

PERANCANGAN APLIKASI SISTEM PRESENSI MAHASISWA TEKNIK ELEKTRO UNIVERSITAS DIPONEGORO MENGGUNAKAN *NEAR FIELD COMMUNICATION* PADA *SMARTPHONE* BERBASIS ANDROID

Geovany Rachman Wacono ^{*)}, Maman Somantri, and Enda W.S.

Departemen Teknik Elektro, Universitas Diponegoro Semarang
Jl. Prof. Sudharto, SH, Kampus UNDIP Tembalang, Semarang 50275, Indonesia

^{*)} *E-mail: geovany.geov@gmail.com*

Abstrak

Sistem presensi memiliki peran penting dalam kehidupan sehari-hari. Dalam kegiatan belajar mengajar di Universitas Diponegoro, Departemen Teknik Elektro masih menerapkan sistem presensi manual. Selama ini, pengisian daftar hadir mahasiswa masih dilakukan dengan tanda tangan. Melihat perkembangan teknologi yang pesat, tidak menutup kemungkinan bahwa ponsel pintar dapat dimanfaatkan sebagai sistem presensi di perguruan tinggi. Dari berbagai ponsel pintar yang populer saat ini, beberapa diantaranya mengukung sistem operasi Android. Salah satu fitur yang ditanamkan ke dalam ponsel pintar Android adalah Near Field Communication (NFC). Pada penelitian ini, penulis merancang aplikasi sistem presensi menggunakan NFC pada ponsel pintar Android. Aplikasi ini memudahkan mahasiswa dalam mengisi kehadiran, yaitu dengan menempelkan tag mahasiswa ke ponsel pintar Android. Hasil pengujian komunikasi data menunjukkan bahwa waktu yang diperlukan dalam mencatat kehadiran mahasiswa sebesar 2208,6 ms. Disamping itu, aplikasi juga dapat menampilkan daftar presensi dosen, daftar peserta kuliah beserta jumlahnya, dan persentase kehadiran mahasiswa, sehingga aplikasi ini dinilai lebih efektif dibandingkan dengan sistem presensi manual.

Kata Kunci: Android, NFC, sistem presensi.

Abstract

Presence system has an important role in daily life. In college activity of Electrical Engineering Department, Diponegoro University is still using manual presence system. So far the students are still using their signature for attendance list. Nowadays, technology has developed rapidly, so it is possible if smartphone can be used as presence system of college activity. There are so many popular varieties of smartphone operating system, One of them is Android. One of features which applied in Android smartphone is Near Field Communication (NFC). In this research, the authors designed a presence system applications by using NFC on Android smartphones. This application works by attaching the student tag to Android smartphone, so it will be easier and more simple than manual presence system. The test of data communication results showed that the average time needed to record student attendance at 2208,6 ms. Beside that, the application also can display a list of lecturers presence, list of participants and their college number and percentage of student attendance, so it is more effective than the manual system presence.

Keyword: Android, NFC, presence system.

1. Pendahuluan

Sistem presensi memiliki peran penting dalam kehidupan sehari-hari terutama di ruang lingkup kerja salah satunya perkuliahan. Dalam kegiatan belajar mengajar di suatu perguruan tinggi, tentu saja terdapat mahasiswa yang perlu dicatat kehadirannya setiap ada kegiatan perkuliahan. Pencatatan kehadiran ini lebih sering dikenal sebagai presensi. Presensi adalah salah satu faktor penting dalam dunia perkuliahan.

Masalah yang terjadi dalam sistem yang telah diterapkan terletak pada pengumpulan data hadir mahasiswa yang dilakukan dalam kelas. Sistem presensi yang berjalan saat ini masih dilakukan secara manual, yaitu dengan pengisian tanda tangan secara langsung oleh mahasiswa. Pengumpulan data secara manual ini dirasa masih memiliki kelemahan sehingga tidak sedikit kecurangan terjadi dalam pengisian presensi. Pada kenyataannya, beberapa departemen di Universitas Diponegoro masih menerapkan sistem presensi secara manual salah satunya adalah Departemen Teknik Elektro.

Melihat perkembangan teknologi yang pesat, tidak menutup kemungkinan bahwa *smartphone* dapat dimanfaatkan sebagai sistem presensi di perguruan tinggi. Dari sekian banyak *smartphone* yang bermunculan, kebanyakan dari mereka mengusung sistem operasi android. Seiring berkembangnya teknologi, banyak *smartphone* bermunculan dengan menawarkan fitur-fitur yang lebih berinovasi. Salah satu fitur *smartphone* saat ini adalah adanya teknologi *Near Field Communication* (NFC) yang diadaptasi dari teknologi *Radio Frequency Identification* (RFID) [1]. Teknologi ini memungkinkan pengguna *smartphone* untuk saling bertukar data dengan *tag* pasif maupun pengguna *smartphone* lainnya pada jarak yang dekat [2].

