	OTP code received3	OTP code entered3	Success/Failure	Description3
4	OTP code received4	OTP code entered4	Success/Failure	Description4
5	OTP code received5	OTP code entered5	Success/Failure	Description5
6	OTP code received6	OTP code entered6	Success/Failure	Description6

The researcher tested OTP validation using a simple format. The first column records the OTP sent by the system to the user. The second column captures the OTP entered by the user, which is compared with the received OTP. The third column shows the validation result: "Success" if the codes match, or "Failure" if they do not. The fourth column, "Remarks," provides additional notes, such as "Code mismatch," "Code expired," or "Validation successful." Table 1 shows the system validation.

Example:

```
Received OTP: 293940 | Entered OTP: 293940 | Status: Success | Remarks:
Valid code.
Received OTP: 938475 | Entered OTP: 768594 | Status: Failure | Remarks:
Code mismatch.
```

After validation testing, a brute force attack simulation will be conducted on the OTP and password systems. This test aims to measure the security strength and how long it would take for a breach to occur.

3.8 Attack Testing

Table 2. OTP Brute Force Attack Testing

No	OTP Code	OTP Cooldown Time	Payload	Result
1	Code OTP 1	1 Minutes	Payload 1	Success/Failure
2	Code OTP 2	2 Minutes	Payload 2	Success/Failure
3	Code OTP 3	3 Minutes	Payload 3	Success/Failure
4	Code OTP 4	4 Minutes	Payload 4	Success/Failure
5	Code OTP 5	5 Minutes	Payload 5	Success/Failure
6	Code OTP 6	6 Minutes	Payload 6	Success/Failure
7	Code OTP 7	7 Minutes	Payload 7	Success/Failure
8	Code OTP 8	8 Minutes	Payload 8	Success/Failure
9	Code OTP 9	9 Minutes	Payload 9	Success/Failure
10	Code OTP 10	10 Minutes	Payload 10	Success/Failure

This study uses a brute-force attack scenario to test OTP security by measuring how many attempts are needed to guess the correct code. A brute force attack systematically tries character combinations, often using ASCII sets, and can be performed remotely by attackers (Stiawan et al., 2019). The goal is to see how easily an OTP can be stolen through brute force, with the attacker's final aim being to disrupt web services and extract sensitive data (Hossain et al., 2020). Table 2 shows the OTP Brute Force Attack Testing in detail.

For testing, the researcher used Burp Suite, a tool for performing security assessments through brute force attacks. Other tools like Hydra and Cain and Abel were also referenced, which are capable of brute forcing various authentication protocols, including FTP, HTTP, and POP3. A table was prepared to record the results of brute force attacks on both OTPs and user passwords.

The table includes several columns: the first lists the targeted OTP codes; the second shows the system's cooldown time (1 to 10 minutes) before OTP expiration; the third details the payload, covering all combinations from "000000" to "999999." The last column records the outcome—success if the OTP was cracked before expiration, or failure if not. The researcher aims to find the optimal cooldown time that prevents successful brute-force attacks. Table 3 shows the User Password Brute Force Attack Testing Table in detail.

Table 3. Brute Force Attack Testing Table Password User

No	Password	Status Code	Results	Description
1	Password 1	1 Minutes	Success/Failure	Weak/Complicated Passwords
2	Password 2	2 Minutes	Success/Failure	Weak/Complicated Passwords
3	Password 3	3 Minutes	Success/Failure	Weak/Complicated Passwords
4	Password 4	4 Minutes	Success/Failure	Weak/Complicated Passwords
5	Password 5	5 Minutes	Success/Failure	Weak/Complicated Passwords
6	Password 6	6 Minutes	Success/Failure	Weak/Complicated Passwords

In the next table format, the researcher tests brute force attacks on user passwords. The first column lists the passwords used in each attempt, categorized into weak passwords (e.g., "123456", "password", "abc123") and complex passwords (e.g., "P@ssw0rd123!"). The second column shows the HTTP server response: a "200" code indicates a successful login (valid password), while "401" means login failure (invalid password). The third column records the result, either success if the password is correct, or failure if it is incorrect. The final column provides notes on password strength, identifying whether it is weak (easy to guess) or strong (complex and harder to breach).

