Jurnal Dinamika Informatika
Volume 12, No 1, September 2023
ISSN 1978-1660 : 70-78
ISSN *online* 2549-8517

Perancangan Otentikasi *One Time Password* menggunakan Kode Unik via Email

Muhammad Rizqy Ath-Thaariq[1)], **Erna Kumalasari Nurnawati** [2)]
Renna Yanwastika Ariyana[3)]
Program Studi Informatika, Fakultas Teknologi Informasi dan Bisnis, Institut Sains dan Teknologi AKPRIND, Yogyakarta

Email: mrizqy.thariq@gmail.com [1)], ernakumala@akprind.ac.id [2)], renna@akprind.ac.id [3)]
Corresponding author: ernakumala@akprind.ac.id

1.1 Intisari

Penggunaan *username* dan *password* sebagai syarat otentikasi pada suatu sistem yang sering digunakan berkali-kali dapat membuat keamanan akun pada sistem tersebut menjadi rentan dilakukan pencurian data. Salah satu metode untuk menghindari pencurian data otentikasi adalah teknik *One Time Password* (OTP) yakni menggunakan sebuah kunci yang bersifat sementara dan akan lenyap setelah beberapa saat ataupun digunakan. Pada penelitian ini, kode OTP yang dikirimkan akan dijadikan sebagai kunci OTP dan pengganti kata sandi untuk otentikasi ke sistem. Metode penelitian ini menggunakan metode *waterfall* yang memiliki tahapan berupa: a) Pengumpulan Data; b) Desain Algoritma dan Aplikasi; c) Pengkodean atau Implementasi; d) Pengujian; e) Deployment. Penelitian ini merumuskan bagaimana alur kerja aplikasi, *routing web service* dan rancangan basis data yang digunakan. Hasil dari pengujian pada penelitian ini berjalan dengan baik dan sesuai desain algoritma serta alur kerja aplikasi yang dirumuskan yang mana pengguna dapat melakukan otentikasi sistem menggunakan email dan kode OTP sebagai pengganti kata sandi. Kode OTP dapat dikirim oleh web service dan diterima oleh pengguna. Dengan penelitian ini diharapkan dapat memberikan kontribusi keamanan sistem dengan otentikasi menggunakan kode unik sebagai pengganti kata sandi yang diterima pengguna melalui email yang terdaftar pada basis data.
Kata Kunci: otentikasi, OTP, kode unik, email

1.2 Abstract

The use of usernames and passwords as authentication requirements on a system that is used repeatedly can make account security on the system vulnerable to data theft. One of the methods to avoid authentication data theft is the One Time Password (OTP) technique, which uses a key that is temporary and will disappear after a while or after use.. In this research, the OTP code sent will be used as an OTP key and password replacement to authenticate to the system. This research method uses waterfall which has stages such as: a) Data Collection; b) Algorithm and Application Design; c) Coding or Implementation; d) Testing; e) Deployment. This study presents how the application workflow, web service routing and database design are used.. The test results in this study went well and according to the algorithm design and the formulated application workflow where users can authenticate the system using email and OTP codes instead of passwords.. The OTP code can be sent by the web service and received by the user. This research is expected to contribute to system security by authentication using a unique code as a substitute for a password that users receive via e-mail registered in the database.
Keywords: *authentication, OTP, unique code, email*

1.3 PENDAHULUAN

Perkembangan teknologi yang sangat pesat sehingga tidak hanya menawarkan keuntungan dengan memudahkan masyarakat, tetapi memiliki kelemahan yakni memudahkan para oknum tidak bertanggung jawab untuk melakukan modus dan operasi kejahatan siber yang beragam [1]. Salah satu dari bentuk operasi kejahatan siber adalah situs web *phishing* atau penyadapan informasi

otentikasi yang umumnya berupa *username* dan *password* sehingga dari sisi keamanan dirasa gagal dalam mengamankan hak aksesnya [2, 3]. Jika suatu aplikasi yang memiliki borang otentikasi yang tidak berfungsi sesuai dengan harapannya, maka keamanan aplikasi tersebut menjadi berkurang yang dapat berakibat fatal terjadinya kebocoran hak akses akun *administrator* [4].