Penelitian mengenai sistem presensi mahasiswa pada dasarnya telah dikembangkan dengan berbagai metode salah satunya menggunakan *QR Code* pada sistem operasi Android [3]. Pada penelitian tersebut, perancangan aplikasi hanya dikhususkan untuk pengguna dosen saja. Desain tampilan dan fitur yang dimiliki aplikasi tersebut masih sangat sederhana, sebatas mengisi dan melihat rekap kehadiran mahasiswa. Dari sisi media penyimpanan, aplikasi masih menggunakan media penyimpanan lokal yaitu *SQLite* yang terdapat dalam android, sehingga seluruh data yang masuk akan tersimpan ke dalam memori *smartphone*. Disamping itu, penerapan *QR Code* pada sistem presensi masih memiliki kekurangan dalam hal keamanan data, yaitu *QR Code* dapat dibaca oleh semua *smartphone* yang mendukung fitur kamera dan dengan mudah digandakan menggunakan printer dan kertas biasa [4]. Berbeda dengan *QR Code*, NFC dapat menerapkan MIME (*Multipurpose Internet Mail Extension*) ke dalam *NFC tag* sehingga data yang tersimpan dalam *tag* hanya dapat dibaca oleh aplikasi yang memiliki tipe MIME yang sama [5].

Berdasarkan permasalahan tersebut, maka muncul ide untuk memanfaatkan NFC pada *smartphone* android sebagai sistem presensi mahasiswa. Dengan memanfaatkan teknologi ini, diharapkan dapat menutupi kelemahan yang ada pada sistem presensi saat ini.

2. Metode

2.1. Metodologi Penelitian

Tahapan pelaksanaan penelitian untuk menyelesaikan Tugas Akhir ini adalah sebagai berikut:

1. Studi Literatur

Studi literatur dilakukan dengan mempelajari permasalahan yang akan dikemukakan pada Tugas Akhir melalui beberapa buku literatur, dan menganalisa data menggunakan tulisan yang berhubungan dengan sistem yang dirancang, baik dari perpustakaan, artikel, maupun internet.

2. Perancangan Sistem

Perancangan sistem dilakukan menggunakan metode *Waterfall* dengan tahapan metode berikut [6]:

- Melakukan identifikasi terhadap kebutuhan sistem berdasarkan studi literatur.
- Menganalisis kebutuhan sistem kemudian menggambarkannya dalam bentuk definisi kebutuhan spesifik.
- Menggunakan definisi kebutuhan untuk mendesain sistem dengan memetakan kebutuhan sistem kedalam bentuk desain sistem termasuk perancangan gambaran aplikasi dan relasi di dalam sistem.
- Mengkodekan sistem menjadi aplikasi yang dapat dijalankan oleh *smartphone* berdasarkan desain sistem.
- Menguji aplikasi yang telah dibuat untuk memastikan aplikasi telah sesuai dengan definisi kebutuhan dan berfungsi dengan benar. Pengkodean ulang dilakukan pada saat aplikasi tidak berjalan seperti seharusnya. Pengujian Sistem

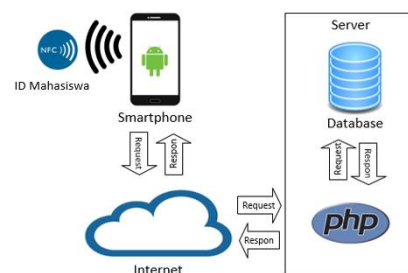
3. Pengujian yang dilakukan pada sistem meliputi:

- Pengujian *white box* pada aplikasi Sistem Presensi berbasis android, dilakukan dengan menggunakan metode *basis path* untuk menguji alur logika pada fitur inti aplikasi Sistem Presensi berbasis android.
- Pengujian komunikasi data, dilakukan dengan mengimplementasikan alamat tujuan dan model pengiriman pada aplikasi *hurl.it*.
- Pengujian komparabilitas perangkat keras, dilakukan dengan menguji secara langsung kepada beberapa tipe ponsel pintar.

2.2. Analisis Kebutuhan

2.2.1. Deskripsi Sistem

Dalam penelitian ini, penulis membuat sebuah sistem presensi mahasiswa di Departemen Teknik Elektro Universitas Diponegoro menggunakan *Near Field Communication* pada perangkat bergerak berbasis android. Konsep yang dibahas adalah bagaimana aplikasi *client* yang tertanam pada perangkat android dapat melakukan pertukaran data dengan *NFC tag* serta berkomunikasi dengan *database server* dengan menggunakan *web service* yang berformat JSON.



Gambar 1. Desain client – server sistem presensi

Pada Gambar 1 menunjukkan bahwa terjadi pertukaran data antara sisi *client* android dengan sisi server. Sisi *client* android membaca ID mahasiswa kemudian melakukan *request* ke *database* server. Untuk mengirim data ke sisi klien android, digunakan mesin penghubung yang berformat JSON. Mesin penghubung melakukan *request* data ke *database server* lalu *database server* memberikan balikan data kepada mesin penghubung berupa data *array* berformat JSON. Data *array* berformat JSON inilah nantinya yang dapat dibaca oleh perangkat android (*client*).