The purpose of brute force testing is to evaluate the system’s resistance against repeated login attempts using various passwords. It also measures the strength of user passwords in protecting their accounts from brute-force attacks. Additionally, it checks if the server correctly responds to login attempts by issuing appropriate status codes ("401" for failure, "200" for success). For example, in one test:

Row 1: Password "123456" → Status Code 200 → Result: Failure → Note: Weak Password.
Row 2: Password "Xt5@Er3!" → Status Code 401 → Result: Failure → Note: Strong Password.

In the final phase, an overall analysis of the OTP system is conducted to confirm whether the OTP functionality works properly and ensures secure access to the main website.

4. Result and Discussion

This research develops One Time Password by implementing encryption technology and creating a Website using HTML, CSS, Java, and PHP. The research steps include Problem analysis, data collection, OTP design and implementation, OTP SHA-3 system design trial, and analysis of the final results of the system with OTP.

4.1 Software Preparation

Visual Studio Code is used for website development due to its wide range of extensions that support multiple programming languages like HTML, CSS, PHP, and JavaScript. This flexibility makes it an ideal tool for building websites. For testing and implementation, XAMPP was utilized as a local server environment. It integrates essential components such as Apache, MySQL, and PHP, allowing the application to run locally and speeding up the development process.

In the One-Time Password (OTP) system, SHA-3 hashing ensures secure OTP generation. These OTPs are temporarily stored in a MySQL database for validation, with encryption handled by PHP scripts. XAMPP plays a crucial role in connecting the application, server, and database during the system's development and testing phases. The front-end part of the application includes the visual layout, design, and user-

interactive functionalities. Figure 9 is the design of the Login Page, while Figure 10 is the design of the OTP Sending Page.

