

Rancang Bangun Aplikasi *Secure Lab* Pada Ruangan Laboratorium Telekomunikasi Politeknik Negeri Jakarta

Viving Frendiana¹, Afif Tadjuddin Shafi²

^{1,2} Program Studi Broadband Multimedia, Jurusan Teknik Elektro,
Politeknik Negeri Jakarta

Jl. Prof. DR. G.A. Siwabessy, Kampus Universitas Indonesia Depok 16425

viving.frendiana@elektro.pnj.ac.id, afif.tadjuddinshafi.te18@mhs.pnj.ac.id

Abstrak

Laboratorium Teknik Telekomunikasi merupakan salah satu laboratorium yang berada di naungan Jurusan Teknik Elektro, Politeknik Negeri Jakarta. Untuk peningkatan keamanan ruangan laboratorium tersebut dapat digunakan suatu aplikasi *mobile* yang bernama *Secure Lab*, dimana suatu kunci pintu dengan kode QR terhubung pada aplikasi tersebut. Kode QR berfungsi sebagai kunci ruangan dibangun menggunakan Android Studio dan memanfaatkan bahasa pemrograman Kotlin. Aplikasi ini juga terkoneksi dengan *Authentication* dan *Realtime Database* pada *Firebase*. Hak akses aplikasi *Secure Lab* ada dua yaitu sebagai dosen dan mahasiswa. Untuk roles sebagai mahasiswa maka perlu membuat akun terlebih dahulu supaya dapat melakukan permohonan akses ruangan menggunakan kode QR dan untuk membuka pintu. *Role* sebagai dosen mempunyai hak untuk memberikan akses kode QR kepada mahasiswa, yang dapat digunakan untuk membuka pintu ruangan dalam rentang waktu yang sudah ditentukan. Pengujian aplikasi *Secure Lab* dimaksudkan untuk mengetahui kelayakan aplikasi yang dibangun. Dari tiga pengujian: aspek *compatibility*, *functional*, dan *portability* didapatkan bahwa Aplikasi *Secure Lab* sangat layak. Dari pengujian pembacaan kode QR diperoleh hasil kode QR yang digenerate aplikasi *Secure Lab* dapat dibaca oleh *scanner*. Pengujian berdasarkan jarak diperoleh 10-16 cm adalah rentang jarak yang masih bisa dibaca oleh *scanner*.

Kata kunci—*Mobile Application, Secure Lab, Kode QR*

Abstract

The Telecommunications Engineering Laboratory is one of the laboratories under the Department of Electrical Engineering, at Jakarta State Polytechnic. To increase the security of the laboratory room, a mobile application called Secure Lab can be used, where a door lock with a QR code is connected to the application. The QR code functions as a room key which was built using Android Studio and uses the Kotlin programming language. This application is also connected to Authentication and Realtime Database on Firebase. There are two access rights to the Secure Lab application, namely as a lecturer and a student. To act as a student, you need to create an account first so you can request room access using a QR code and open the door. Acting as a lecturer has the right to provide QR code access to students, which can be used to open the room door within a predetermined period. Secure Lab application testing is intended to determine the application's feasibility. From three tests: Compatibility, Functional Suitability, and Portability aspects, very decent results were obtained. From the QR code reading test, the result was that a scanner could read the QR code produced by the Secure Lab application. Testing based on the distance obtained results that 10-16 cm is the distance that can still be read by the scanner.

Keywords—*Mobile Application, Secure Lab, QR Code*

1. PENDAHULUAN

Kemajuan di bidang teknologi sangat berdampak dalam aktivitas sehari-hari, tidak terkecuali di bidang pendidikan. Dimana dalam kegiatan pengajaran banyak memanfaatkan teknologi, yang membutuhkan perangkat sistem informasi dan komputer untuk mendukung kegiatan tersebut (Ghaffur, 2017). Pemanfaatan teknologi yang banyak digunakan adalah aplikasi berbasis *mobile* dengan sistem operasi Android. Android dibangun dengan menggunakan bahasa pemrograman Java atau Kotlin dan berjalan disebuah lingkungan *Software Development Kit* (SDK). Android Studio adalah salah satu editor yang bisa digunakan untuk membangun suatu aplikasi android.