Dalam Sistem Presensi ini terdapat dua bagian inti yang dapat mengintegrasikan android dengan *server* yaitu aplikasi Sistem Presensi (Java) dan *server* yang terdiri dari basis data dan file-file penghubung yang dibuat menggunakan bahasa pemrograman PHP. Hubungan Antara perangkat android (klien) dan *server* dapat dilihat pada Gambar 1.

2.2.2. Kebutuhan Fungsional

Kebutuhan fungsional merupakan gambaran mengenai fungsi-fungsi yang dapat dilakukan oleh sistem ini. Aplikasi yang dirancang adalah sebuah sistem presensi pada perangkat bergerak berbasis android. Kebutuhan fungsional sistem meliputi:

- a. Sistem yang dirancang memiliki tiga jenis hak akses kepada pengguna yaitu admin, akademik, dan dosen.
- b. Pengguna admin memiliki akses untuk mengelola seluruh data *user* dan data mahasiswa.
- c. Pengguna akademik memiliki akses untuk mengelola jadwal kuliah dan mengelola presensi dosen maupun mahasiswa.
- d. Pengguna dosen memiliki akses untuk mengisi presensi dosen dan presensi mahasiswa. Pengguna dosen juga diberi fasilitas dalam melihat jadwal kuliah masing-masing dosen, melihat daftar hadir dosen dan mahasiswa setiap mata kuliah, dan melihat daftar peserta kuliah.
- e. Seluruh pengguna diberi fitur untuk mengubah *password*.

2.2.3. Kebutuhan Non Fungsional

Kebutuhan non-fungsional adalah kebutuhan sistem meliputi kinerja, kelengkapan operasi pada fungsi-fungsi yang ada, serta kesesuaian dengan lingkungan penggunaannya. Kebutuhan non-fungsional ini melingkupi beberapa kebutuhan yang mendukung kebutuhan fungsional, rumusan kebutuhan non-fungsional meliputi:

- a. Kebutuhan Keamanan
 - Halaman admin hanya dapat diakses oleh pengguna yang memiliki hak akses admin.
 - Halaman akademik hanya dapat diakses oleh pengguna yang memiliki hak akses akademik.
 - Halaman dosen hanya dapat diakses oleh pengguna yang memiliki hak akses dosen.

- Menggunakan mekanisme enkripsi *password MD5* sebagai sistem keamanan (verifikasi pengguna).
 - Satu pengguna hanya dapat *login* pada satu *device* saja.
 - *NFC tag* yang telah ditulis akan dikunci permanen sehingga tidak dapat ditulis ulang.
 - Aplikasi hanya dapat membaca *NFC tag* dengan tipe MIME yang sesuai pada aplikasi.
- b. Kebutuhan Operasional
 - Aplikasi klien dapat didistribusikan melalui internet dan harus dapat diinstal pada perangkat android.
 - Aplikasi ini dibangun dengan menggunakan bahasa pemrograman Java.
 - Mahasiswa hanya dapat mengisi presensi kuliah saat dalam sesi kuliah dilaksanakan.
 - Mahasiswa yang telah mengisi presensi kuliah tidak dapat mengisi presensi lagi sebelum sesi kuliah berakhir.
 - Dosen tidak dapat membuka halaman pengisian presensi mahasiswa setelah sesi kuliah berakhir.
 - Sistem ini menggunakan pertukaran data dengan format JSON.

2.2.4. Kebutuhan Perangkat

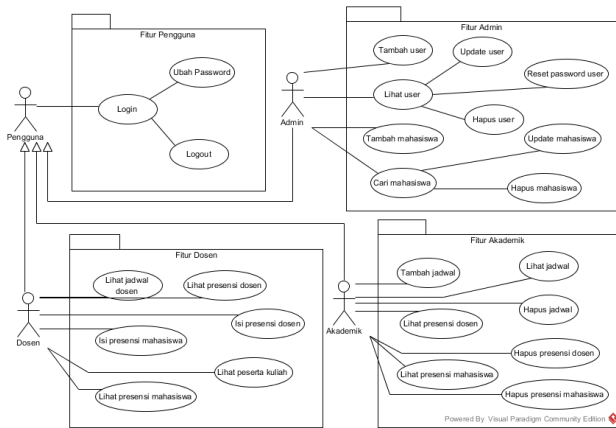
Dalam pembuatan aplikasi ini, dibutuhkan suatu perangkat keras dan perangkat lunak. Kebutuhan perangkat keras meliputi *Notebook* Sony Vaio VPCS137GG dengan spesifikasi Intel(R) Core(TM) i5-560M CPU @ 2.66GHz, memori sebesar 4096MB, dan kapasitas *harddisk* sebesar 432GB. Pada perangkat lunak dibutuhkan Android Studio 1.5.1 sebagai IDE dalam penulisan kode aplikasi sistem presensi, dan Visual Paradigm v.13.0 untuk desain pemodelan objek.

