Metode Absensi Mahasiswa Berbasis *QR Code* dan *Time-Based One-Time Password*

Aprianti Nanda Sari[1], Trisna Gelar Abdillah[2]

[1,2] Jurusan Teknik Komputer, Politeknik Negeri Bandung, Indonesia
[1]aprianti.nanda@polban.ac.id, [2]trisna.gelar@polban.ac.id

Abstrak

Sistem absensi manual dengan cara memanggil nama mahasiswa satu persatu oleh dosen atau dengan menandatangani list kehadiran tentunya akan memakan waktu yang cukup lama dan tidak aman. Maka dibangunlah sistem absensi terkomputerisasi berbasis sidik jari, RFID, pengenalan wajah dan lainnya. Akan tetapi metode-metode tersebut memerlukan biaya tambahan untuk membeli alat khusus. Sebagai solusi, akhirnya sistem absensi berbasis *QR code* semakin diminati karena hanya memerlukan *webcam* bawaan dari laptop atau PC, atau kamera *smartphone*. Mahasiswa yang hadir di kelas hanya butuh memindai *QR code* yang tertera pada saat kelas berlangsung melalui *smartphone* mereka sebagai tanda kehadiran. Akan tetapi sistem absensi berbasis *QR code* ini memiliki celah misalnya mahasiswa yang hadir dapat mengambil gambar dan mengirimkan *QR code* di kelas kepada temannya yang tidak hadir agar dapat dianggap hadir. Sehingga diperlukan sistem autentikasi dengan keamanan tambahan agar kecurangan pada sistem absensi berbasis *QR code* tidak terjadi. Salah satu upayanya adalah dengan menggunakan *One-Time Password* yang dibuat unik dan dinamis untuk setiap mahasiswa dan hanya berlaku sekali saja dalam waktu singkat saja atau yang lebih dikenal sebagai *Time-based One-Time Password* atau disingkat TOTP. Dari hasil analisa, penggunaan TOTP dapat menghindari berbagai kemungkinan kecurangan oleh mahasiswa dan penyerangan oleh pihak ketiga.

Kata kunci : sistem absensi mahasiswa, *QR code*, *one-time password*, *time-based one-time password*, TOTP

1. Pendahuluan

Sistem absensi merupakan hal penting dalam berlangsungnya kegiatan belajar mengajar di perguruan tinggi terutama bagi mahasiswa. Di sebagian perguruan tinggi, sistem absensi masih menggunakan tanda tangan. Hal ini tentunya tidak praktis karena rekap kehadiran mahasiswa harus dilakukan secara manual.Untuk itu beberapa sistem absensi terkomputerisasi dibangun dengan berbagai metode seperti sidik jari, RFID, pengenalan wajah (*face recognition*) dan masih banyak lainnya. Namun metode-metode tersebut memakan biaya tambahan karena memerlukan alat khusus.

Sebagai solusi untuk membangun sistem absensi yang minim biaya, beberapa sistem absensi yang sudah dibangun memanfaatkan *QR Code*. Selain tidak membutuhkan biaya karena bisa menggunakan kamera *smartphone* atau *webcam* sebagai pemindai, *QR Code* mudah diimplementasikan dan tidak membutuhkan sistem komputasi yang rumit. Beberapa sistem absensi berbasis *QR Code* yang sudah dibangun adalah oleh Nuddin, *et al* (2015), Ardhianto (2016), Aini, *et al* (2017), Hermanto, *et al* (2019), dan Pujastuti *et al* (2020). Akan tetapi sistem absensi berbasis *QR Code* ini juga masih memiliki celah. Misalnya, mahasiswa yang hadir di kelas dapat mengambil foto *QR Code* yang ada di kelas kemudian mengirimkannya ke mahasiswa lain yang tidak hadir sehingga dapat dianggap hadir. Untuk itu, diperlukan sistem absensi berbasis *QR Code* yang aman sehingga mengurangi kemungkinan terjadinya kecurangan dalam sistem absensi.