Aplikasi *mobile* dapat dimanfaatkan untuk membuat suatu sistem keamanan secara otomatis, yang bisa diterapkan untuk mengamankan ruangan-ruangan laboratorium. Terutama Laboratorium Teknik Telekomunikasi PNJ yang masih menggunakan kunci manual. Alat dan perlengkapan praktikum di suatu ruangan laboratorium bisa lebih aman dengan menerapkan kunci yang bisa dikontrol menggunakan aplikasi berbasis *mobile*.

Smart Storage merupakan penelitian yang berfokus pada aplikasi android (Arief, 2020). Dimana dalam penelitian tersebut menggunakan mikrokontroler ESP32 CAM dan EM Lock dengan input dari aplikasi *mobile*. Indikator ketersediaan loker diperoleh sebagai output sensor jarak ultrasonik. Dari penelitian tersebut dapat dikembangkan untuk sistem pengaman pintu pada suatu ruangan laboratorium berbasis aplikasi *mobile*. Program aplikasi yang dibuat yaitu berupa aplikasi *mobile* berbasis Android bernama "Secure Lab". Aplikasi ini dapat digunakan oleh mahasiswa yang telah mendapat akses untuk membuka kunci pintu laboratorium. Aplikasi menampilkan status dari mahasiswa apakah sudah memiliki akses atau belum. Selain itu, ditampilkan juga kode QR yang sudah dibuat berdasarkan data diri mahasiswa. Kepemilikan kode QR hanya untuk mahasiswa dan dosen di Lab Telekomunikasi PNJ. Mahasiswa harus meminta akses dalam rentang waktu tertentu terlebih dahulu kepada dosen agar kode QR dapat digunakan untuk membuka pintu. Lalu, kode QR hanya dapat digunakan hanya dalam rentang waktu permintaan akses yang diajukan oleh mahasiswa.

2. METODE PENELITIAN

Metode penelitian yang digunakan adalah mencari referensi terkait topik yang akan diambil, merancang pembuatan aplikasi, merealisasikan hasil rancangan tersebut dan melakukan pengujian aplikasi.

2.1. Studi Literatur

Pada tahap pertama adalah melakukan studi literatur untuk mengumpulkan berbagai referensi terkait topik penelitian.

2.2. Perancangan

Pada tahap ini dibahas mengenai *user requirement* yang merupakan kebutuhan user dan fitur-fitur yang terdapat pada aplikasi dari sisi pengguna. *User requirement* ini akan menentukan perencanaan selanjutnya mulai dari halaman yang akan dibuat beserta komponen dan fitur yang terdapat didalamnya. Pada perancangan aplikasi *Secure Lab*, membutuhkan dua *user requirement* yaitu dari sisi mahasiswa dan dosen.



Gambar 1. Bagan Alur Cara Kerja Aplikasi *Secure Lab*

Adapun penjelasan dari setiap tahapan pada gambar 1 yaitu sebagai berikut.

1. Pengguna memasang aplikasi pada *smartphone*.
2. Melakukan proses *login* sebagai mahasiswa atau dosen. Apabila mahasiswa belum mempunyai akun, maka klik *sign up* untuk membuat akun.
3. Masukkan data diri mahasiswa lalu tekan *sign up*.
4. Setelah proses *login*, pengguna akan diarahkan ke dalam menu *home*.
5. Jika *login* sebagai dosen, maka menu *home* akan berisi daftar mahasiswa yang sedang menunggu akses untuk diberikan. Lalu, akan ada *button accept* untuk menerima akses, dan *button reject* untuk menolak akses.
6. Jika *login* sebagai mahasiswa, maka akan ada status akses, *button Make Request*, dan *button Show QR Code*. Di samping menu *home* juga ada menu *profile* untuk menampilkan data diri mahasiswa yang sebelumnya sudah dimasukkan.
7. Jika status akses masih merah, maka itu bertanda bahwa mahasiswa belum memiliki akses, dan tekan *button Make Request*.
8. Masukkan waktu mulai hingga waktu selesai untuk akses dan klik *Submit*.
9. Tunggu hingga dosen memberikan akses pada mahasiswa.
10. Jika akses sudah diberikan, maka status akses akan berubah menjadi hijau, dan tekan *button Show QR Code* untuk menampilkan kode QR.
11. Kode QR siap digunakan untuk membuka kunci.