2.3. Desain Aplikasi

2.3.1. Diagram Use Case

Diagram *use case* menggambarkan fungsi-fungsi yang ada pada sistem. Diagram ini lebih berfokus pada fitur-fitur sistem dari sudut pandang pihak luar, yang dalam hal ini adalah pengguna aplikasi [7]. Gambar 2 merupakan diagram *use case* aplikasi sistem presensi berbasis android.

Pada Gambar 2 terlihat bahwa fitur aplikasi dilihat dari sudut pandang pengguna dapat dibagi menjadi fitur pengguna, admin, akademik, dan dosen. Selain itu, terdapat empat macam aktor yaitu pengguna, admin, akademik, dan dosen. Aktor pengguna merupakan aktor induk dari aktor-aktor lainnya. Semua hal yang dapat dilakukan aktor induk, dapat pula dilakukan oleh aktor anak.

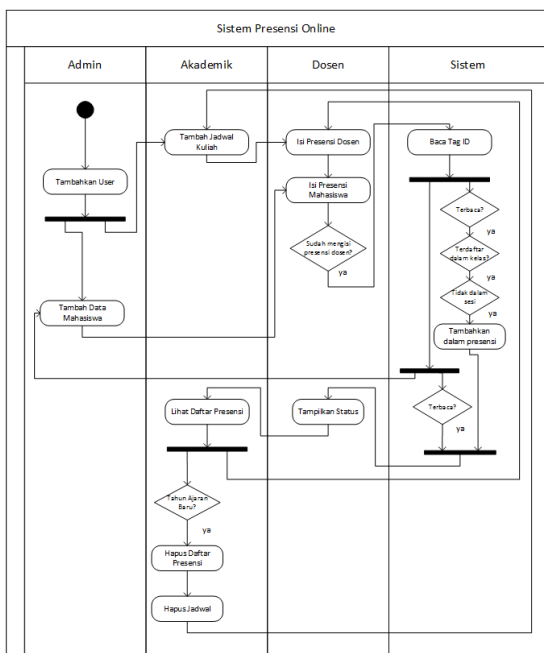


Gambar 2. Diagram use case aplikasi sistem presensi

2.3.2. Diagram Aktivitas

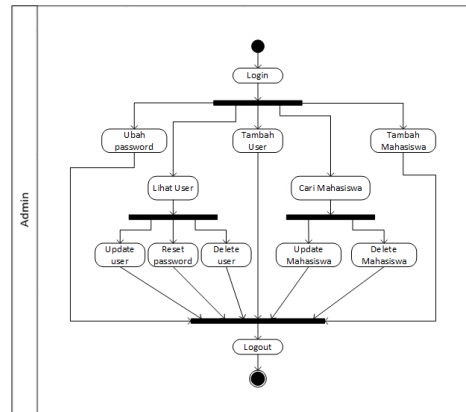
Diagram aktivitas menggambarkan aliran proses suatu perilaku atau aktivitas entitas yang ada di dalam sistem. Entitas dalam diagram aktivitas dapat berupa pengguna atau pun sistem itu sendiri[7].

Diagram aktivitas kegiatan inti pada aplikasi sistem presensi dapat dilihat pada Gambar 3. Kegiatan inti pada aplikasi meliputi penambahan *user*, penambahan mahasiswa, pengisian jadwal kuliah, pengisian kehadiran dosen, penginputan kehadiran mahasiswa, melihat presensi dosen, dan melihat presensi mahasiswa.



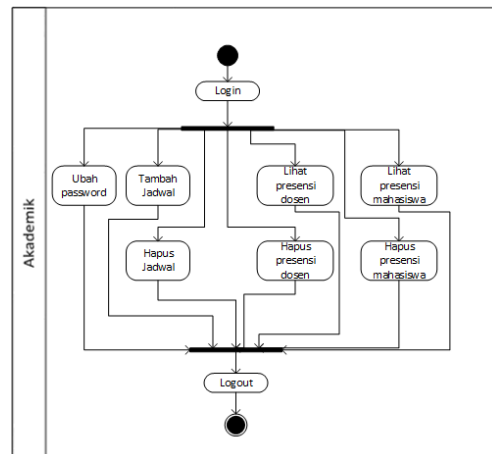
Gambar 3. Diagram aktivitas kegiatan inti

Pada Gambar 4 dapat dilihat diagram aktivitas pengguna berjenis admin dari saat login ke aplikasi, mengakses fitur-fitur admin, sampai logout dari aplikasi.