Penggunaan *username* dan *password* sebagai otentikasi terhadap suatu sistem yang sering digunakan berkali-kali, membuat keamanan akun pada sistem tersebut menjadi rentan dilakukan pencurian data [5, 6]. Untuk menghindari pencurian data melalui otentikasi, salah satunya adalah menggunakan metode otentikasi *One Time Password* (OTP) yang menggunakan sebuah kunci yang bersifat sementara dan akan lenyap setelah beberapa saat ataupun setelah digunakan[2]. Penggunaan OTP memiliki keuntungan yakni apabila terdapat penyusup yang hendak meretas akun, maka kode OTP yang telah diberikan tidak dapat digunakan Kembali [7]. Untuk menerapkan OTP pada sistem terdapat tiga jenis OTP yakni: *Single Factor Authentication* (SFA), *Two Factor Authentication* (2FA) dan *Multi Factor Authentication* (MFA) [8].

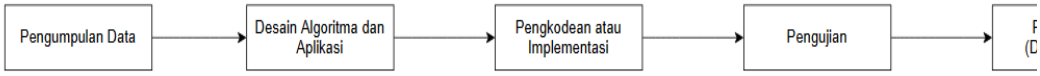
Penelitian sebelumnya menerangkan bahwa cara kerja dari OTP adalah kode OTP akan dikirimkan kepada pengguna melalui server gateway yang dilengkapi dengan algoritma enkripsi yang nantinya sebelum masuk ke dalam system, pengguna diminta untuk memasukkan kode OTP yang telah diberikan dan system akan melakukan validasi terhadap kode OTP tersebut[5]. Penelitian lain melakukan pengujian port SMTP 587 sebagai notifikasi email terhadap kode OTP yang dikirimkan dari server *Gateway* kepada pengguna. Dalam penelitiannya dijelaskan bahwa pada saat pengguna *login* ke dalam web, kode OTP akan di regenerate dan dienkripsi menggunakan algoritma MD5 yang kemudian di simpan dalam basis data. Setelah itu, kode OTP akan diambil dari basis data dan didekripsi yang menghasilkan sebuah *plaintext* yang nantinya *plaintext* tersebut dikirimkan ke email pengguna [8]. Penelitian lainnya menerangkan bahwa alur proses otentikasi OTP adalah pihak klien akan mengirimkan data kepada web server. Pada web server tersebut, data akan dienkripsi dengan algoritma DES yang selanjutnya web server akan melakukan pencocokan query ke basis data untuk mendapatkan data pengguna yang nantinya akan digunakan sebagai bahan untuk *generate* kode OTP. Kemudian, Kode OTP akan dikirimkan kepada klien sebagai *response* proses otentikasi. Kode OTP yang diterima klien akan disimpan dalam sebuah variable yang kemudian dikirim ke halaman verifikasi OTP menggunakan Intent[9].

Penelitian ini bertujuan untuk merancang sebuah sistem keamanan otentikasi kode OTP dengan menggunakan *Two Factor Authentication* (2FA). Otentikasi yang dilakukan pada penelitian ini hanya memasukkan email yang telah terdaftar pada basis data dan kode OTP akan dikirimkan ke email yang selanjutnya digunakan sebagai pengganti *password* untuk melakukan otentikasi ke dalam sistem.

1.4 METODOLOGI PENELITIAN

1. Alur Penelitian

Dalam penelitian ini menggunakan metode *Waterfall*, dikarenakan metode *waterfall* merupakan salah satu metode pengembangan aplikasi yang tradisional yang menekankan pada tahapan-tahapan yang berurutan dan sistematis. Metode *waterfall* secara harfiah adalah air terjun yang berarti bahwa tahapan-tahapan dilakukan secara berurutan dari atas ke bawah dan tidak boleh melakukan tahapan yang berbeda secara bersamaan [10]. Selanjutnya adalah tahapan-tahapan pada penelitian ini diilustrasikan pada Gambar 1.