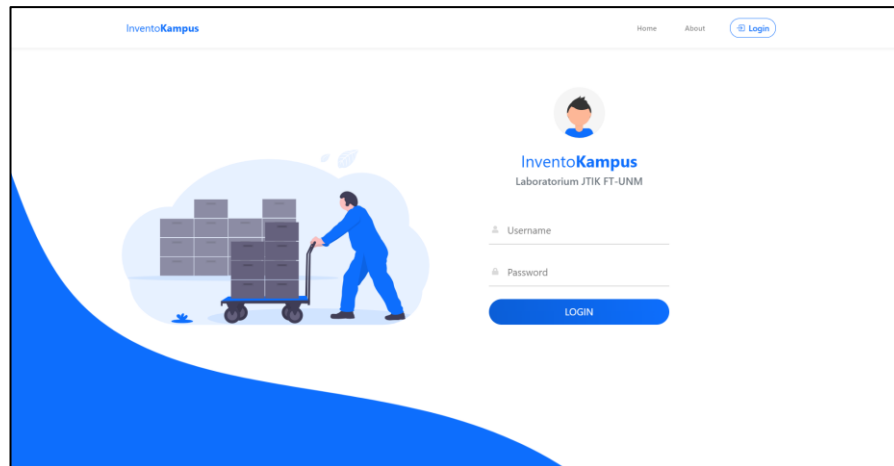


Figure 9. Login Page

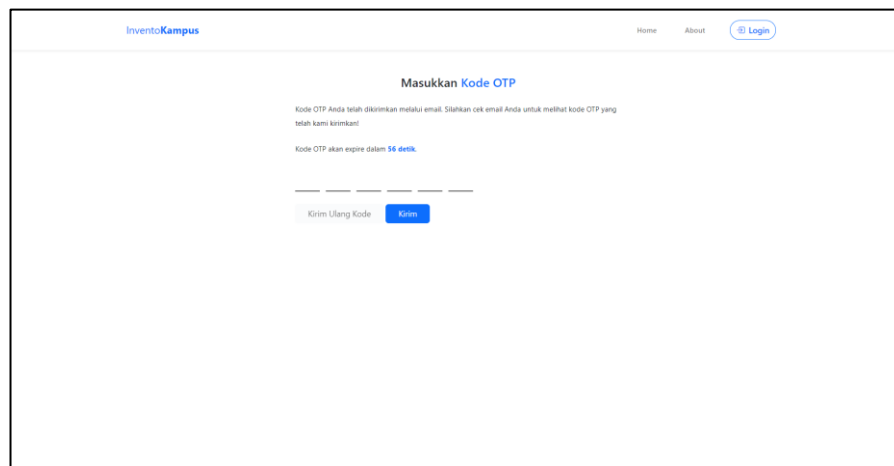


Figure 4. OTP Sending Page

The back end refers to the part of a website that operates behind the scenes and is not directly visible to users. It is responsible for ensuring that the website functions properly by handling various processes in the background. In this system, the back end manages tasks such as generating OTP codes, setting the validity period of each OTP, updating records and marking expired OTPs in the database, establishing expiration time indicators, and managing input fields on the OTP submission page. The following scripts are Generate OTP and Time Expiration.

```
$otp = rand(100000, 999999);
$otp_hash = hash('sha3-256', $otp);
$otp_expired = date('Y-m-d H:i:s', time() + 60);
```

The code generates a 6-digit One Time Password (OTP) using `rand(100000, 999999)`, hashes it with the SHA3-256 algorithm via `hash('sha3-256', $otp)` for secure storage, and then sets its expiration to one minute from now by adding 60 seconds to the current time with `time()`, ensuring the OTP is valid for only 60 seconds. The following script is a

program to update the expired OTP data in the database and set the expiration time indication.

```
mysql_query($conn, "UPDATE table_user SET otp = '$otp_hash',
otp_expired = '$otp_expired' WHERE username = '$username'");
$result = fetch("SELECT * FROM table_user WHERE username =
'$username' LIMIT 1")[0];
$otp_expire = strtotime($result['otp_expired']);
$current_time = time();
$sisawaktu = $otp_expire - $current_time;
$sisawaktu = $sisawaktu < 0 ? 0 : $sisawaktu;
```

The code first updates the user table by setting the OTP column to the hashed OTP value and the otp_expired column to the expiration timestamp for the specified \$username. It then retrieves that user's record with a SELECT query and converts the returned expiration string into a UNIX timestamp using strtotime(). By subtracting the current time (time()) from this expiration timestamp, it computes the remaining time (\$sisawaktu). Finally, it ensures that if this difference is negative—meaning the OTP has already expired—\$sisawaktu is reset to 0 so that it never goes below zero. Next is the OTP Input validation and redirection script.

```
if (isset($_POST['submit'])) {
    $otp_input = "";
    for ($j = 0; $j <= 5; $j++) {
        $otp_input .= $_POST[$j];
    }
    $otp = $result['otp'];
    $otp_input = str_replace(' ', '', $otp_input);
    $otp_input = hash('sha3-256', $otp_input);

    if ($otp === $otp_input && $sisawaktu > 0) {
        $_SESSION['login'] = $username;
        echo
        "<script>window.location.replace('$BASE_URL/admin')</scri
        pt>";
    } else {
        $error = "Kode OTP yang Anda masukkan salah!";
    }
}
```

The code above is executed when the "submit" button is clicked: first, the variable \$otp_input is initialized as an empty string, then a for loop combines the values from the six OTP input fields (\$_POST[0] to \$_POST[5]) into a single string. This string is then cleaned of any spaces using str_replace and hashed using the SHA3-256 algorithm. The resulting hash is compared with the stored OTP value (\$result['otp']), and it also checks if there is any remaining time (sisawaktu, the expiration time of the OTP) that is still positive. If both conditions are satisfied, the login session is set (\$_SESSION['login']), and the user is redirected to the dashboard page (\$BASE_URL/admin). If not, the \$error variable will be populated with the message "The OTP you entered is incorrect!" to be displayed on the page.