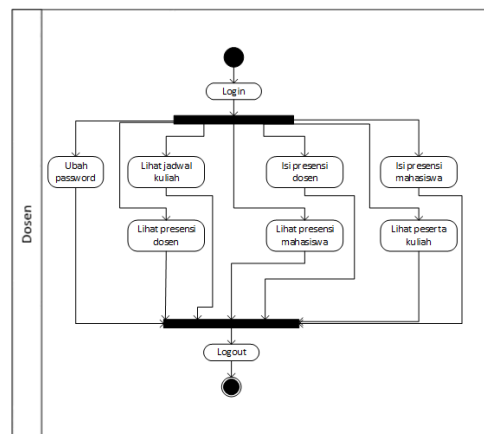
Gambar 4. Diagram aktivitas admin

Pada Gambar 5 dapat dilihat diagram aktivitas pengguna berjenis akademik dari saat login ke aplikasi, mengakses fitur-fitur akademik, sampai logout dari aplikasi.



Gambar 5. Diagram aktivitas akademik

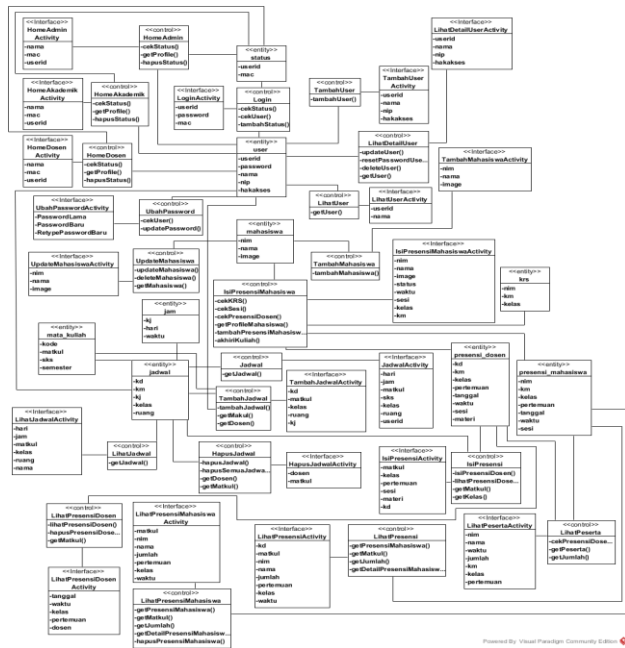
Pada Gambar 6 dapat dilihat diagram aktivitas pengguna berjenis dosen dari saat login ke aplikasi, mengakses fitur-fitur dosen, sampai logout dari aplikasi.



Gambar 6. Diagram aktivitas dosen

2.3.3. Diagram Kelas

Diagram kelas menggambarkan komponen-komponen pembentuk sistem dan hubungan antar komponen tersebut [7].

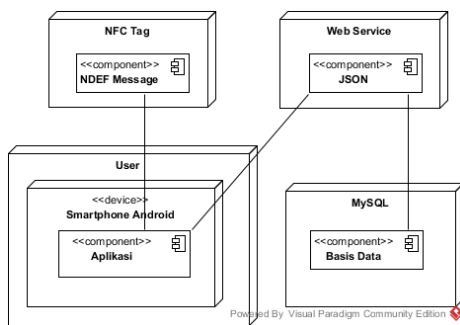


Gambar 7. Diagram kelas aplikasi sistem presensi

Pada Gambar 7 dapat dilihat bahwa diagram kelas sistem presensi dibagi menjadi kelas *interface*, *control*, dan *entity*. Kelas *interface* berfungsi sebagai kelas antarmuka pengguna. Kelas *control* berfungsi sebagai kelas yang mengatur komponen-komponen model dan *view*, sedangkan kelas *entity* yang merupakan komponen-komponen model.

2.3.4. Diagram Deployment

Diagram *deployment* menggambarkan arsitektur sistem yang dapat berupa konfigurasi komponen-komponen perangkat keras, atau konfigurasi komponen-komponen perangkat lunak [7]. Diagram *deployment* aplikasi sistem presensi dapat dilihat pada Gambar 8.



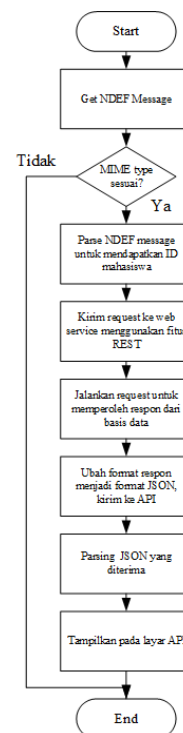
Gambar 8. Diagram deployment aplikasi sistem presensi

Secara garis besar, sistem terdiri dari NFC tag, user, web service, dan server. Pada sisi NFC tag terdapat NDEF message di dalamnya yang berisi informasi mengenai ID mahasiswa. User menggunakan fitur REST (Representation State Transfer) pada web service untuk melakukan request data dari user ke basis data MySQL melalui API (Application Program Interface). Data kemudian dikirimkan kembali ke sisi user berupa data array dengan format JSON sehingga dapat diakses oleh user dan dapat ditampilkan pada aplikasi android.