2.3. Pembuatan Aplikasi

Pada tahap ini dilakukan realisasi dan implementasi berdasarkan rancangan yang sudah didesain dan direncanakan sebelumnya.

Tabel 1. Spesifikasi Perangkat Keras (Hardware)

No.	Nama Perangkat Keras	Spesifikasi
1.	Laptop Acer E5-475G	Operating System 64bit; Kapasitas RAM 8GB; OS Windows 10.
2.	Handphone Samsung Galaxy A12	OS Android R 11; Kapasitas RAM 6GB; Ukuran layar 6.5" 720x1600 <i>pixels</i>

Tabel 1 merupakan perangkat keras yang digunakan untuk merancang *aplikasi* yaitu menggunakan Laptop Acer E5-475G dan Handphone Samsung Galaxy A12. *Operating System (OS)* yang digunakan adalah Windows 10.

Tabel 2. Spesifikasi Perangkat Lunak (Software)

No.	Nama Perangkat Keras	Spesifikasi
1.	Windows 10	Operating System 64bit, Kapasitas RAM 8GB; OS Windows 10 Enterprise.
2.	Android Studio IDE	Versi 4.2.1
3.	OS Android	OS Android R 11

Tabel 2 terdapat daftar perangkat lunak yang digunakan untuk merancang aplikasi yaitu Android Studio IDE versi 4.2.1 dan OS Adroid 11.

2.4. Pengujian

Setelah aplikasi berhasil dibuat maka dilakukan pengujian dengan 15 user yang terdiri dari mahasiswa dan dosen. Pengujian ini dilakukan dengan menggunakan standar ISO/IEC 25010. Terdapat 8 aspek dalam ISO 25010. Dalam pengujian ini, akan digunakan 3 dari 8 aspek. Aspek – aspek tersebut adalah *compatibility*, *portability*, dan *functional Suitability*. Pengujian *compatibility* akan mengukur sejauh mana sistem dapat berjalan bersamaan dengan aplikasi lain. Pengujian *portability* akan mengukur sejauh mana keefektifan dan efisiensi sebuah sistem dapat digunakan diberbagai versi android dan perangkat lainnya. *Functional suitability* merupakan sejauh mana suatu produk atau sistem yang memenuhi kebutuhan ketika digunakan untuk mencapai tujuan efektif, efisien dan kepuasan dalam konteks pengguna.

$$\% \text{ Keberhasilan} = \frac{i}{r} \times 100\%$$

Keterangan:

i = jumlah kebutuhan fungsional yang berhasil diimplementasikan

r = jumlah total kebutuhan fungsional

Dari persentase tersebut, dapat ditarik kesimpulan menjadi data kualitatif dengan menggunakan tabel 3 seperti berikut.

Tabel 3. Kategori Penilaian *Functional Suitability*

No	% Skor	Interpretasi
1	0%-20%	Sangat tidak layak
2	21%-40%	Tidak layak
3	41%-60%	Cukup layak