Gambar 1. Metode Waterfall

a. Pengumpulan Data

Pengumpulan data dilakukan dengan cara: melakukan studi literatur pada jurnal ataupun *video* tutorial yang berkaitan dengan teknik otentikasi OTP; teknologi web yang digunakan meliputi bahasa pemrograman dan basis data; serta algoritma proses otentikasinya. Bahasa pemrograman yang digunakan adalah PHP versi 8.2 yang mana adalah bahasa skrip sisi

Jurnal Dinamika Informatika
Volume 12, No 1, September 2023
ISSN 1978-1660 : 70-78
ISSN *online* 2549-8517

server yang paling banyak digunakan dalam pengembangan aplikasi web karena menawarkan fleksibilitas, mudah digunakan serta mudah dipelajari. Bahasa ini juga memiliki fitur yang intuitif, terstruktur, eksekusi yang efisien, *open source*, dapat digunakan secara *cross-platform* dan mendukung SQL [11]. Sementara basis data yang digunakan adalah PostgreSQL versi 15.2 dikarenakan pada penelitian yang dilakukan oleh Fontaine Rafamantanantsoa (2018) menerangkan bahwa PostgreSQL untuk waktu respons rata-ratanya tidak berubah meskipun terjadi peningkatan jumlah koneksi klien dibandingkan MySQL yang menjadi lambat jika terjadi peningkatan jumlah koneksi klien. Selain itu, respons rata-rata server web apabila menggunakan PostgreSQL relatif stabil dibandingkan MySQL [12]. Dan *web server* yang digunakan adalah Nginx 1.24.0 dikarenakan bahwa Nginx lebih unggul daripada Apache dimana *bandwidth* yang dihasilkan jauh lebih baik sehingga dapat menampung data yang banyak [13].

b. Desain Algoritma dan Aplikasi

Pada tahap ini akan dilakukan perancangan desain algoritma otentikasi dengan menggunakan Unified Modelling Language (UML) Activity Diagram (Diagram Aktifitas) karena UML Diagram Aktifitas dapat memodelkan alur kerja sistem bisnis yang kompleks secara efektif [14] dan sesuai untuk memetakan alur kerja aplikasi pada penelitian ini. Desain algoritma alur kerja aplikasi menggunakan situs web app.diagram.net.

Selain menggambarkan perancangan desain algoritma, penggambaran perancangan basis data juga diperlukan pada penelitian ini dikarenakan untuk menyimpan informasi akun untuk proses otentikasinya.

c. Pengkodean atau Implementasi

Pengkodean dilakukan dengan menerapkan hasil rancangan algoritma dan basis data pada pengkodean dan tetap menjaga alur kerja aplikasi sesuai dengan pada tahap sebelumnya. Rancangan pada tahap sebelumnya diterapkan pada tahap ini berdasarkan analisis masalah dan tujuan penelitian.

d. Pengujian

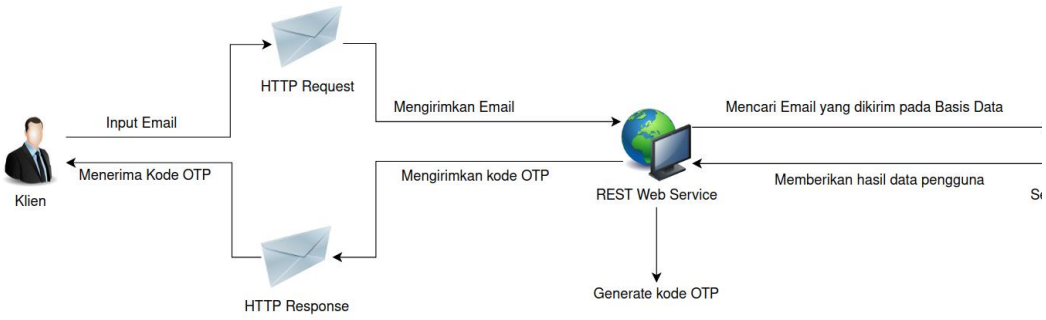
Tahap selanjutnya adalah pengujian setelah aplikasi selesai dibuat dan sesuai dengan tahapan-tahapan sebelumnya. Pengujian dilaksanakan dengan cara melakukan blackbox testing. Blackbox testing adalah teknik pengujian fungsional yang menguji sistem berdasarkan informasi dari spesifikasi yang diberikan. Dengan menggunakan blackbox testing pengujiannya tidak memiliki akses ke sumber kode, melainkan hanya berfokus pada luaran (output) yang dihasilkan sebagai respons terhadap masukan (input) yang dipilih dan kondisi eksekusi [15].

e. Deployment

Dan tahap terakhir adalah deployment yang mana hasil pengujian ini akan diterapkan pada studi kasus sesungguhnya.

