

SMART PRESENSI MENGGUNAKAN QR-Code DENGAN ENKRIPSI VIGENERE CIPHER

Moh. Lukman Sholeh¹⁾, Lutfi Ali Muharom²⁾

^{1,2)}*Prodi Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Jember*

¹⁾mohlukmansholeh3001@gmail.com, ²⁾lutfi.muharom@unmuhiember.ac.id

Abstrak

Penggunaan smart presensi yang dikombinasikan dengan teknologi QR-Code dapat memberikan kepraktisan dan dapat memberikan solusi agar presensi berjalan dengan baik dan efisien. Pengawas ujian tidak lagi membubuhkan tanda tangan pada kartu ujian dan mahasiswa tidak lagi mencetak kartu ujian. Pemanfaatan fungsi dari smartphone akan memudahkan dosen dalam melakukan presensi secara online. Nomor Ujian dan NIM mahasiswa akan tersimpan dalam database dan akan ditampilkan menggunakan QR-Code, saat pengawas ujian melakukan scanning QR-Code, maka mahasiswa akan menyerahkan QR-Code yang telah tercetak di HP. Pengamanan data yang dilakukan adalah dengan memanfaatkan kode batang QR-Code menggunakan enkripsi vigenere cipher. Aplikasi Smart Presensi pada Ujian di Universitas Muhammadiyah Jember merupakan alternative untuk mempermudah dan menyederhanakan proses presensi.

Katakunci: *Qr-Code; Smart Presensi; Kriptografi; vigenere cipher*

1. Pendahuluan

Pada perkembangan teknologi yang semakin maju dan pesat saat ini akan tentu berpengaruh pada kemudahan-kemudahan yang diberikan dalam kehidupan sehari-hari terutama dalam bidang pemerintahan, perusahaan dan pendidikan, sekolah, universitas dan tempat lain. Sistem presensi dalam kegiatan belajar mengajar di dalam suatu perguruan tinggi, tentu saja memiliki mahasiswa/mahasiswi yang harus di catat setiap hari. Pencatatan kehadiran ini lebih sering dikenal sebagai presensi. Presensi adalah salah satu faktor penting dalam dunia universitas.

Masalah yang terjadi dalam sistem yang telah diterapkan terletak pada pengumpulan data-data hadir mahasiswa yang dilakukan dalam kelas, di mana sistem presensi yang berjalan saat ini masih dilakukan secara manual, yaitu mengumpulkan semua kartu ujian dan ditanda tangani oleh dosen pengawas dan masih offline. Di era modern seperti sekarang ini, tidak menutup kemungkinan bahwa telepon seluler dapat dimanfaatkan untuk sistem presensi di perguruan tinggi.

Salah satu fitur dari *Smartphone* yang menarik adalah kemampuannya untuk mengambil, menyimpan, serta menampilkan gambar dengan format JPEG karena sebagian besar *Smartphone* memiliki kamera. Ide yang muncul adalah untuk memanfaatkan *QR Code* dan *Smartphone* Android untuk menjadi sistem presensi. Dengan memanfaatkan *QR Code*, data mahasiswa dapat disimpan dalam bentuk gambar *QR Code* yang kemudian disimpan di dalam ponsel ataupun di cetak.

Beberapa hal inilah yang mendorong pemikiran mengenai membangun sistem yang dapat melakukan absensi mahasiswa secara mobile, cepat, efektif dan efisien. Android dan Qr-Code digunakan dalam penelitian ini karena android merupakan OS (*Operating System*) *mobile* yang sangat populer dan banyak digunakan dan QR Code merupakan media yang digunakan dalam penyampaian informasi secara cepat dan mendapat respons yang cepat tanpa melakukan input secara manual dengan cara mengetik. Informasi yang dikodekan dalam QR Code dapat berupa URL, nomor telepon, pesan SMS, V-Card, atau teks apapun (Ashford, 2010).

Penggunaan sistem pengamanan dengan menggunakan sandi klasik yang memiliki konsep relative sederhana dan banyak digunakan sampai sekarang yaitu vigenere cipher akan membuat sistem ini memiliki sistem pengamanan yang tidak dengan mudah dipalsukan. Modifikasi yang sedemikian rupa yang digabungkan dengan QR-Code menjadikan sandi ini menjadi simple dan rumit dalam pemecahannya.