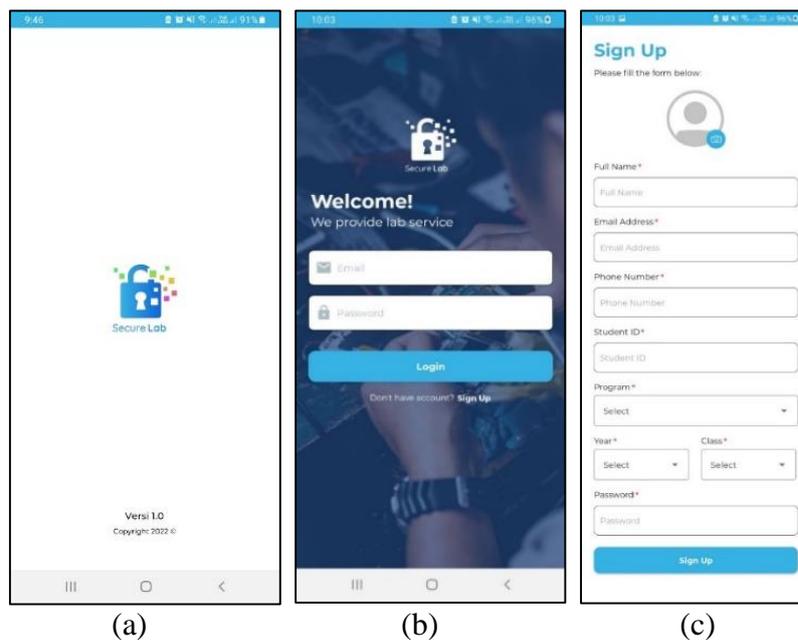
No	% Skor	Interpretasi
4	61%-80%	Layak
5	81%-100%	Sangat layak

3. HASIL DAN PEMBAHASAN

3.1. Realisasi Aplikasi

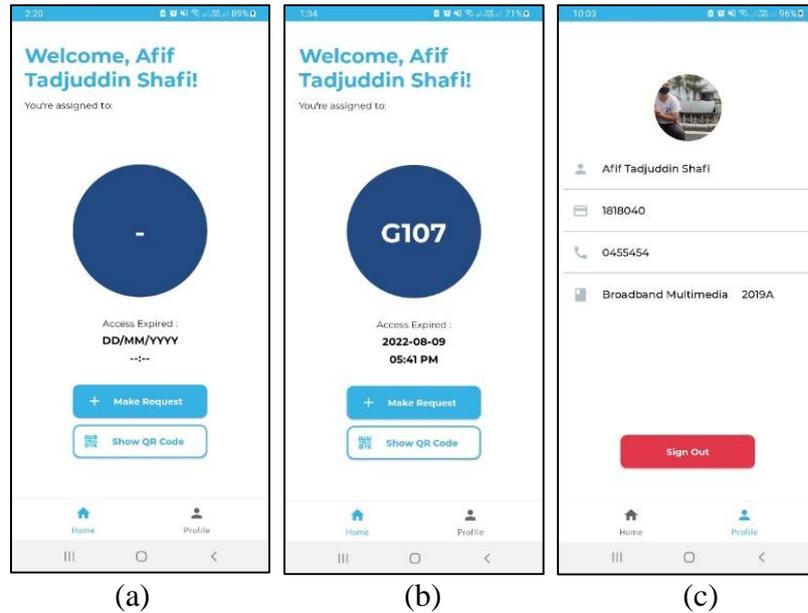
Aplikasi *Secure Lab* direalisasikan dengan bantuan *tools* Android Studio. Lalu, bahasa pemrograman yang digunakan adalah Kotlin. Berdasarkan Gambar 3(a) *splash screen* berisikan logo serta teks. Untuk merealisasikannya digunakan atribut – atribut berupa *ImageView* dan *TextView*. Kedua atribut tersebut kemudian berada dalam satu jenis *layout* yaitu *RelativeLayout*.

Gambar 3(b) terlihat ada dua *input* untuk melakukan *login*, yaitu *email* dan *password*. Jika belum memiliki akun, maka dapat menekan tulisan *Sign Up* yang terletak di bawah *button login*. Pada halaman ini, atribut – atribut yang paling penting digunakan adalah *EditText* dan *Button*. *EditText* berfungsi untuk pengguna melakukan *input data* yang berupa *email* dan *password*. Sedangkan *Button* digunakan sebagai *trigger* yang akan ditambah fungsi programmnya pada skrip kotlin.