4.2 OTP Validation Testing

This study tests the OTP code to validate whether it can be confirmed, allowing users to access the website dashboard. If the OTP code is incorrect, an error message will be displayed. Table 4 shows the OTP Code Validation Trial.

Table 4. OTP Code Validation Trial

No	OTP time received	Received OTP code	OTP code entered	OTP input time	Status validation	Description
1	15.19	900353	900353	15.19	Successful	-
2	15.24	591180	310191	15.24	Failed	Incorrect OTP
3	16.06	603461	201922	16.06	Failed	Incorrect OTP
4	16.15	904646	904646	16.16	Failed	OTP Expired
5	17.2	452936	452936	17.2	Successful	-
6	17.45	357027	698699	17.45	Failed	Incorrect OTP

The OTP validation data shows the time of receipt, the received OTP code, the entered OTP code, input time, validation status, and remarks. Data analysis indicates successful validation when the code and time are correct, and failure when the code is incorrect or expired. For example, row 1 is valid, rows 2 and 3 fail due to incorrect codes, row 4 fails due to expiration, and row 5 is valid.

4.3 Testing Trial attacks

In this test, the researcher will conduct an Attack Trial on the login page and the OTP submission page of the website using the Brute Force Attack method. The application used for the trial is Burp Suite Professional. Brute Force Attack Test Table for OTP in detail.

Table 5. Brute Force Attack Test Table for OTP

No	OTP code	OTP Cooldown Time	Payload	Result
1	598011	1 Minute	500000	Failed
2	608992	2 Minute	600000	Successful
3	360111	3 Minute	300000	Failed
4	926195	4 Minute	900000	Successful
5	248422	5 Minute	200000	Successful
6	783331	6 Minute	700000	Failed
7	339827	7 Minute	300000	Successful
8	494227	8 Minute	400000	Successful
9	575231	9 Minute	500000	Successful
10	231299	10 Minute	200000	Successful

A Brute Force OTP attack test was conducted to determine a safe cooldown period. Results showed that shorter cooldown times (1–3 minutes) effectively prevented attacks, while longer cooldown times (4–10 minutes) made the system increasingly vulnerable, even against smaller payloads. Table 6 is the Brute Force Attack Test Table for passwords in detail.

Table 6. Brute Force Attack Test Table for password

No	Password	Status Code	Results	Description
1	kakukaku	200	Berhasil	Weak Passwords
2	c@mpB3ll	401	Gagal	Complex passwords
3	qwerty	200	Berhasil	Weak Passwords

No	Password	Status Code	Results	Description
4	P@s5w0rd	401	Gagal	Complex passwords
5	0ctOpuS	401	Gagal	Complex passwords
6	pass123	200	Berhasil	Weak Passwords

This data comes from a Brute Force Password attack test aimed at finding what makes a password secure. The results showed that short and simple passwords without a mix of uppercase letters, numbers, and special characters were easy to hack. Examples like "kakukaku," "qwerty," and "pass123" were quickly broken. On the other hand, more complex passwords such as "c@mpB3ll," "P@5sw0rd," and "0ctOpuS," which used a combination of letters, numbers, and symbols, successfully resisted the attacks and proved much stronger.

5. Conclusion

Based on the research findings, several conclusions can be drawn. First, the OTP delivery system on the Inventory Lab website of the Department of Informatics and Computer Engineering successfully sends OTP codes to the registered user's email. The OTP code is not sent if the username or password is incorrect. Additionally, both passwords and OTP codes are encrypted using the SHA-3 method, ensuring that they appear random in the database and remain unreadable. Second, validation tests confirm that an OTP code is considered correct if it matches the one sent via email; however, if an incorrect OTP is entered or if the correct OTP is used after its expiration time, authentication fails. Third, in brute force attack simulations on OTP codes, three attempts successfully retrieved the OTP while seven failed. The successful breaches were due to an extended countdown timer or OTP codes being too predictable. To enhance security against brute force attacks, an expiration time of one minute is recommended for OTP codes. Lastly, brute force attacks on passwords yielded mixed results, with three successful and three failed attempts. Weak passwords lacking character combinations were more vulnerable, whereas passwords containing a mix of uppercase and lowercase letters, numbers, and symbols demonstrated higher resistance against brute force attacks.