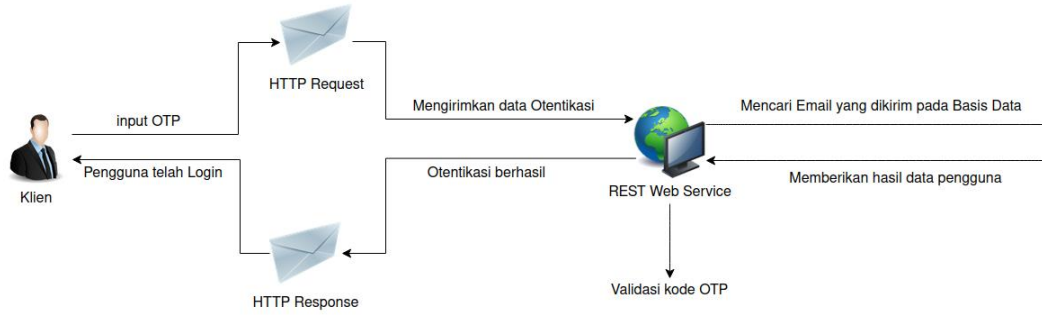
2. Rancangan Alur Kerja Aplikasi

Perancangan alur kerja aplikasi pada penelitian ini mengacu pada penelitian yang dilakukan oleh Anggit Prayogo (2018) dan memiliki perbedaan yakni pada penelitian Anggit Prayogo (2018) memiliki tahapan *enkripsi* dan *dekripsi* terhadap data otentikasi berupa: email; kata sandi; dan kode OTP pada saat dikirimkan [9], sementara pada penelitian ini tidak dilakukan *enkripsi* dan *dekripsi* pada data yang dikirimkan disebabkan pada penelitian ini data otentikasi berupa email dan kode OTP.



Gambar 2. Rancangan alur kerja proses generate kode OTP

Pada Gambar 2 diterangkan bahwa klien mengirimkan data berupa alamat email yang telah diisikan sebagai *HTTP Request* kepada *web service*. Selanjutnya *web service* akan melakukan pencocokan data dengan menjalankan *query* di basis data, mencari alamat email yang tersimpan di basis data yang cocok dengan yang dikirimkan oleh klien. Setelah data cocok, maka selanjutnya basis data menyatakan bahwa email terdaftar dan memberikan hasilnya kepada *web service*. Setelah data diterima oleh *web service*, maka akan dilakukan *generate* kode OTP. Setelah kode OTP berhasil di*generate*, kode OTP tersebut akan dikirimkan ke email yang telah dicocokkan pada *query* basis data sebelumnya sebagai *HTTP Response* dari *web service* dan *web service* akan menyimpan kode OTP tadi ke dalam sebuah variabel yang akan digunakan untuk proses validasi kode OTP. Dan terakhir, kode OTP diterima klien pada laman emailnya.



Gambar 3. Rancangan alur kerja proses otentikasi

Berdasarkan Gambar 3, Setelah kode OTP telah di*generate* maka selanjutnya klien akan mengisikan kode OTP dan mengirimkan data otentikasi ke *web service* sebagai *HTTP Request* yang akan diterima oleh *web service* dan diteruskan ke basis data untuk dilakukan proses *query* pencocokan data otentikasi dengan email yang tersimpan di basis data. Setelah proses *query* pencocokan berhasil selanjutnya adalah validasi kode OTP yang diterima klien dengan variabel *web service* yang menyimpan kode OTP dan setelah proses validasi telah berhasil maka klien telah berhasil *login* ke dalam sistem.

3. Rancangan *Routing Web Service*

Rancangan *web service* pada penelitian ini berisikan *route* akses yang akan digunakan untuk mengirimkan *HTTP Request* dan mendapatkan *HTTP Response* dari *web service* yang memiliki pembagian dan keterangannya seperti yang disajikan pada Tabel 1.