Gambar 3. Halaman (a) *Splash Screen*, (b) *Login*, (c) *Sign Up*

Pada Gambar 3(c), halaman ini hanya dikhususkan untuk pengguna mahasiswa. Terdapat beberapa *input* wajib yang apabila tidak diisi akan muncul *error*. Beberapa *input* yang perlu diisi diantaranya foto profil, nama lengkap, NIM, program studi, tahun angkatan, dan kelas. Setelah semua *input* diisi, maka tekan *Sign Up* untuk mengirim *data* menuju *Firebase*. Pada halaman ini, digunakan salah satu *plugin* khusus, yaitu *CircleImageView*. *Plugin* ini digunakan untuk merubah tampilan foto menjadi latar lingkaran yang nantinya digunakan sebagai foto profil dari akun mahasiswa.

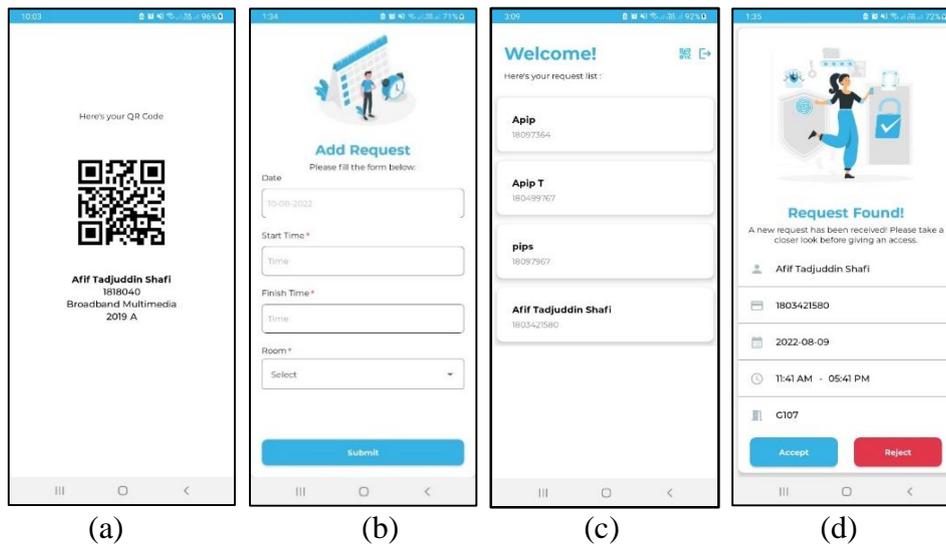


Gambar 4. Halaman (a) Home (Tanpa Akses), (b) Home (dengan Akses), (c) Profil

Pada Gambar 4(a) terlihat bahwa status akses masih tidak aktif atau belum dapat akses. Hal ini ditandai dengan *string* strip (-) pada status akses. Lalu di bawahnya terdapat tanggal dan waktu berakhirnya akses yang masih kosong. Jika mahasiswa sudah mendapat akses, maka status akses akan berubah dan menampilkan *string* ruangan yang sudah diberikan akses kepada mahasiswa.

Pada Gambar 4(b), status akses sudah berubah menjadi aktif ditandai dengan *string* nama ruangan yang ditampilkan. Tanggal dan waktu berakhirnya akses juga akan berubah sesuai dengan permintaan mahasiswa.

Pada Gambar 4(c), halaman profil akan terus menarik *data* dari *database*. Jika *data* pada *database* diubah, maka isi dari halaman profil juga akan berubah. Dalam halaman ini juga ada *button Sign Out* untuk keluar dan kembali ke halaman *login*.



Gambar 5. Halaman (a) Show QR Code, (b) Add Request, (c) Home pada Dosen, (d) Request Details

Pada gambar 5(a) digunakan *library* khusus untuk melakukan *generate* kode QR, yaitu *zxing*. *Library* ini ditambahkan ke dalam *dependencies* *build.gradle* agar dapat digunakan.

Pada gambar 5(b) ditampilkan beberapa *input* yang berupa tanggal, waktu awal akses dan waktu berakhirnya akses. *Input* tanggal tidak dapat diubah karena akses hanya akan diberikan pada hari itu saja. Permohonan akses tidak dapat dibuat untuk hari esok atau beberapa hari kedepan. Selain itu untuk *input* waktu, ditampilkan dalam format 12 jam.