2. Tinjauan Pustaka

2.1 Quick Response Code

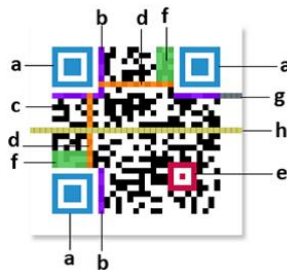
Quick Response Code sering di sebut *Qr Code* atau Kode *QR* adalah semacam simbol dua dimensi yang dikembangkan oleh Denso Wave yang merupakan anak perusahaan dari Toyota sebuah perusahaan Jepang pada tahun 1994. Tujuan dari *Qr Code* ini adalah untuk menyampaikan informasi secara cepat dan juga mendapat tanggapan secara cepat. Pada awalnya *Qr Code* digunakan untuk pelacakan bagian kendaraan untuk *manufacturing*. Namun sekarang, telah digunakan untuk komersil yang ditujukan pada pengguna telepon seluler. *Qr Code* adalah perkembangan dari *barcode* atau kode batang yang hanya mampu menyimpan informasi secara horizontal sedangkan *QR Code* mampu menyimpan informasi lebih banyak, baik secara horizontal maupun vertikal.



Gambar 1 Contoh Qr Code “Saya adalah mahasiswa teknik informatika di Universitas Muhammadiyah Jember”

QR Code biasanya berbentuk persegi putih kecil dengan bentuk geometris hitam (dapat dilihat di gambar 1), meskipun sekarang banyak yang telah berwarna dan digunakan sebagai brand produk. Informasi yang dikodekan dalam *QR Code* dapat berupa URL, nomor telepon, pesan SMS, *V-Card*, atau teks apapun (Ashford,2010). *QR Code* telah mendapatkan standarisasi internasional SO/IEC18004 dan Jepang JIS-X-0510 (Denso, 2011).

2.1.1 Anatomi Qr Code

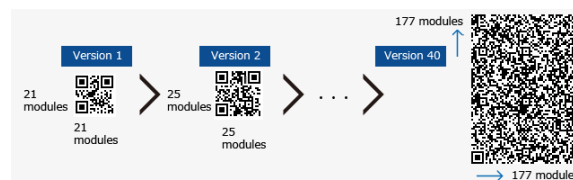


Gambar 2 Anatomi *Qr Code*

Beberapa penjelasan anatomi *Qr Code* Menurut Ariadi (2011) antara lain

- Finder Pattern* berfungsi untuk identifikasi letak *Qr Code*.
- Format Information* berfungsi untuk informasi tentang *error correction level* dan *mask pattern*.
- Data* berfungsi untuk menyimpan data yang dikodekan.
- Timing Pattern* merupakan pola yang berfungsi untuk identifikasi koordinat pusat *Qr Code*, berbentuk modul hitam putih.
- Alignment Pattern* merupakan pola yang berfungsi memperbaiki penyimpangan *Qr Code* terutama distorsi non linier.
- Version Information* adalah versi dari sebuah *Qr Code*.
- Quiet Zone* merupakan daerah kosong di bagian terluar *QR Code* yang mempermudah mengenali pengenalan *QR* oleh sensor *CCD*.
- QR Code* version adalah versi dari *Qr Code* yang digunakan.

2.1.2 Versi Qr Code



Gambar 3 Versi *Qr Code* (Sumber: qrcode.com)

Qr Code dapat menghasilkan 40 versi yang berbeda dari versi 1 (21 x 21 modul) sampai versi 40 (177 x 177 modul). Tingkatan Versi *Qr Code* 1 dan 2 berbeda 4 modul berlaku sampai dengan versi 40. Setiap versi memiliki konfigurasi atau jumlah modul yang berbeda. Modul ini mengacu pada titik hitam dan putih yang membentuk suatu *QR Code*. Setiap versi *QR Code* memiliki kapasitas maksimum data, jenis karakter dan tingkat koreksi kesalahan. Jika Jumlah data yang ditampung banyak maka modul yang akan diperlukan dan menjadikan *Qr Code* menjadi lebih besar (Denso, 2011).