Salah satu cara untuk menghindari kecurangan tersebut adalah dengan penggunaan *password* dinamis atau yang lebih dikenal sebagai *One-Time Password* (OTP) ((Lamport, 1981), (Babkin *et al*, 2019), (Liao *et al*, 2009), dan (Malik *et al*, 2014)). Pada penelitian ini, OTP berbasis waktu (*Time-based One-Time Password*) atau yang dikenal sebagai TOTP digunakan karena *password* hanya berlaku pada rentang waktu yang singkat. Hal ini bertujuan agar mahasiswa tidak melakukan kecurangan seperti mengirimkan *QR Code* tidak memiliki waktu yang cukup.

2. Landasan Teori

Pada bab ini, teori dasar mengenai *QR Code*, TOTP dibahas secara singkat

2.1 QR *Code*

QR Code atau *Quick Response Code* merupakan bentuk dua dimensi dari barcode. *QR Code* pertama kali dikenalkan oleh Denso Wave pada tahun 1994 (Soon, 2008), (Hanks, 2012), (Singh *et al* 2016). Seiring berkembangnya teknologi, saat ini *QR Code* tidak hanya dikenali dan dibaca melalui pemindai

khusus tetapi kamera pada *smartphone*, bahkan *webcam*. Biasanya *QR Code* berisi informasi seperti teks, *link* URL, *geo-location*, nomor telepon, kartu nama, dan hal lainnya yang dapat disematkan. *QR Code* terlihat seperti sebuah kotak kecil berisi piksel berwarna hitam dan putih yang ditempatkan secara acak seperti pada Gambar 1.

Gambar 1 Contoh *QR Code*

2.2 Time-based One-Time Password

One-Time Password (OTP) merupakan serangkaian symbol atau angka yang dibuat sebagai *password* dan digunakan satu kali saja. Salah satu penggunaan OTP adalah untuk autentikasi pengguna, misalnya pada media sosial, *online banking*, maupun dompet digital.

Ada banyak algoritma untuk membangkitkan OTP, salah satunya adalah *Time-based One-Time Password* (TOTP). Menurut M'Raihi, *et.al* (2011), cara kerja TOTP adalah membuat *password* berdasarkan kunci dan waktu saat algoritma TOTP digunakan. Secara umum, TOTP memiliki skema yang sama dengan *HMAC-based One-Time Password* atau HOTP, namun yang membedakan adalah proses perhitungannya melibatkan kunci dan waktu sehingga TOTP dapat dinotasikan sebagai persamaan (1). Di mana K adalah kunci yang bersifat acak (*random*) dan T adalah waktu.

$$\text{TOTP} = \text{HOTP}\,(K, T) \qquad (1)$$

Nilai dari T dihitung berdasarkan persamaan (2). Di mana $T_{current}$ adalah waktu saat ini dalam satuan detik, T_0 adalah waktu inisiasi yang disepakati biasanya bernilai 0 ($T_0 = 0$). Sedangkan v adalah parameter yang menentukan berapa lama sebuah TOTP valid. Biasanya $v = 30$ sehingga TOTP hanya valid dalam waktu 30 detik.

$$T = \frac{T_{current} - T_0}{v} \qquad (2)$$

3. Metode Penelitian

Penelitian ini terdiri dari empat tahap utama yaitu studi literatur, perancangan fungsional sistem, pembuatan purwarupa, serta pengujian dan analisa hasil purwarupa. Studi literatur bertujuan untuk mengumpulkan sumber literasi dan menganalisa permasalahan yang ada pada sistem absensi berbasis *QR Code* yang sudah dibahas pada Bab 2. Pada tahap perancangan, bertujuan untuk membuat gambaran awal dari metode yang diajukan. Setelah dirancang, maka dilanjutkan dengan implementasi berupa pembuatan purwarupa yang kemudian diuji dan dianalisa.

Sistem absensi mahasiswa yang dibangun terdiri dari dua modul yaitu modul *server* dan modul *smartphone* yang memiliki fungsi berbeda. Modul *server* yang dibangun adalah berbasis *web* dan digunakan oleh staf administrasi kampus dan dosen untuk mengelola data mahasiswa, kelas, dan dosen. Selain itu, modul *server* juga berfungsi sebagai pembangkit TOTP, pemindai *QR Code*, serta merekap kehadiran mahasiswa. Modul *server* hanya dapat diakses oleh jaringan lokal, sehingga hanya staf administrasi kampus dan dosen yang dapat mengakses.