Acknowledgments: Thanks to all colleagues and lecturers at the Department of Computer Engineering, Makassar State University, South Sulawesi, Indonesia, so that this research can be completed well, and there is still a need for future updates, and hopefully can be cited by many researchers in the same field.

Author contributions: The authors are responsible for building Conceptualization, Methodology, analysis, investigation, data curation, writing—original draft preparation, writing—review and editing, visualization, supervision of project administration, funding acquisition, and have read and agreed to the published version of the manuscript.

Funding: The study was conducted without any financial support from external sources.

Availability of data and Materials: All data are available from the authors.

Conflicts of Interest: The authors declare no conflict of interest.

Additional Information: No Additional Information from the authors.

References

- [1] Advances In Cryptology - Eurocrypt 2013: 32nd Annual International Conference On The Theory And Applications Of Cryptographic Techniques, Athens, Greece, May 26-30, 2013. Proceedings (1st Edition). (2013). Springer.
- [2] Afiifah, K., Azzahra, Z. F., & Anggoro, A. D. (2022). Analisis Teknik Entity-Relationship Diagram Dalam Perancangan Database Sebuah Literature Review. *Intech*, 3(1), 8–11. <https://doi.org/10.54895/Intech.V3i1.1261>

- [3] Andi Rosano, Nur Ali Farabi, & Aliffah Kusumaningrum. (2018). Perancangan Sistem Internet Banking (Ibank) Menggunakan One-Time-Password (Otp) Untuk Pengaman Transaksi (Studi Kasus Bank Mega Tbk). 3.
- [4] Ciputra, R. D. A. (2017). Implementasi One Time Password Mobile Token Dengan Algoritma Secure Hash Algorithm 1 (Sha1) Pada Login Website Pusdakrimti Kejaksaan Agung Republik Indonesia.
- [5] Del Sole, A. (2021). Visual Studio Code Distilled: Evolved Code Editing For Windows, MacOS, And Linux. Apress. <https://doi.org/10.1007/978-1-4842-6901-5>
- [6] Documentation For Visual Studio Code. (T.T.). Diambil 27 April 2024, Dari <https://code.visualstudio.com/docs>
- [7] Dworkin, M. J. (2015). Sha-3 Standard: Permutation-Based Hash And Extendable-Output Functions (Nist Fips 202; Hlm. Nist Fips 202). National Institute Of Standards And Technology. <https://doi.org/10.6028/Nist.Fips.202>
- [8] Haerulah, E., & Ismiyati, S. (2017). Aplikasi E-Commerce Penjualan Souvenir Pernikahan Pada Toko "Xyz." 4(1).
- [9] Hapsari, N. S., Fatman, Y., & Isbandi, I. (2020). Implementasi Metode One Time Password Pada Sistem Pemesanan Online. Jurnal Media Informatika Budidarma, 4(4), Article 4. <https://doi.org/10.30865/Mib.V4i4.2195>
- [10] Hasugian, P. S. (2018). Perancangan Website Sebagai Media Promosi Dan Informasi. 3(1).
- [11] Herring, S. C., Stein, D., Virtanen, T., & Bublitz, W. (Ed.). (2013). Pragmatics Of Computer-Mediated Communication. De Gruyter Mouton.
- [12] Hossain, M. D., Ochiai, H., Doudou, F., & Kadobayashi, Y. (2020). Ssh And Ftp Brute-Force Attacks Detection In Computer Networks: Lstm And Machine Learning Approaches. 2020 5th International Conference On Computer And Communication Systems (Icccs), 491–497. <https://doi.org/10.1109/Icccs49078.2020.9118459>