2.1.3 Mengoreksi kesalahan Qr Code

QR Code mampu mengoreksikesalahan dan pengembalian data dalam pembacaan kode apabila *qr code* kotor atau rusak. Menurut Denso (2011), ada 4 tingkatan koreksi kesalahan dalam *Qr Code*:

Tabel 1 Level Koreksi

Level Koreksi kesalahan	Jumlah Perkiraan
L	7%
M	15%
Q	25%
H	30%

Semakin tinggi tingkat koreksi kesalahan semakin besar juga versi *Qr Code*. Faktor lokasi dan lingkungan operasi perlu di timbangkan dalam menentukan level *Qr Code*. Level Q dan H baik digunakan di pabrik yang kotor, sedangkan L untuk tempat yang bersih. Level yang sering digunakana dalah level M dengan perkiraan koreksi mencapai 15% (qrcode.com, 2013).

2.1.4 Manfaat Qr Code

Beberapa manfaat yang terdapat pada *Qr Code* menurut Denso (2011) antara lain:

1. Kapasitas tinggi dalam menyimpan data
Sebuah *Qr Code* tunggal dapat menyimpan sampai 7.089 angka.
2. Ukuran yang kecil
Sebuah *Qr Code* dapat menyimpan jumlah data yang sama dengan *barcode ID* dan tidak memerlukan ruang besar.
3. Dapat mengoreksi kesalahan
Tergantung pada tingkat koreksi kesalahan yang dipilih, data pada *Qr Code* yang kotor atau rusak sampai 30% dapat diterjemahkan dengan baik.
4. Banyak jenis data
Qr Code dapat menangani angka, abjad, simbol, karakter bahasa Jepang, Cina atau Korea dan data biner.
5. Kompensasi distorsi
Qr Code tetap dapat dibaca pada permukaan melengkung atau terdistorsi.
6. Kemampuan menghubungkan

Sebuah *QR Code* dapat dibagi hingga 16 simbol yang lebih kecil agar sesuai dengan ruang. Simbol-simbol kecil yang dibaca sebagai kode tunggal apabila di scan menurut urutan.

2.1.5 Macam-macam Qr Code

a) *QR Code* model 1 dan model 2

- *QR Code* model 1



QR Code Model 1

Gambar 4 Contoh *QR Code* Model 1
(Sumber: qrcode.com)

Model 1 adalah *QR Code* asli, dapat menampung 1.167 angka dengan versi maksimum 14 (73 x 73 modul) (qrcode.com, 2013).

- *QR Code* model 2

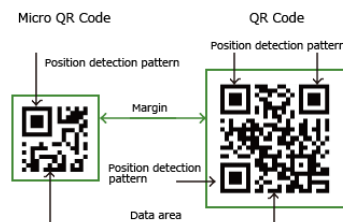


QR Code Model 2

Gambar 5 Contoh *QR Code* Model 2
(Sumber: qrcode.com)

Model 2 adalah penyempurnaan dari model 1 dengan versi terbesar 40 (177 x 177 modules), yang mampu menyimpan sampai 7.089 angka (qrcode.com, 2013).

b) *Micro QR Code*



Gambar 6 Contoh *Micro QR Code*
(Sumber: qrcode.com)

Versi terbesar dari kode ini adalah M4 (17x17 modul) yang dapat menyimpan hingga 35 angka. Fitur utama dari *Micro QR Code* adalah hanya memiliki satu pola deteksi posisi, dibandingkan dengan regular *QR Code* yang memerlukan sejumlah tempat karena pola deteksi posisi yang terletak di tiga sudut simbol. *QR Code* biasa membutuhkan setidaknya empat modul yang lebar di sekitar simbol, sedangkan *Micro QR Code* hanya membutuhkan cukup dua modul margin. Konfigurasi *Micro QR*

Code memungkinkan pencetakan di tempat lebih kecil dari *Qr Code* (qrcode.com,2013).

c) *iQr Code*



Gambar 7 Contoh *iQr Code*
(Sumber: qrcode.com)

Kode yang dapat dihasilkan dari salah satu modul, persegi atau persegi panjang. Dan dapat di cetak sebagai kode inversi hitam putih atau kode pola *dot* (bagian penanda). Versi terbesar dari kode ini dapat mencapai 61 (422x422 modul), yang dapat menyimpan 40.000 angka (qrcode.com, 2013).

d) *SQRC*



Gambar 8 Contoh *SQRC* (Sumber: qrcode.com)

Jenis *Qr Code* ini dilengkapi dengan membaca fungsi pembatas. Ini dapat digunakan untuk menyimpan informasi pribadi untuk mengelola informasi internal perusahaan dan sejenisnya (qrcode.com, 2013).

e) *LogoQ*



Gambar 9 Contoh *LogoQ*

Jenis *Qr Code* yang dapat menggabungkan fitur desain tingkat tinggi seperti ilustrasi, huruf dan logo. *Qr Code* ini menggunakan Logika *Since proprietary* [qrcode.com,2013].