Pada gambar 5(c), halaman ini berisi *RecyclerView* dan menampilkan *list request* yang telah dikirim oleh mahasiswa. Selain itu, juga terdapat *button Show QR* untuk menampilkan kode QR dosen dan *button Sign out* di sebelahnya jika dosen ingin keluar dari akun.

Pada gambar 5(d), selain *data request* yang ditampilkan, ada dua *button* untuk menentukan respon dari dosen. Jika dosen menolak, maka *data request* akan dibuang dari *database*. Namun jika akses diterima, maka nilai status pada *database request* menjadi aktif.

3.2. Pengujian Compatibility

Pengujian *compability* akan mengukur sejauh mana sistem dapat berjalan bersamaan dengan aplikasi lain yang termasuk ke dalam subkarakteristik *co-existance*. Aplikasi *Secure Lab* diharapkan dapat berbagi sumber daya dengan sistem lain tanpa merugikan sistem tersebut. Berikut adalah hasil dari pengujian aspek *compatibility* berdasarkan ISO 25010 dengan subkarakteristik *co-existance* ditampilkan pada tabel 4.

Tabel 4. Hasil Pengujian *Co-existance* Aspek *Compatibility*

No	Nama Aplikasi	Status	
		Berhasil	Gagal
1.	Twitter	✓	-
2.	Youtube	✓	-
3.	Gmail	✓	-
4.	Linkedin	✓	-
Total		4	0

Tabel 4 merupakan kumpulan hasil data pengujian pada aplikasi *Secure Lab* dengan menggunakan aspek *compatibility* dengan sub karakteristik *co-existance*. Dari empat aplikasi yang dijalankan bersamaan dengan aplikasi *Secure Lab*, tidak ada yang mendapatkan *error*, *force close* atau mengalami *crash*. Persentase kelayakan yang didapatkan yaitu:

$$\begin{aligned} \text{Persentase kelayakan (\%)} &= \frac{4}{4} \times 100\% \\ &= 100\% \end{aligned}$$

Persentase kelayakan yang diperoleh sebesar 100%, hasil ini membuktikan Aplikasi *Secure Lab* dapat berjalan dengan baik dan tidak terjadi *crash* ketika dijalankan dengan aplikasi lain.

3.3. Pengujian Functional Suitability

Pengujian *Functional Suitability* akan mengukur aplikasi dengan melakukan *Test case* dan menggunakan metode *Black-Box Test*.

Tabel 5. Hasil Pengujian Aspek *Functional Suitability*

Keberhasilan	
Ya	Tidak
26	0

Berdasarkan data yang didapatkan pada Tabel 5, terdapat 26 *test case* dan semua *test case* berhasil. Sehingga persentase kelayakannya adalah:

$$\begin{aligned} \text{Persentase kelayakan (\%)} &= \frac{26}{26} \times 100\% \\ &= 100\% \end{aligned}$$

Jika melihat dari persentase kelayakan, aplikasi ini memiliki skor 100% dengan kategori Sangat Layak dan dapat digunakan oleh pengguna.

3.4. Pengujian Portability

Pengujian *portability* akan mengukur sejauh mana keefektifan dan efisiensi sebuah sistem dapat digunakan diberbagai versi android dan perangkat lainnya yaitu aplikasi dapat berjalan diatas SDK Android 26 atau Versi Android 8.0 (Oreo). Berikut adalah hasil pengujian pada 4 perangkat yang berbeda ditampilkan pada tabel 6.