Modul *smartphone* diinstall dan digunakan oleh mahasiswa untuk mengecek jadwal kuliah, penampil *QR code* berisi identitas diri, serta penampil *QR Code* berisi TOTP yang berlaku sebagai tanda kehadiran.

Sistem ini terdiri dari dua prosedur utama yaitu prosedur registrasi dan prosedur verfikasi. Terdapat beberapa notasi yang digunakan pada tulisan ini seperti yang dijelaskan pada Tabel 1.

Tabel 1 Notasi persamaan yang digunakan

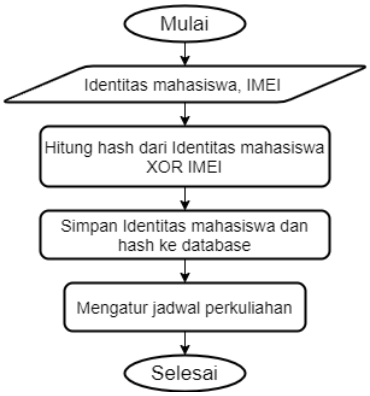
Notasi	Keterangan
$h(a)$	Fungsi *hash* satu arah dari parameter bernama a.
$E_{QR}(a)$	Fungsi untuk mengubah parameter a menjadi *QR Code*
$D_{QR}(a)$	Fungsi untuk mengubah QR code bernama a menjadi link *autofilled*
$a \oplus b$	Operasi XOR antara a dan b
$L(a)$	Membuat *link* sesuai parameter a
ID_s	ID dari sesi kuliah berjalan
NIM	Nomor Induk Mahasiswa
$IMEI$	*International Mobile Equipment Identity* dari *smartphone* mahasiswa

3.1 Prosedur Registrasi

Tujuan dari prosedur registrasi adalah mengumpulkan data mahasiswa beserta identitas *smartphone* yang digunakan seperti pada Gambar 2 dengan rincian sebagai berikut.

- Langkah 1: Modul *server* yang dioperasikan oleh staf administrasi kampus akan meminta dan menyimpan identitas mahasiswa dan identitas *smartphone* yang digunakan. Identitas ini haruslah unik seperti NIM dan IMEI (I*nternational Mobile Equipment Identity*).
- Langkah 2: Modul *server* akan menghitung nilai hash (x) dari NIM dan IMEI mahasiswa dengan menggunakan persamaan (3):

$$x = h(NIM \oplus IMEI) \qquad (3)$$

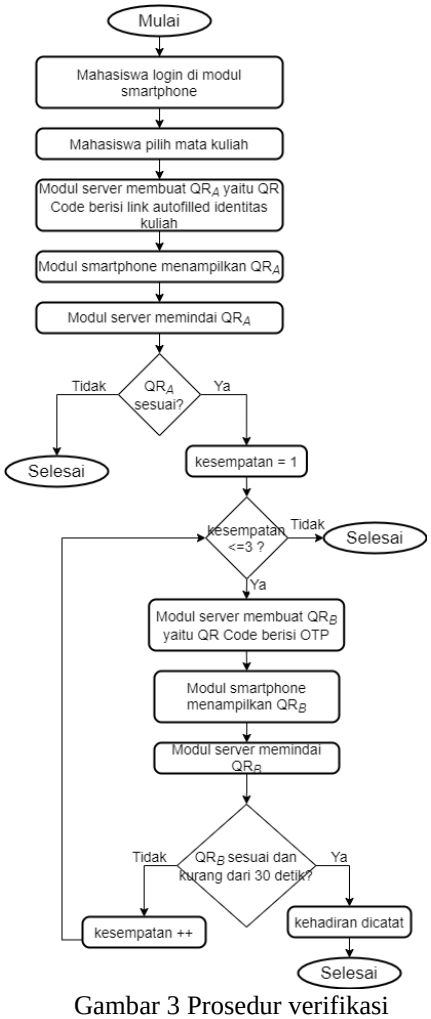


Gambar 2 Prosedur registrasi

- Langkah 3: Modul *server* akan menyimpan nilai x bukan IMEI
- Langkah 4: Staf administrasi kampus akan mengatur jadwal perkuliahan dan identitas mahasiswa yang mengikuti perkuliahan tersebut.

3.1 Prosedur Verifikasi

Prosedur verifikasi dilakukan saat mahasiswa mencatatkan kehadirannya pada saat perkuliahan. Prosedur verifikasi secara garis besar dapat dilihat pada Gambar 3 yang memiliki rincian sebagai berikut.