2.2 Sandi Vigenere

Sandi Vigenere merupakan bagian dari kriptografi. kriptografi adalah kajian mengenai teknik matematika yang digunakan untuk keamanan informasi. Sandi Vigenere menurut Bruen & Forcinito (2011) adalah sebuah sandi klasik yang memiliki konsep yang relatif sederhana dan banyak digunakan sampai sekarang. Ide dari sandi Vigenere ini adalah sandi Caesar yang dimodifikasi. Jika sandi Caesar menggunakan kata kunci sandi tunggal,

sedangkan sandi vigenere menggunakan kata kunci yang diulang sebanyak yang diperlukan dengan panjang pesan. Huruf yang akan disandikan di sesuaikan dengan angka, $a = 0, b = 1, c = 2, \dots, z = 25$. Kemudian tambahkan angka kata kunci dan angka pesan. Lalu hasilnya dimodulukan dengan 26, dan hasilnya yang berupa angka tersebut dirubah ke dalam huruf untuk mendapatkan huruf yang tersandi. Sandi Vigenere merupakan sebuah algoritma kriptografi klasik, algoritma ini tergolong algoritma dasar karena menggunakan algoritma berbasis karakter [Kurniawan, 2012].

2.3 Android

Android adalah sistem operasi mobile berbasis *open source* yang di miliki raksasa internet saat ini, Google. Android dikembangkan dengan menggunakan kernel linux. Android memungkinkan untuk di modifikasi secara bebas dan di distibusikan oleh pembuat perangkat tersebut. Dengan sifat open source tersebut telah banyak mendorong komunitas pengembang aplikasi untuk menggunakan *source code* android sebagai dasar proyek pembuatan aplikasi.

Android dimulai sebagai sebuah start up rahasia pada tahun 2003, dan dibeli oleh Google pada tahun 2005 dan sebagai jalan google untuk memasuki pasar perangkat lunak bergerak. *Handphone* komersil pertama yang menggunakan S Android adalah HTC Dream, yang diluncurkan pada 22 Oktober 2008. Dikutip dari okezone.com (2013), terungkap pula sebanyak 4,5 juta smartphone yang berhasil terjual di Indonesia selama Januari sampai Maret 2013, sebanyak 2,28 juta di antaranya menjalankan OS Android.

2.2.1 Arsitektur android

Android sebagai sistem operasi memiliki beberapa layer komponen di antaranya (Asmono, 2013), adalah:

1. Linux kernel

Linux kernel merupakan layer dasar dari sistem operasi android. Di dalam layer ini berisi file-file sistem seperti *system processing, memory, resource, drivers* dan sistem lainnya.

2. Libraries

Pada layer ini terdapat fitur-fitur android berada, sering diakses untuk menjalankan aplikasi. Beberapa libraries yang terdapat pada android di antaranya adalah:

- a. Libraries media untuk memutar video dan audio.
- b. Libraries manajemen tampilan.
- c. Libraries Graphics
- d. Libraries SQLite digunakan untuk dukungan database.
- e. Libraries SSL dan WebKit
- f. Libraries LiveWebcore
- g. Libraries 3d

3. *Android runtime*

Layer ini berfungsi untuk menjalankan aplikasi di android. Proses dalam menjalankan aplikasi menggunakan implementasi dari Linux. Android RunTime dibagi dua bagian, yaitu:

a. *Core Libraries*

Berfungsi untuk menerjemahkan bahasa java atau bahasa C.

b. *Dalvik Virtual machine*

Merupakan sebuah mesin berbasis register untuk menjalankan fungsi-fungsi pada android secara efisien.

4. *Applications framework*

Layer ini di peruntukan bagi para pembuat aplikasi untuk menggunakan komponen-komponen yang terdapat pada layer ini untuk membuat aplikasi mereka. Komponen-komponen yang termasuk di dalam layer ini antara lain:

a. *Views*

b. *Content Providers*

c. *Resource Manager*

d. *Notification Manager*

e. *Activity Manager*

5. *Applications and Widget*

Layer ini berhubungan dengan aplikasi inti yang berjalan ada sistem operasi android.