Tabel 1: Pembagian *Route Web Service*

No.	Metode HTTP	URL	Keterangan
1.	GET	/login.php	Mendapatkan tampilan antarmuka pengguna berupa borang otentikasi yang terdiri dari 4 elemen, yakni: kolom email, tombol submit OTP, kolom OTP dan tombol submit *login.*

Jurnal Dinamika Informatika
Volume 12, No 1, September 2023
ISSN 1978-1660 : 70-78
ISSN *online* 2549-8517

2.	POST	/kirim_otp.php	Mengirimkan isian email ke *web service* untuk dilakukan validasi pencocokan dengan data email di basis data, setelah itu melakukan *generate* kode OTP dan mengirimkannya ke email yang telah valid.
3.	POST	/proses_login.php	Mengirimkan data otentikasi berupa email dan kode OTP yang telah diisikan pada /login.php ke *web service* dan dilakukan validasi data otentikasi dengan data dari *web service* untuk menentukan apakah data otentikasi sesuai atau tidak.
4.	GET	/index.php	Halaman yang pertama kali tampil dan sebagai halaman yang menentukan apakah pengguna telah login atau belum.
5.	GET	/logout.php	Mengirimkan *request* ke *web service* untuk melakukan otentikasi.

4. Rancangan Basis Data

Rancangan basis data yang dibutuhkan pada penelitian ini hanya cukup memerlukan minimal tiga kolom seperti yang ditampilkan pada Tabel 2.

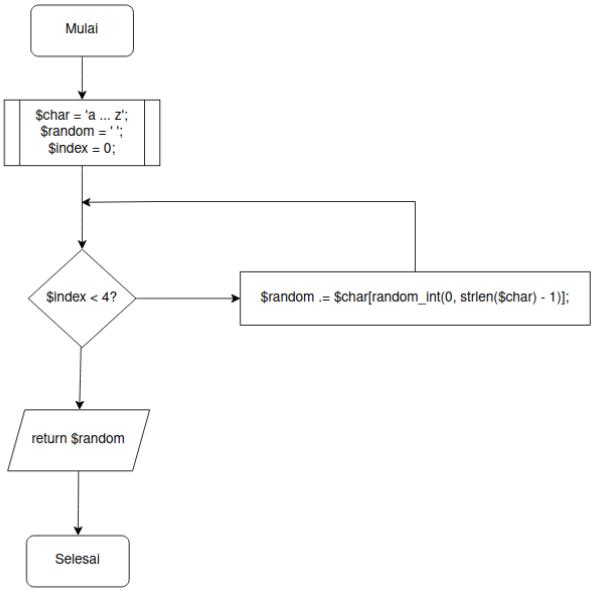
Tabel 2: Rancangan Basis Data

No	Nama Field	Tipe	Ukuran	Keterangan
1.	kode_pengguna	Char	5	PK
2.	nama_pengguna	Varchar	50	Not Null
3.	email_pengguna	Varchar	30	Unique

Pada penelitian ini tidak memerlukan kata sandi untuk proses otentikasi sehingga kolom untuk menyimpan kata sandi tidak diperlukan. Proses validasi pengguna cukup dilakukan menggunakan kode OTP yang dikirimkan ke email yang mana email tersebut telah disimpan di dalam basis data dan kode OTP disimpan pada sebuah variabel di *web service*.

1.5 HASIL DAN PEMBAHASAN

Pada saat klien atau pengguna mengakses halaman *login* maka akan dihadapkan dengan kolom email untuk memasukkan emailnya. Setelah itu, email yang diisikan akan dikirim ke *web service* untuk dicek validasinya apakah emailnya valid atau tidak, apabila tidak valid maka akan menampilkan pesan *error* dan apabila valid maka *web service* akan melakukan generate kode OTP yang mana kode tersebut dihasilkan menggunakan logika sederhana dengan mengacak urutan huruf abjad menggunakan perulangan *for* dan *function built-in* pada bahasa pemrograman PHP yang proses alur kerjanya, alur diagram mengacak kode OTP disajikan pada Gambar 4.