Gambar 3 Prosedur verifikasi

- Langkah 1: sebelum masuk kelas, mahasiswa membuka modul *smartphone* yang terlebih dahulu diinstall.
- Langkah 2: mahasiswa *login* pada modul *smartphone* dengan menginputkan username yang berupa NIM dan password.
- Langkah 3: mahasiswa akan diberikan list jadwal sesi kuliah pada hari ini.
- Langkah 4: modul *server* akan membuat *link* unik yang bersifat *autofilled* berisi identitas sesi kuliah, mahasiswa, dan *smartphone*. Link ini dinotasikan sebagai L_a berdasarkan persamaan (4).

$$L_a = (ID_s, NIM, IMEI) \quad (4)$$

- Langkah 5: modul *smartphone* akan mengumpulkan identitas sesi kuliah yang dipilih, identitas mahasiswa, dan identitas *smartphone* yang digunakan secara otomatis
- Langkah 6: modul *smartphone* memodifikasi L_a menjadi $L_a{}'$ sesuai data yang diperoleh di Langkah 6. Kemudian mengakses link $L_a{}'$ yang merupakan *link autofilled*.
- Langkah 7: modul *smartphone* akan membuat QR code berisi link $L_a{}'$ dan menampilkannya di layar

$$QR_1 = E_{QR}(L_a{}') \quad (5)$$

- Langkah 8: modul *server* kemudian memindai QR_1 dan mengubahnya menjadi link $L_a{}'$

$$L_a{}' = D_{QR}(QR_1) \quad (6)$$

- Langkah 9: berdasarkan data sesi kuliah dan NIM yang dikirim pada Langkah 6, modul *server* akan mencocokan data dari $L_a{}'$ yang diberikan dengan daftar mahasiswa yang mengikuti sesi kuliah tersebut. Jika data tidak cocok, maka modul *smartphone* dan *server* akan memberikan peringatan bahwa mahasiswa tersebut tidak terdaftar pada sesi kuliah, kemudian prosedur verifikasi selesai. Jika data cocok, maka modul *server* akan menghitung nilai hash y seperti pada persamaan (7). Tujuannya untuk memastikan apakah mahasiswa tersebut menggunakan *smartphone* miliknya.

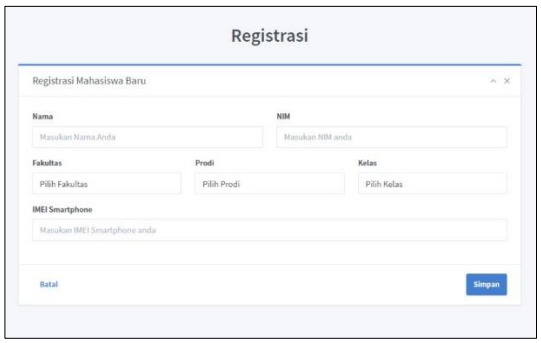
$$y = h(NIM \oplus IMEI) \quad (6)$$

- Langkah 10: modul *server* akan mencocokan apakah y memiliki nilai yang sama dengan x pada Prosedur Registrasi. Jika nilai x tidak sama y maka modul *smartphone* akan memberikan peringatan bahwa mahasiswa harus menggunakan *smartphone* dengan IMEI yang terdaftar kemudian prosedur verifikasi selesai. Jika x dan y identik, maka proses dilanjutkan ke Langkah 11. Langkah pencocokan identitas ini dilakukan sebagai pemicu agar modul server tidak membuat *password* dinamis TOTP setiap waktu dan hanya dilakukan jika mahasiswa benar-benar teridentifikasi dan berada di dekat komputer kelas yang terhubung dengan modul *server*.