Tabel 6. Daftar Perangkat Pengujian Berdasarkan Versi Android yang Berbeda

No.	Nama	Tipe Smartphone	Versi Android	Berhasil	
				Ya	Tidak
1.	Diah	Xiaomi Redmi 6	8 (Oreo)	V	-
2.	Yogi	Oppo A1K	9 (Pie)	V	-
3.	Genta	Oppo F7	10 (Quince Tart)	V	-
4.	Afif	Samsung Galaxy A21	11 (Red Velvet Cake)	V	-

Dari data yang didapatkan pada Tabel 6 didapatkan bahwa aplikasi berjalan pada 4 perangkat yang berbeda dengan versi android beragam, mulai dari versi 8 (Oreo) – 11 (Red Velvet Cake) Jika dihitung persentase kelayakannya:

$$\begin{aligned} \text{Persentase kelayakan (\%)} &= \frac{4}{4} \times 100\% \\ &= 100\% \end{aligned}$$

Didapatkan persentase kelayakan sebesar 100% pada *Secure Lab*. Aplikasi ini membuktikan bawa berhasil berjalan di versi android 8 keatas.

3.5. Pengujian Membaca Data QR

Pada pengujian ini, akan diuji keluaran hasil pembacaan kode QR yang berasal dari aplikasi *Secure Lab*. Aplikasi akan diuji pada 3 akun *user* dan 1 akun *admin* untuk mendapatkan indikator. Selain itu dilakukan juga pengukuran jarak pembacaan kode QR antara alat dengan *handphone* yang sudah melakukan instalasi aplikasi *Secure Lab*. Hasil pembacaan kode QR ditunjukkan pada Tabel 7.

Tabel 7. Hasil Pembacaan Kode QR

No.	Role	Nama Akun	Kondisi	Respon Terbaca di Alat
1	User	Afif Tadjuddin	Tidak Ada Akses	0,Afif Tadjuddin,180318040
2	User	Afif Tadjuddin	Tidak Ada Akses	0,Afif Tadjuddin,180318040
3	User	Afif Tadjuddin	Ada Akses	1,Afif Tadjuddin,180318040
4	User	Afif Tadjuddin	Ada Akses	1,Afif Tadjuddin,180318040
5	User	Meutia Khairiyah	Tidak Ada Akses	0,Meutia Khairiyah,1803421006
6	User	Meutia Khairiyah	Tidak Ada Akses	0,Meutia Khairiyah,1803421006
7	User	Meutia Khairiyah	Ada Akses	1,Meutia Khairiyah,1803421006
8	User	Meutia Khairiyah	Ada Akses	1,Meutia Khairiyah,1803421006
9	User	Andreas Genta	Tidak Ada Akses	0,Andreas Genta,1803421010
10	User	Andreas Genta	Tidak Ada Akses	0,Andreas Genta, 1803421010
11	User	Andreas Genta	Ada Akses	1,Andreas Genta,1803421010
12	User	Andreas Genta	Ada Akses	1,Andreas Genta, 1803421010
13	Admin	Viving Ferdiana	Ada Akses	1,Viving Frendiana,3020401
14	Admin	Viving Ferdiana	Ada Akses	1,Viving Frendiana,3020401
15	Admin	Viving Ferdiana	Ada Akses	1,Viving Frendiana,3020401

Selanjutnya, untuk mengukur jarak deteksi kode QR pada aplikasi *Secure Lab* terhadap *scanner*, maka digunakan bantuan penggaris. Pada pengujian ini dilakukan deteksi secara bertahap selama 2 detik. Dalam pengujian tersebut, dilakukan pembacaan respon hingga sejumlah 11 buah. Berikut adalah hasil pembacaan kode QR dengan *scanner* ditampilkan pada tabel 8.

Tabel 8. Hasil Pembacaan Kode QR Berdasarkan Jarak

No.	Jarak Pengujian (cm)	Banyaknya Respon Terbaca	Banyaknya Respon Tidak Terbaca
1	5	0	11
2	8	0	11
3	10	9	2
4	12	8	3
5	14	5	6
6	16	5	6
7	18	0	11
8	20	0	11

Berdasarkan pengujian yang telah dilakukan, didapatkan bahwa kode QR dapat terbaca sesuai dengan tujuan yang diharapkan. Dengan menggunakan persamaan persentase kelayakan, maka didapatkan:

$$\begin{aligned} \text{Persentase kelayakan (\%)} &= \frac{11}{11} \times 100\% \\ &= 100\% \end{aligned}$$

Berdasarkan perhitungan di atas, didapatkan bahwa aplikasi mendapatkan skor kelayakan 100% dan dikategorikan Sangat Layak. Aplikasi berhasil menampilkan kode QR dan kode tersebut dapat terbaca pada *scanner*.