3. Hasil dan Analisa

3.1. Implementasi Sistem

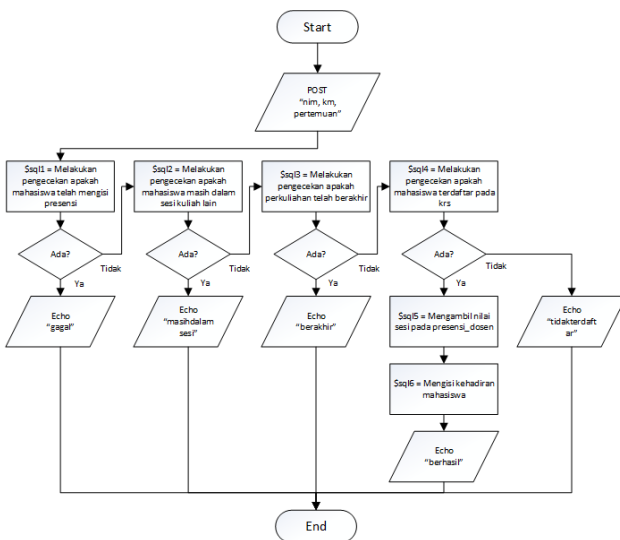
Berdasarkan hasil analisis sistem, penerapan teknologi NFC pada aplikasi sistem presensi adalah dalam pembacaan ID mahasiswa pada NFC tag. Pada saat NFC tagging, sistem android membaca NDEF message yang tersimpan pada NFC tag. Sistem membaca record yang berisi MIME (Multipurpose Internet Mail Extension) type ID mahasiswa, kemudian sistem mencocokkan dengan MIME type yang tertera pada aplikasi sistem presensi. Apabila MIME type pada NFC tag sesuai dengan MIME type pada aplikasi, maka aplikasi mengambil payload yang terdapat pada record selanjutnya, yang berisi ID mahasiswa. ID yang didapatkan akan dikirimkan ke web service. Web service menjalankan request untuk memperoleh respon dari basis data. Respon yang telah didapatkan akan dikirimkan kembali ke klien dalam format JSON agar dapat ditampilkan dalam aplikasi Sistem Presensi. Alur kerja penerapan NFC dalam proses pembacaan dapat dilihat pada Gambar 9.



Gambar 9. Proses pembacaan NFC tag

Tahap selanjutnya adalah tahap implementasi ketika analisa sistem diterapkan ke bentuk pengkodean, sehingga dapat diketahui apakah sistem yang telah dibuat sudah sesuai dengan perancangan. Salah satu senarai fitur inti program terdapat pada berkas isi_presensi_mahasiswa.php.

Pada senarai fitur inti program, terdapat beberapa query yang berbeda untuk mengakses basis data. Query ini memiliki fungsi untuk mengetahui status pengisian kehadiran mahasiswa. Respon yang dihasilkan oleh query berupa data string yang akan ditampilkan pada aplikasi sistem presensi. Gambar 10 menunjukkan proses aliran data seranai dari proses request hingga dapat menampilkan status kehadiran mahasiswa.



Gambar 10. Proses aliran data seranai

Berdasarkan Gambar 10 dapat diketahui bahwa respon yang akan muncul apabila kondisi pertama terpenuhi adalah “gagal”. Hal ini mengakibatkan aplikasi sistem presensi menampilkan status kehadiran seperti pada Gambar 11.



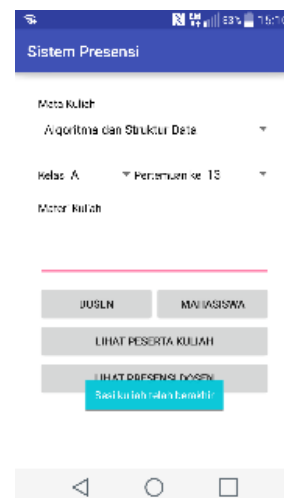
Gambar 11. Status kehadiran dengan respon “gagal”

Apabila kondisi pertama tidak terpenuhi maka sistem akan mencari kondisi berdasarkan query yang kedua. Apabila kondisi kedua terpenuhi, maka respon yang akan muncul adalah “masihdalamsesi”. Hal ini mengakibatkan aplikasi sistem presensi menampilkan status kehadiran seperti pada Gambar 12.



Gambar 12. Status kehadiran dengan respon “masihdalamsesi”

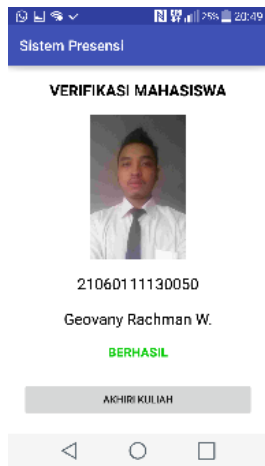
Apabila kondisi kedua tidak terpenuhi maka sistem akan mencari kondisi berdasarkan query yang ketiga. Apabila kondisi ketiga terpenuhi, maka respon yang akan muncul adalah “berakhir”. Hal ini mengakibatkan aplikasi sistem presensi menampilkan status kehadiran seperti pada Gambar 13.