- Langkah 11: modul *server* akan membuat *password* dinamis TOTP yang hanya berlaku 30 detik saja seperti pada persamaan (1) dan (2) dengan nilai $v = 30$
- Langkah 12: modul *server* akan menghitung nilai *hash* dari IMEI *smartphone* mahasiswa (yang dihasilkan dari Langkah 6) dan TOTP yang dihasilkan kemudian menyimpannya sebagai r berdasarkan fungsi berikut.
$$r = h(IMEI \oplus TOTP) \quad (7)$$
- Langkah 13: modul server akan membuat *link* yang bersifat *autofilled* berisi nilai hash IMEI dan TOTP. *Link* ini dinotasikan sebagai L_b berdasarkan fungsi berikut.
$$L_b = (r) \quad (8)$$
- Langkah 14: modul *smartphone* akan membuat *QR Code* berisi *link* L_b berdasarkan persamaan (9) dan menampilkannya di layar.
$$QR_2 = E_{QR}(L_b) \quad (9)$$
- Langkah 15: modul *server* kemudian memindai QR_2 dan mengubahnya menjadi *link* L_b dan mengakses *link* tersebut
$$L_b = D_{QR}(QR_2) \quad (10)$$
- Langkah 16: modul *server* akan mengevaluasi nilai hash dari IMEI dan TOTP yang diberikan, nilai hash ini dinotasikan sebagai s
- Langkah 17: modul *server* akan mencocokan nilai r pada Langkah 12 dan s pada Langkah 16. Jika r dan s identik, maka kehadiran mahasiswa dicatat. Jika tidak identik, maka kembali ke Langkah 11. Jika Langkah 11 sudah dilakukan tiga kali (mahasiswa sudah mencoba memindai QR code berisi TOTP sebanyak tiga kali), maka proses verifikasi terhenti dan mahasiswa dianggap tidak hadir.
- Langkah 18: pada akhir sesi kuliah, dosen dapat mengakses modul *server* diberi kesempatan untuk mengecek ulang kehadiran mahasiswa di kelasnya.

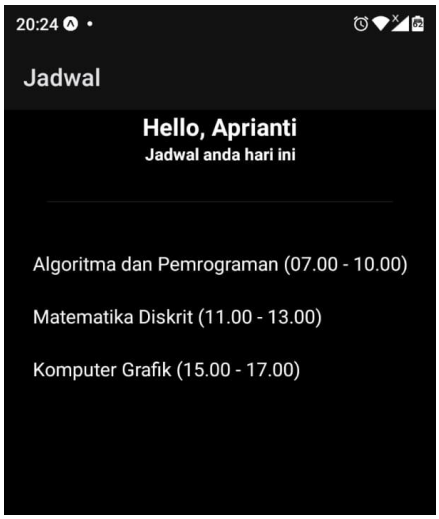
4. Implementasi

Sistem yang sudah dirancang, akan diimplementasikan dalam sebuah purwarupa sistem. Berikut adalah tampilan purwarupa sistem.

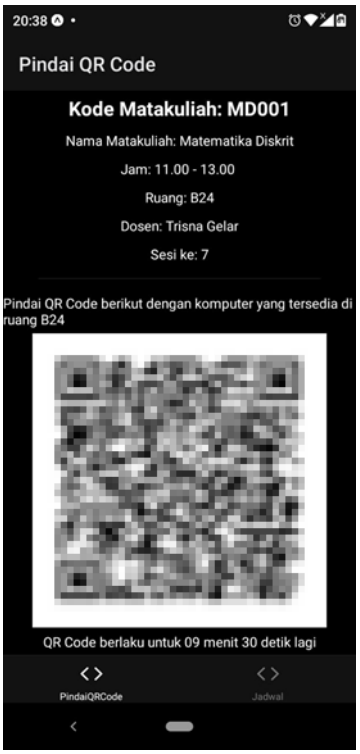


Gambar 4 Registrasi

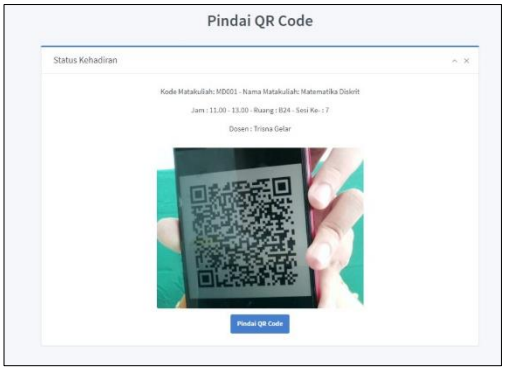
Gambar 3 adalah tampilan modul server saat prosedur registrasi. Pada Gambar 4, modul *smartphone* menampilkan jadwal mahasiswa pada hari ini. Setelah memilih kelas, maka akan tampil *QR Code* berisi *link autofilled* identitas kelas yang dipilih seperti pada Gambar 5. *QR Code* tersebut hanya berlaku pada saat jadwal perkuliahan di mulai.