- [13] Huang, Y., Huang, Z., Zhao, H., & Lai, X. (2013). A New One-Time Password Method. Ieri Procedia, 4, 32–37. <https://doi.org/10.1016/j.ieri.2013.11.006>
- [14] Lase, H. & Mufti. (2018). Implementasi One Time Password (Otp) Mobile Token Dengan Menggunakan Metode Algoritma Md5 Dan Sha. 1(1).
- [15] Li, Q., & Chen, Y.-L. (2009). Data Flow Diagram. Dalam Q. Li & Y.-L. Chen, Modeling And Analysis Of Enterprise And Information Systems (Hlm. 85–97). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-89556-5_4
- [16] Maulana, G. G. (2017). Pembelajaran Dasar Algoritma Dan Pemrograman Menggunakan El-Goritma Berbasis Web. Jurnal Teknik Mesin, 6(2), 8. <https://doi.org/10.22441/jtm.V6i2.1183>
- [17] Ni Putu Ana Rainita, Anak Agung Istri Callysta Athalia, Made Diva Putera Ananta, I Ketut Pratista Tri Pramana, Gede Arna Jude Saskara, & I Made Edy Listartha. (2023). Analisis Perbandingan Vulnerability Scanning Pada Website Dvwa Menggunakan Owasp Nikto Dan Burpsuite. Jurnal Informatika Dan Teknologi Komputer (Jitek), 3(2), 89–97. <https://doi.org/10.55606/jitek.V3i2.908>
- [18] Rizki, & Mulyati, S. (2020). Implementasi One Time Password Menggunakan Algoritma Sha-512 Pada Aplikasi Penagihan Hutang Pt. Xht. Edumatic : Jurnal Pendidikan Informatika, 4(1), 111–120. <https://doi.org/10.29408/Edumatic.V4i1.2158>
- [19] Rosaly, R., & Prasetyo, A. (2019). Pengertian Flowchart Beserta Fungsi Dan Simbol-Simbol Flowchart Yang Paling Umum Digunakan.
- [20] Schneier, B. (2015). Applied Cryptography: Protocols, Algorithms, And Source Code In C (Second Edition, 20th Anniversary Edition). John Wiley & Sons.
- [21] Seta, H., Wati, T., & Kusuma, I. C. (2019). Implement Time Based One Time Password And Secure Hash Algorithm 1 For Security Of Website Login Authentication. 2019 International Conference On Informatics, Multimedia, Cyber And Information System (Icimcis), 115–120. <https://doi.org/10.1109/Icimcis48181.2019.8985196>
- [22] Soleh, M. Y. (2010). Studi Dan Implementasi Algoritma Keccak.
- [23] Stiawan, D., Idris, Mohd. Y., Malik, R. F., Nurmaini, S., Alsharif, N., & Budiarto, R. (2019). Investigating Brute Force Attack Patterns In Iot Network. Journal Of Electrical And Computer Engineering, 2019, 1–13. <https://doi.org/10.1155/2019/4568368>
- [24] Tuasamu, Z., Lewaru, N. A. I. M., Idris, M. R., Syafaat, A. B. N., Faradilla, F., Fadlan, M., Nadiva, P., & Efendi, R. (2023). Analisis Sistem Informasi Akuntansi Siklus Pendapatan Menggunakan Dfd Dan Flowchart Pada Bisnis Porobico. Jurnal Bisnis Dan Manajemen (Jurbisman), 1(2), 495–510.
- [25] Wahid, A., Bahar, M. M., Nurwahid, M. S., Putra, S. A., Parenreng, J. M., & Irmawati, I. (2021). Perancangan Sistem Informasi Manajemen Kepegawaian (Simpeg) Berbasis Web Pada Universitas Negeri Makassar. Journal Of Embedded Systems, Security And Intelligent Systems, 2(1), 1. <https://doi.org/10.26858/Jessi.V2i1.16056>
- [26] Wibawa, S., Suryanto, S., & Ningsih, R. (2024). Perlindungan Data Digital Dengan Time-Based One-Time Password (Totp). Insantek, 5(1), 30–36. <https://doi.org/10.31294/Insantek.V5i1.3495>