4. KESIMPULAN

Berdasarkan data hasil pengujian didapatkan kesimpulan berikut:

1. Kode QR pada *Secure Lab* App berhasil dibuat dan berhasil dideteksi oleh alat scanner.
2. Pengujian yang dilakukan dari tiga aspek: *compatibility*, *functional suitability*, dan *portability* memperoleh keberhasilan sebesar 100% maka berdasarkan kategori penilaian pada tabel 3, aplikasi mendapatkan hasil yang “Sangat Layak”.

5. SARAN

Perlu adanya penelitian lebih lanjut untuk menambahkan fitur-fitur pendukung dari sisi keamanan yang lebih *Secure*.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada pihak yang membantu dalam penyelesaian penelitian ini terutama pada seluruh civitas akademik Program Studi Broadband Multimedia, Jurusan Teknik Elektro, Politeknik Negeri Jakarta.

DAFTAR PUSTAKA

- Asfihan, Akbar. (2021) *Firestore* Adalah: Cara Kerja, Sejarah, Manfaat, Kelebihan dan Kekurangannya. 7 Januari 2021. <https://Adalah.Co.Id/Firebase>. Diakses Pada 1 Agustus 2022
- Arief, H. B. D., & Frenidiana, V. (2022). Aplikasi Android Loker Penyimpanan *Keyless* Berbayar pada Stasiun MRT. *Spektral*, 3(1).
- Darina, L. (2021, 6 Februari) How Fast Is Technology Growing Statistics [Updated 2022]. (2022)., dari <https://lefronic.com/blog/how-fast-is-technology-growing-statistics/> . Diakses pada 18 Februari 2022
- Dicoding (2020). Apa Itu Firebase? Pengertian, Jenis-Jenis, dan Fungsi Kegunaannya. <https://Www.Dicoding.Com/Blog/Apa-Itu-Firebase-Pengertian-Jenis-Jenis-Dan-Fungsi-Kegunaannya/> . Diakses Pada 1 Agustus 2022
- Ghaffur, T. A. (2017). Analisis Kualitas Sistem Informasi Kegiatan Sekolah Berbasis *Mobile* Web di SMK Negeri 2 Yogyakarta. *Elinvo (Electronics, Informatics, And Vocational Education)*, 2(1).
- Gunawan, H., & Saputro, A. K. H. (2018). Pemanfaatan Aplikasi *Mobile* untuk Mempercepat Pencarian Tempat Indekos Berbasis Android. *Jurnal Muara Sains, Teknologi, Kedokteran dan Ilmu Kesehatan*, 1(2), 85-96.
- Iso25000. Iso/Iec 25010. <https://Iso25000.Com/Index.Php/En/Iso-25000-Standards/Iso-25010?Start=0>. Diakses Pada 1 Agustus 2022.
- Maulana, I. F. (2020). Penerapan *Firestore Realtime Database* pada Aplikasi E-Tilang Smartphone berbasis *Mobile* Android. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, 4(5), 854-863.
- Muslihudin, M., Renvillia, W., Taufiq, T., Andoyo, A., & Susanto, F. (2018).

Implementasi Aplikasi Rumah Pintar Berbasis Android dengan Arduino Microcontroller. Jurnal Keteknikan dan Sains (JUTEKS), 1(1), 23-31.

Prayoga, D., & Simanjuntak, P. (2020). Rancang Bangun Prototipe dan Aplikasi Android Qrcode *Mobile* Parking Berbasis Arduino. Journal Information System Development (ISD), 5(2).

Rahmalia, Nadiyah (2021). Praktis Berbagi Informasi dengan Qr Code, Sudahkah Kamu Mencobanya? 19 Januari 2021. <https://Glints.Com/Id/Lowongan/Qr-Code-Adalah/#.Ypgjp6jiviu>. Diakses Pada 28 Juli 2022.