Gambar 5 Modul *smartphone* menampilkan jadwal kuliah



Gambar 6 Modul *smartphone* menampilkan QR Code berisi identitas kuliah

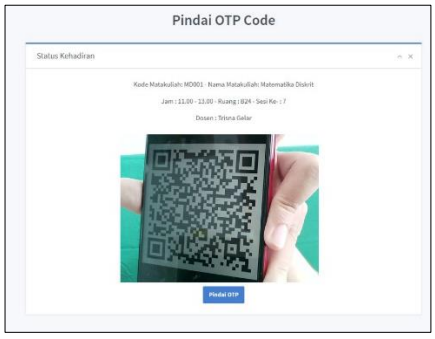


Gambar 7 Modul server memindai QR Code berisi identitas kuliah

Modul *server* kemudian akan memindai *QR Code* berisi *link autofilled* identitas kelas seperti pada Gambar 6. Jika sesuai, maka modul *smartphone* akan menampilkan *QR Code* berisi TOTP seperti pada Gambar 7.

Gambar 8 Modul smartphone menapilkan QR Code berisi TOTP

Setelah itu, modul *server* akan memindai QR Code berisi TOTP seperti pada Gambar 8. Jika sesuai, maka kehadiran mahasiswa akan dicatat seperti pada Gambar 9.



Gambar 9 Modul server memindai QR Code berisi OTP

Gambar 10 Kehadiran mahasiswa berhasil dicatat

5. Pengujian dan Analisa

Pada bab ini akan dibahas mengenai pengujian terhadap fungsionalitas sistem serta analisa keamanan.

5.1 Pengujian *Blackbox*

Pengujian *blackbox* hanya bertujuan menjaga fungsionalitas sistem tanpa memperhatikan detailnya.

Tabel 2 Pengujian *blackbox* prosedur registrasi

No	Pengujian	Hasil
1	Input identitas mahasiswa	Berhasil
2	Input IMEI	Berhasil
3	Menghitung nilai hash	Berhasil
4	Menyimpan identitas mahasiswa	Berhasil
5	Menyimpan nilai hash	Berhasil

Tabel 3 Pengujian *blackbox* prosedur verifikasi

No	Pengujian	Hasil
1	Modul server membuat QR_A (*QR Code* berisi *link autofilled* identitas kuliah)	Berhasil
2	Modul *smartphone* menapilkan QR_A	Berhasil
3	Modul *server* memindai QR_A	Berhasil
4	Modul *server* mengecek apakah QR_A sesuai dengan jadwal	Berhasil
5	Modul server membuat QR_B (*QR Code* berisi TOTP)	Berhasil
6	Modul *smartphone* menapilkan QR_B	Berhasil
7	Modul *server* memindai QR_B	Berhasil
8	Modul *server* mengecek apakah QR_B sesuai dan masih berlaku (tidak lebih dari 30 detik)	Berhasil

5.2 Analisa Keamanan

Analisa keamanan bertujuan untuk mengetahui kemungkinan penyerangan yang dilakukan.

5.2.1 Penyerangan Terhadap IMEI *Smartphone*

Jika seseorang dapat menebak IMEI *smartphone* yang mahasiswa gunakan, maka ada kemungkinan ia juga bisa memodifikasi TOTP pada tahap verifikasi Langkah 10. Akan tetapi, hal ini sulit dilakukan karena IMEI yang disimpan dalam bentuk *hash* sesuai dengan persamaan (3). Seperti yang kita

ketahui bahwa *hash* adalah sebuah fungsi satu arah sehingga tidak bisa dipecahkan (Singh & Garg 2009).

5.2.2 *Man-in-the-middle attack*

Man-in-the-middle attack adalah sebuah istilah yang menggambarkan pihak ketiga yang mencoba memposisikan diri atau berpura-pura menjadi pengguna dari sebuah sistem. Tujuannya adalah untuk memata-matai atau berpura-pura menjadi pihak lain sehingga terjadi pertukaran informasi yang dirahasiakan (Mallik *et al*, 2019).