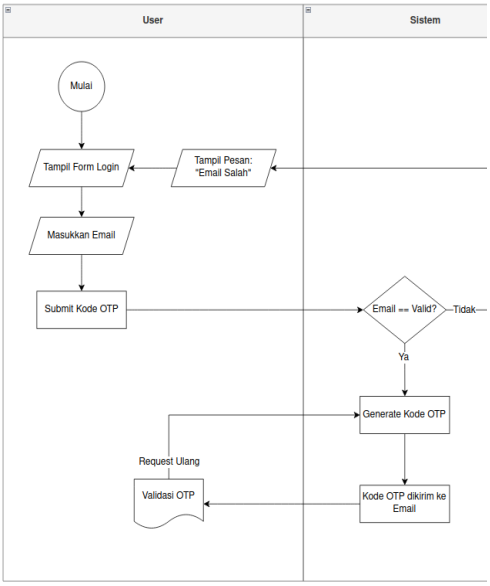
Gambar 4: Diagram alir mengacak kode OTP

Setelah meng*generate* kode OTP, maka kode OTP akan disimpan dalam sebuah variabel *session* pada *web service* yang akan digunakan untuk proses validasi OTP[9]. Dan kode OTP yang berhasil di*generate* akan dikirimkan ke email yang valid pada proses validasi email sebelumnya. Ilustrasi tahapan alur pengguna melakukan *generate* kode OTP disajikan pada Gambar 5.

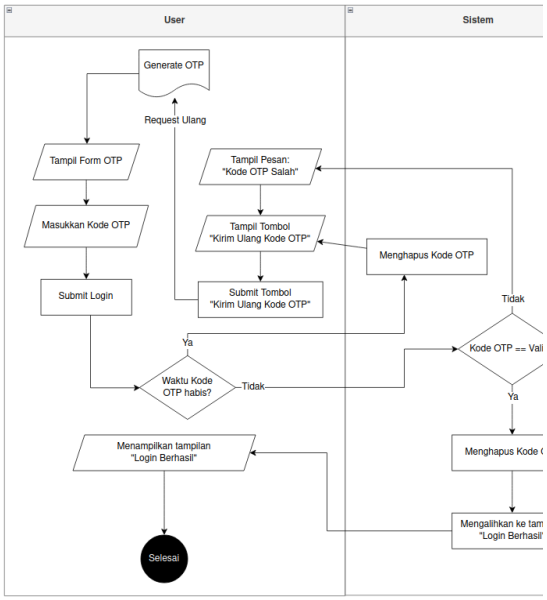
Setelah melakukan *generate* kode OTP, maka pada tampilan antar muka pengguna akan muncul kolom kode OTP dan pengguna mengisikan kode OTP yang telah diterima di laman emailnya. Selanjutnya adalah, kode OTP diisikan dan kemudian dikirimkan ke *web service* untuk diperiksa apakah masa berlaku kode OTP telah kadaluwarsa atau belum. Apabila kode OTP telah kadaluwarsa maka pada tampilan antar muka pengguna akan muncul tombol kirim ulang kode OTP dan melakukan proses "Request Ulang" serta menghapus variabel kode OTP yang tersimpan pada *web service*.

Apabila kode OTP tidak kadaluwarsa maka selanjutnya adalah proses validasi kode OTP apakah kode OTP yang dikirimkan sesuai dengan isi dari variabel *web service* yang menampung kode OTP pada saat proses *generate*. Apabila tidak valid maka akan muncul pesan 'kode OTP salah' pada tampilan antar muka pengguna, sementara apabila valid maka selanjutnya *web service* akan menghapus variabel yang berisikan kode OTP yang telah di*generate* pada proses sebelumnya dan selanjutnya melakukan pengalihan (*redirect*) ke halaman berhasil. Ilustrasi validasi OTP disajikan pada Gambar 6.