Gambar 13. Status kehadiran dengan respon “berakhir”

Apabila kondisi ketiga tidak terpenuhi maka sistem akan mencari kondisi berdasarkan query yang keempat. Apabila kondisi keempat terpenuhi, maka sistem akan menambahkan record ke dalam basis data dan respon

yang akan muncul adalah “berhasil”. Hal ini mengakibatkan aplikasi sistem presensi menampilkan status kehadiran seperti pada Gambar 14.



Gambar 14. Status kehadiran dengan respon “berhasil”

Apabila kondisi keempat tidak terpenuhi, maka respon yang akan muncul adalah “tidakterdaftar”. Hal ini mengakibatkan aplikasi sistem presensi menampilkan status kehadiran seperti pada Gambar 15.



Gambar 15. Status kehadiran dengan respon “tidakterdaftar”

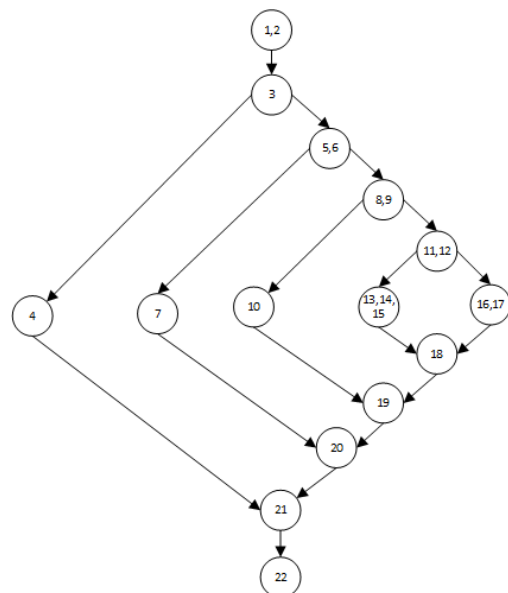
3.2. Pengujian Sistem

3.2.1. Pengujian White Box

Pengujian *white box* digunakan untuk meyakinkan semua perintah dan kondisi pada fitur utama aplikasi dieksekusi secara minimal. Pengujian *white box* menggunakan jenis pengujian *basis path* untuk mengetahui seluruh jalur independen pada suatu model telah digunakan minimal satu kali. Pengujian ini dilakukan pada fitur isi presensi mahasiswa untuk mengetahui apakah semua perintah pada proses pengisian presensi mahasiswa berjalan dengan baik. Berikut merupakan *pseudocode* yang digunakan.

1. Get Ndef Message from NFC tag
2. Post Request \$nim, \$km, \$kelas, \$pertemuan to web service
3. If nim=\$nim, km=\$km, pertemuan=\$pertemuan from presensi_mahasiswa = 1
4. Then echo “gagal”
5. Else
6. If nim=\$nim, tanggal=now(), sesi>=now() from presensi_mahasiswa = 1
7. Then echo “masihdalamsesi”
8. Else
9. If km=\$km, kelas=\$kelas, pertemuan=\$pertemuan, tanggal=now(), sesi<=now() from presensi_dosen = 1
10. Then echo “berakhir”
11. Else
12. If nim=\$nim, km=\$km, kelas=\$kelas from KRS = 1
13. Then get sesi from presensi_dosen
14. Set to presensi_mahasiswa (nim,km,kelas,pertemuan,waktu,tanggal,sesi) VALUES (\$nim,\$km,\$kelas,\$pertemuan,now(),now(),\$sesi)
15. echo “berhasil”
16. Else
17. echo “tidakterdaftar”
18. End if
19. End if
20. End if
21. End if
22. Show response

Setelah membuat *pseudocode* sebagai algoritma proses pengambilan data, dilakukan perubahan menjadi *flow graph*. Perubahan *pseudocode* menjadi *flow graph* dapat dilihat pada Gambar 16.



Gambar 16. Flow graph hasil pengubahan dari pseudocode

Pada Gambar 16 menggambarkan *flow graph* dengan lingkaran menggambarkan kondisi, sedangkan anak panah menggambarkan aksi. Dari Gambar 16 dapat dihitung *cyclomatic complexity* sebagai berikut:

$$V(G) = E - N + 2$$

$$V(G) = 17 - 14 + 2 = 5$$

Dimana E merupakan jumlah *edge* atau anak panah pada grafik alir, sedangkan N merupakan *node* atau lingkaran pada grafik alir. Dari perhitungan rumus tersebut didapatkan *cyclomatic complexity* bernilai 5. Berdasarkan hasil tersebut, maka dibutuhkan minimal 5 *path* untuk melewati semua *node*. Tabel 1 merupakan tabel berisi *path cyclomatic complexity*.