Dalam kasus ini, misalnya seorang penyelundup mencoba menyadap ketika modul *server* dan modul *smartphone* bertukar informasi mengenai TOTP pada prosedur verifikasi langkah 15. Akan tetapi, hal ini tidak mungkin terjadi karena *QR Code* yang berisi TOTP sudah berupa *hash* sesuai persamaan (7).

6. Kesimpulan dan Saran

Berdasarkan hasil pengujian *blackbox*, sistem ini berhasil bekerja secara fungsional. Selain itu, dengan penggunaan TOTP pada penelitian ini, kecurangan mahasiswa yang mengambil gambar *QR Code* dan memberikannya pada mahasiswa yang tidak hadir dapat dihindari karena penggunaan *QR Code* yang bersifat dinamis dan TOTP yang hanya berlaku dalam waktu singkat. Selain itu, kemungkinan mahasiswa meminjam *smartphone* temannya untuk absen juga dapat dihindari karena sistem absesni ini hanya dapat diakses oleh mahasiswa yang IMEI *smartphone*-nya terdaftar.

Tidak hanya itu, berdasarkan analisa keamanan penggunaan *hash* untuk menyimpan identitas mahasiswa dan *smartphone* juga dapat menghindari penyerangan terhadap IMEI dan metode penyerangan *man-in-the-middle attack*.

Karena sistem ini masih berupa purwarupa, maka penulis menyarankan untuk melanjutkan pengembangan sistem yang lebih kompleks dan sesuai dengan standar prosedur absensi di perguruan tinggi.

Daftar Pustaka:

Aini, Q., Graha, Y. I., & Zuliana, S. R. (2017). *Penerapan Absensi QRCode Mahasiswa Bimbingan Belajar pada Website berbasis YII Framework*. Sisfotenika, 7(2), 207-218.

Ardhianto, E. (2016). Mesin Presensi Cepat Dengan Menggunakaqr Code Dan Webcam. Jurnal Informatika Upgris, 2(2).

Babkin, S., & Epishkina, A. (2019, January). *Authentication Protocols Based on One-Time Passwords*. In 2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus) (pp. 1794-1798). IEEE.

Hanks, H. "*Successful Scanning: A Guide to QR Code Best Practices*." (2012): 1-18.

Hermanto, N., & Riyanto, N. R. D. R. (2019). Aplikasi sistem presensi mahasiswa berbasis android. Simetris: Jurnal Teknik Mesin, Elektro dan Ilmu Komputer, 10(1), 107-116.

Liao, K. C., Lee, W. H., Sung, M. H., & Lin, T. C. (2009, August). *A one-time password scheme with QR-code based on mobile phone*. In 2009 Fifth International Joint Conference on INC, IMS and IDC (pp. 2069-2071). IEEE.

Lamport, L. (1981). *Password authentication with insecure communication*. Communications of the ACM, 24(11), 770-772.

Mallik, A. (2019). *Man-in-the-middle-attack: Understanding in simple words*. International Journal of Data and Network Science 3(2): 77-92.

Malik, J., Girdhar, D., Dahiya, R., & Sainarayanan, G. (2014). *Multifactor Authentication Using a QR Code and a One-Time Password*. JIPS, 10(3), 483.

M'Raihi, D., Machani, S., Pei, M., & Rydell, J. (2011). *Totp: Time-based one-time password algorithm*. Internet Engineering Task Force, May 2011

Nuddin, Mukhamad Taqwa, and Diana Laily Fithri. "*Sistem Absensi Asisten Dosen Menggunakan QR Code Scanner Berbasis Android Pada Program Studi Sistem Informasi Universitas Muria Kudus*." Prosiding SNATIF (2015): 303-310.

Pujastuti, E., & Laksito, A. D. (2020). *Usability Testing on QR Code Scanner Application for Lecture Presence*. Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika, 6(1), 50-57.

Singh, A., & Singh, P. (2016). *A review: QR codes and its image pre-processing method*. International Journal of Science, Engineering and Technology Research, 5(6), 1955-1960.

Singh, M., & Garg, D. (2009, March). *Choosing best hashing strategies and hash functions*. In 2009 IEEE International Advance Computing Conference (pp. 50-55). IEEE.

Soon, T. J. (2008). *QR code*. Synthesis Journal, 2008, 59-78.