Jurnal Dinamika Informatika
Volume 12, No 1, September 2023
ISSN 1978-1660 : 70-78
ISSN *online* 2549-8517

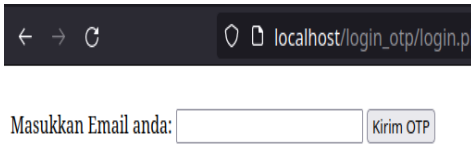


Gambar 5. Diagram Aktivitas Generate Kode OTP

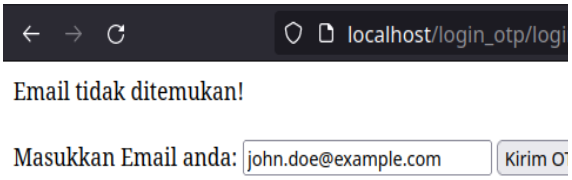


Gambar 6. Diagram Aktivitas Validasi Kode OTP

Pada Gambar 7 disajikan tampilan halaman awal pada saat klien hendak melakukan otentikasi ke dalam sistem, pada halaman awal login hanya terdapat satu kolom input yakni kolom input email dan satu buah tombol untuk mengirimkan *request generate* kode OTP. Apabila email tidak terdaftar di basis data akan ditolak (Gambar 8).



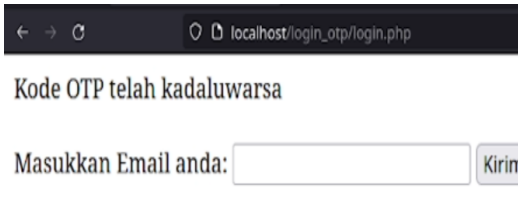
Gambar 7. Login untuk Request OTP



Gambar 8. Email tidak terdaftar di basis data

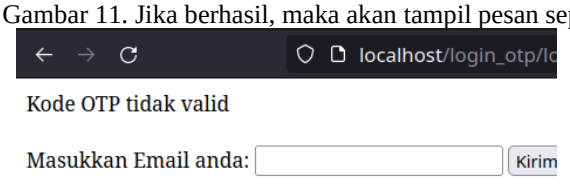
Pada Gambar 9, halaman login yang semula hanya terdapat kolom email dan tombol kirim OTP kini terdapat perubahan yakni tombol kirim OTP telah hilang. Kemudian, muncul satu buah kolom isian kode OTP dan tombol login yang akan mengakses URL /proses_login.php. Pada Gambar 10, di kotak masuk email pengguna akan menerima pesan yang berisikan kode OTP yang telah di*generate* dan sebagai kode *login* untuk melakukan otentikasi ke dalam sistem. Selain terdapat kode OTP, disertakan juga durasi kode OTP bertahan yakni selama 5 menit lamanya.

Gambar 9. Pengiriman OTP

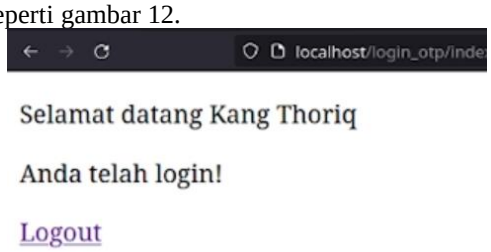


Gambar 10. Notofikasi Kode OTP telah Kadaluwarsa

Dan apabila pengguna mengisikan kode OTP yang tidak sesuai dengan yang diberikan, maka web server akan menghapus kode OTP sehingga kode OTP yang ada di email tidak dapat digunakan dan pengguna akan mengisikan kembali emailnya seperti pada tahap sebelumnya. Kejadian apabila kode OTP tidak valid atau tidak sesuai, web server akan mengalihkan pengguna ke halaman login dan menampilkan pesan "Kode OTP tidak valid" seperti yang ditampilkan pada Gambar 11. Jika berhasil, maka akan tampil pesan seperti gambar 12.



Gambar 11. Pengguna tidak bisa masuk sistem



Gambar 12. Pengguna berhasil masuk sistem

1.6

1.7 KESIMPULAN

Perancangan otentikasi OTP menggunakan kode unik via email yang dapat digunakan sebagai alternatif pengamanan untuk masuk ke suatu sistem karena pengguna hanya mendapatkan satu kali kode setiap akan memasuki sistem. Hal ini dapat diterapkan pada sistem informasi baik sistem berbasis desktop, sistem berbasis web maupun sistem berbasis perangkat bergerak. Penggunaan OTP via email juga menjamin pengguna harus mendaftarkan email kepada sistem, sehingga menambah keamanan dalam penggunaan sistem.

1.8 SARAN

Penelitian ini berfokus pada perancangan dan pengujian konsep otentikasi OTP menggunakan kode unik via email sehingga diperlukan penelitian lanjutan berupa tinjauan dari sisi keamanan *web server*, penerapan pada sistem aplikasi yang berjalan dan tanggapan dari pengguna terhadap konsep otentikasi OTP menggunakan kode unik via email ini.