2.2.2 Dasar aplikasi android (Android Application Fundamentals)

Aplikasi android ditulis dengan menggunakan bahasa pemrograman java. Untuk meng*compile* kode-kode tersebut Android menggunakan android SDK (Software development Kit) tool bersama dengan data dan files resource dan di pakatkan menjadi satu dan file hasil compile tersebut diberi akhiran .APK. APK ini dapat digunakan untuk menginstal aplikasi yang telah dibuat di sistem operasi android.

2.2.3 Komponen aplikasi android

Komponen aplikasi android adalah bagian penting dari sebuah aplikasi android. Setiap komponen mempunyai peranan yang berbeda sesuai dengan keperluan aplikasi. Ada 4 jenis komponen aplikasi android (Asmono, 2013) yang memiliki peranan atau tujuan yang berbeda di antaranya adalah:

1. *Activities*

Activities biasanya untuk menyajikan *User Interface* (UI) kepada user untuk melakukan interaksi. Untuk pindah dari satu *activity* ke *activity* lain dapat melakukannya dengan satu *even*, misalnya klik tombol, atau memilih opsi.

2. *Services*

Services merupakan komponen yang berjalan di *background* saat melakukan pekerjaannya. *Service* tidak memiliki *user interface*. Contohnya pada layanan pemutar musik yang dapat menjalankan satu memutar lagu di *background* saat pengguna menjalankan aplikasi yang berbeda.

3. *Content providers*
Antarmuka yang digunakan untuk berbagi data antar aplikasi. Dengan *Content provider programmer* dapat menyimpan data dalam sistem file, SQLite dan lain- lain. Sebagai contoh android menyediakan aplikasi kontak pengguna. Dengan demikian aplikasi apapun yang memerlukan kontak pengguna atas izin pengguna dapat mengaksesnya.
4. *Broadcast receivers*
Komponen yang merespon pengumuman ke user dari sistem android. Ini berfungsi untuk mengingatkan *user* itu sendiri. Misalnya baterai lemah, pesan masuk, telepon seluler telah dimatikan atau lain sebagainya. *Broadcast receivers* merespon dengan menghidupkan lampu latar, *led light*, getaran, suara dan lain- lain.

3. Metode Penelitian

3.1 Prosedur Penelitian

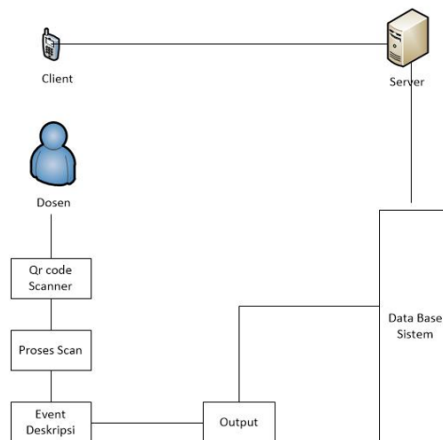
Deskripsi umum mengenai metode penelitian ini diperlihatkan pada gambar 10 berikut:



Gambar 10 Tahapan Penelitian

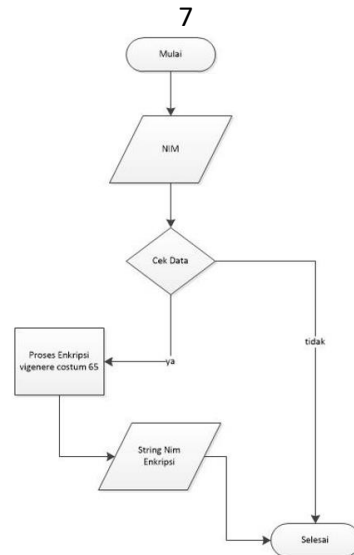
3.2 Arsitektur Sistem

Arsitektur sistem digunakan untuk menerjemahkan bagian-bagian dari keseluruhan sistem yang lebih bersifat khusus secara terstruktur dan dengan bertujuan yang menjawab kebutuhan sistem.



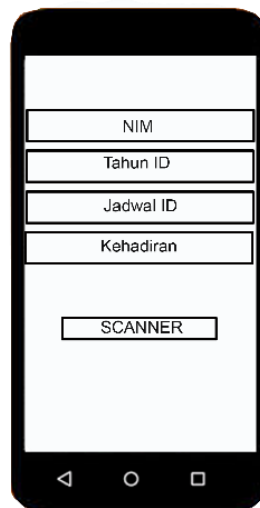
Gambar 11 Arsitektur Sistem pada Platform Android

Pada proses enkripsi ini, data yang akan dienkripsi adalah berupa NIM Mahasiswa. Jadi, nantinya NIM tersebut akan diekripsi kedalam *vigenere cipher*. Pertama, data nim akan dicek apakah ada atau tidak. Jika NIM ada maka akan dilanjutkan kedalam proses penyandian *vigenere cipher* sampai mengeluarkan string enkripsi (Gambar 12).



Gambar 12 Proses Enkripsi

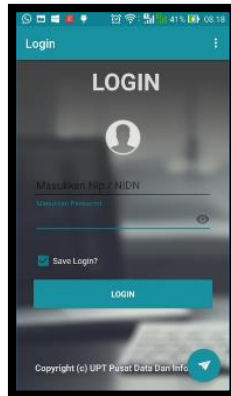
Data kemudian dilanjutkan pada proses scan Qr Code. Scan Qr Code ini dilakukan oleh dosen penguji/penjaga saat ujian akhir semester/ujian tengah semester. Proses dilakukan oleh dosen yaitu dengan set data terlebih dahulu seperti: pilih mata kuliah, jam kuliah, hari, dan penguji. Setelah data di set, kemudian data dilanjutkan pada scan Qr Code dan dikirim kedalam database.



Gambar 13. Desain Awal Scan Absensi

4. Hasil dan Pembahasan

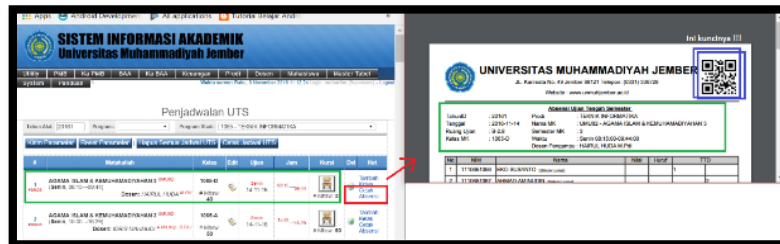
4.1 Hasil Pembangunan Aplikasi



Gambar 14. Tampilan Login

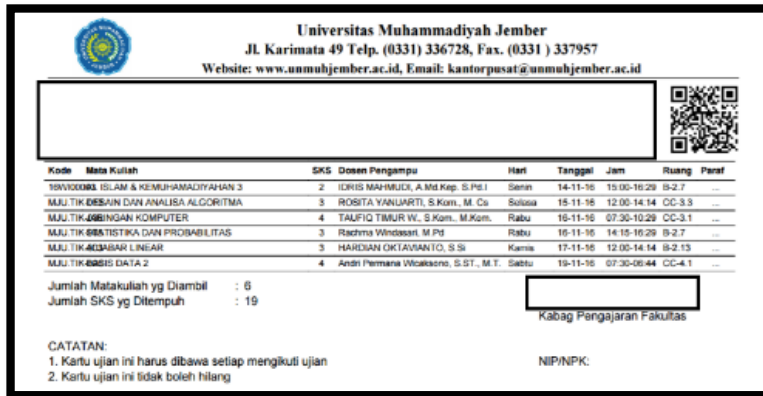
Form ini digunakan untuk dosen pengawas login. Sistem ini terintegrasi dengan sistem informasi akademik di Universitas Muhammadiyah Jember, sehingga *user* dan *password* menggunakan yang telah ada di sistem informasi akademik.

Langkah berikutnya adalah dosen pengawas men-*scan* lembar kehadiran yang dicetak dari sistem akademik, untuk menentukan mata kuliah dan prodi yang menyelenggarakan (Gambar 14).



Gambar 15 Absensi Ujian dari SIA dan QR-Code dalamnya

Berikutnya, dosen pengawas tinggal melakukan scan terhadap kartu ujian mahasiswa. Scan ini dapat dilakukan dengan 2 cara, menscan kartu yang dicetak kertas atau men-*scan* dari HP mahasiswa yang telah diinstall aplikasi SIA Andorid (Gambar 16 dan 17).



Gambar 16 Kartu Ujian Kertas



Gambar 17 Kartu Ujian Digital

4.2 Perbandingan Sistem

Integrasi Pada Sistem Informasi Akademik (SIA) ini dikembangkan untuk mempermudah pengguna/administrator ataupun dosen penguji dalam melakukan presensi. Data pengembangan sistem dan perbedaan sistem lama dengan sistem baru dapat dilihat pada tabel 2.