1.9 DAFTAR PUSTAKA

[1] S. Bodhi and D. Tan, "Keamanan Data Pribadi Dalam Sistem Pembayaran E-Wallet Terhadap Ancaman Penipuan Dan Pengelabuan (Cybercrime)," *UNES Law Review*, vol. 4, no. 3, pp. 297–308, 2022, doi: 10.31933/unesrev.v4i3.236.

[2] N. I. Yahya and S. Amini, "Pengimplementasian One Time Password Dan Notifikasi Email Menggunakan Fungsi Hash SHA-512 Berbasis Web Pada SMK Cyber Media," *Skanika*, vol. 1, no. 2, pp. 745–750, 2018.

[3] M. Fida and A. Arokiaraj Jovith, "Anti-phishing strategy model for detection of phishing website in e-banking," *International Journal of Control Theory and Applications*, vol. 9, no. 16, pp. 7697–7702, 2016, doi: 10.19107/ijisc.2016.01.07.

[4] M. F. Londjo, "Implementasi White Box Testing Dengan Teknik Basis Path Pada Pengujian Form Login," *Jurnal Siliwaangi*, vol. 7, no. 2, pp. 35–40, 2021.

[5] D. E. Kurniawan, M. Iqbal, J. Friadi, F. Hidayat, and R. D. Permatasari, "Login Security Using One Time Password (OTP) Application with Encryption Algorithm Performance," *J Phys Conf Ser*, vol. 1783, no. 1, 2021, doi: 10.1088/1742-6596/1783/1/012041.

[6] A. Senol, G. Acar, M. Humbert, and F. Z. Borgesius, "Leaky Forms: A Study of Email and Password Exfiltration Before Form Submission," *Proceedings of the 31st USENIX Security Symposium, Security 2022*, pp. 1813–1830, 2022.

[7] A. Emam, "Additional Authentication and Authorization using Registered Email-ID for Cloud Computing," *International Journal of Soft Computing and …*, no. 2, pp. 110–113, 2013.

[8] D. Kurnia, U. Pembangunan, P. Budi, J. Jend, G. Subroto, and S. Sikambing, "Teknik One Time Password Dalam Pengamanan Page Login Website Dengan Notifikasi Email," *... Teknologi Informasi & Komunikasi Ke-7*, pp. 201–208, 2020.

[9] A. Prayogo and M. A. Rony, "Implementasi One Time Password pada Sistem Login dengan Algoritma SHA-256 dan DES pada Aplikasi EO Blucampus Berbasis Client Server," *Skanika*, vol. 1, no. 2, pp. 448–454, 2018.

[10] R. M. Firzatullah, "Development of XYZ University's Student Admission Site Using Waterfall Method," *Jurnal Mantik*, vol. 3, no. 2, pp. 10–19, 2019.

[11] M. Laaziri, K. Benmoussa, S. Khoulji, and M. L. Kerkeb, "A Comparative study of PHP frameworks performance," *Procedia Manuf*, vol. 32, pp. 864–871, 2019, doi: 10.1016/j.promfg.2019.02.295.

[12] F. Rafamantanantsoa and M. Laha, "Analysis and Neural Networks Modeling of Web Server Performances Using MySQL and PostgreSQL," *Communications and Network*, vol. 10, no. 04, pp. 142–151, 2018, doi: 10.4236/cn.2018.104012.

[13] Z. P. Putro and R. A. Supono, "Comparison Analysis of Apache and Nginx Webserver Load Balancing on Proxmox VE in Supporting Server Performance," vol. 7, no. 3, pp. 144–151, 2022.

[14] T. Ahmad, J. Iqbal, A. Ashraf, D. Truscan, and I. Porres, "Model-based testing using UML activity diagrams: A systematic mapping study," *Comput Sci Rev*, vol. 33, pp. 98–112, 2019, doi: 10.1016/j.cosrev.2019.07.001.

[15] S. Nidhra and J. Dondeti, "How to Write a L iterature R eview," *Project Management Journal*, vol. 2, no. 2, pp. 29–50, 2012.