Tabel 2 Perbandingan Sistem.

Sistem Lama	Sistem Baru
- Absensi masih menggunakan manual dan tidak terkoneksi ke Sistem Informasi Akademik (SIA)	- Absensi yang dilakukan secara online dan terkoneksi ke Sistem Informasi Akademik (SIA)
- Kemungkinan kecurangan mahasiswa dalam mengikuti ujian masih ada	- Kecurangan mahasiswa dalam mengikuti ujian (bayar / belum bayar kuliah) dapat diatasi dengan menggunakan aplikasi android

4.3 Pengujian Sistem

Pengujian alpha atau *black box* merupakan metode pengujian yang berfokus pada kebutuhan fungsional dari aplikasi. Pengujian black box dilakukan dengan fokus pada hasil keluaran yang diharapkan dari sistem yang diuji, apakah dapat berjalan sesuai yang diharapkan atau tidak. Tabel pengujian *black box* dapat dilihat pada Tabel 3.

Tabel 3 Pengujian Black Box

No.	Skenario Pengujian	Hasil Yang Diharapkan	Hasil Pengujian
1	<i>Scanning</i> pada <i>Qr Code</i> di kartu ujian	Aplikasi masuk ke halaman <i>scanning</i> .	Berhasil
2	Mendapatkan data dari hasil <i>scanning</i>	Aplikasi menampilkan data mahasiswa	Berhasil
3	Deskripsi <i>Qr Code</i> melalui aplikasi android	Deskripsi berupa nim asli	Berhasil
4	Mahasiswa mencetak kartu ujian yang telah tersisipi kode <i>Qr Code</i>	Kartu Ujian telah tersisipi otomatis kode <i>Qr Code</i>	Berhasil
5	Data yang discan masuk kedalam data base server	Aplikasi menampilkan data yang telah terkoneksi ke data base server	Berhasil

5. Kesimpulan dan Saran

Dari hasil pengujian dapat disimpulkan beberapa hal sebagai berikut:

1. Implementasi pada integrasi Sistem Informasi Akademik (SIA), dapat menghubungkan data yang dikirim dari Sistem Absensi menggunakan *QR Code Scanner*.
2. Sistem yang dikembangkan dapat mempermudah dosen dalam proses absensi Ujian Tengah Semester (UTS) ataupun Ujian Akhir Semester (UAS).
3. Sistem yang dikembangkan dapat menguji kecurangan mahasiswa dalam mengikuti ujian, dikarenakan pada proses *Scanner* ada notifikasi bahwa mahasiswa tersebut telah membayar ataupun belum.

Setelah dilakukan pembangunan dan pengujian sistem diberikan saran pengembangan sebagai berikut:

1. Diterapkan sistem enkripsi pada nomer ujian dan nim mahasiswa yang tercetak pada QR-Code
2. Pada layout data set masih menggunakan menu tab, disarankan menggunakan tab *layout* dan *swipe*.

DAFTAR PUSTAKA

- [1] Android. 2013. Dashboards. *Developer.android.com*, (diakses 15 Januari 2016).
- [2] Ariadi. (2011). *Analisis dan Perancangan Kode Matriks Dua Dimensi Quick Response (QR) Code*. Skripsi. Universitas Sumatera Utara.
- [3] Ashford, Robin. 2010. *QR Code and academic libraries eaching mobile users*. (Online) <http://crln.acrl.org/content/71/10/526.full> (22 januari 2016).
- [4] Asmono, Cahya Rizky D. (2013). *Perancangan Aplikasi Hijaiyah Pada Android Dengan Menggunakan Metode Rectangles Collision Detection*. Skripsi. Universitas Sumatera Utara.
- [5] Denso Wave Incorporated. 2013. Answers to your question about the QR Code. (Online) <http://www.qrcode.com/en/> (20 februari 2016).
- [6] Denso ADC. (2011). QR Code Essentials (Online), <http://www.nacs.org/LinkClick.aspx?fileticket=D1FpVAvvJuo%3D&tabid=1426&mid=4802>(5januari2016)
- [7] Setyawan, Antonius Hendra. (2010). *Perancangan Aplikasi Sistem Presensi Mahasiswa Menggunakan Qr Code Pada Sistem Operasi Android*. Diponegoro: Universitas diponegoro.