Tabel 1. *Path cyclomatic complexity*

Path	Aliran Proses
Path 1	1,2,3,4,21,22
Path 2	1,2,3,5,6,7,20,21,22
Path 3	1,2,3,5,6,8,9,10,19,20,21,22
Path 4	1,2,3,5,6,8,9,11,12,13,14,15,18,19,20,21,22
Path 5	1,2,3,5,6,8,9,11,12,16,17,18,19,20,21,22

Hasil pengujian *white-box* menunjukkan semua alur pengujian terlewati.

3.2.2. Pengujian Komunikasi Data

Pengujian komunikasi data merupakan pengujian untuk mengetahui hasil dari pertukaran data yang dilakukan *web service* berupa besar data yang dikirimkan dan durasi waktu pengiriman data. Gambar 16 adalah salah satu hasil pengujian komunikasi data dengan berkas *isi_presensi_mahasiswa.php*



Gambar 17. Hasil pengujian komunikasi data pada berkas *isi_presensi_mahasiswa.php*

Pada Gambar 17 terlihat bahwa pengujian komunikasi data telah menangkap balikan data yang dilakukan *web service* dengan durasi waktu menyelesaikan proses eksekusi pengujian adalah 1969 ms, serta besar data yang diterima berukuran 8 bytes. Setelah melakukan lima kali uji coba pada berkas *isi_presensi_mahasiswa.php*, maka diperoleh hasil seperti pada Tabel 2.

Tabel 2. Hasil pengujian komunikasi data pada berkas *isi_presensi_mahasiswa.php*.

Uji ke-	Respon	Besar Data (bytes)	Waktu Durasi (ms)
1	berhasil	8	1969
2	berhasil	8	2677
3	berhasil	8	2386
4	berhasil	8	1897
5	berhasil	8	2114
TOTAL		40	11043
RATA-RATA		8	2208,6

Pada Tabel 2 dapat dilihat bahwa setelah dilakukan lima kali uji coba, diperoleh rata-rata data yang diterima sebesar 8 bytes dengan rata-rata durasi waktu sebesar 2208,6 ms.

3.2.3. Pengujian Pada Perangkat Keras

Pengujian pada perangkat keras merupakan pengujian secara langsung kepada beberapa perangkat. Pengujian perangkat keras bertujuan untuk mengetahui kompatibilitas aplikasi sistem presensi pada beberapa merek dan spesifikasi *smartphone*. Setelah melakukan pengujian, maka didapatkan hasil dan analisa pada Tabel 3.

Tabel 3. Hasil pengujian pada perangkat keras.

Merk	Spesifikasi	Keterangan
Samsung Galaxy E5	- Resolusi Layar : 720 x 1280 pixels - Dimensi Layar : 5 inci - Sistem Operasi : Android versi 4.4.4 (KitKat) - Memori : 16 GB, 1.5GB RAM - Processor : ARM Cortex-A53 1200 MHz	Aplikasi dapat berjalan dengan baik dan lancar
Samsung Galaxy Tab 2	- Resolusi Layar : 1280 x 800 pixels - Dimensi Layar : 10.1 inci - Sistem Operasi : Android versi 4.1.2 (Jellybean) - Memori : 8 GB, 1 GB RAM - Processor : Dual Core 1,2 GHz Qualcomm Snapdragon	Aplikasi dapat berjalan dengan baik dan lancar, namun menu terlihat kurang simetris
Oppo F1	- Resolusi Layar : 720 x 1280 pixels - Dimensi Layar : 5 inci - Sistem Operasi : Android versi 5.1 (Lollipop) - Memori : 16 GB, 3GB RAM - Processor : ARM Cortex-A53 1700 MHz	Aplikasi dapat berjalan dengan baik dan lancar
Sony Xperia C4	- Resolusi Layar : 1080 x 1920 pixels - Dimensi Layar : 5.5 inci - Sistem Operasi : Android versi 5.0 (Lollipop) - Memori : 16 GB, 2GB RAM - Processor : ARM Cortex-A53 1700 MHz	Aplikasi dapat berjalan dengan baik dan lancar
Nexian Journey One	- Resolusi Layar : 480 x 854 pixels - Dimensi Layar : 4.5 inci - Sistem Operasi : Android versi 6.0 (Marshmallow) - Memori : 8 GB, 1 GB RAM - Processor : Quad-core MTK6582 1.3 GHz	Aplikasi dapat berjalan dengan baik dan lancar.
Asus ZE500KL	- Resolusi Layar : 720 x 1280 pixels - Dimensi Layar : 5 inci - Sistem Operasi : Android versi 6.0 (Marshmallow) - Memori : 16 GB, 2 GB RAM - Processor : Quad-core 1.2 GHz Cortex-A53	Aplikasi dapat berjalan dengan baik dan lancar

Berdasarkan Tabel 3 terlihat bahwa pengujian kompatibilitas perangkat keras ini dapat di implementasikan pada berbagai tipe perangkat keras. Pada pengujian perangkat keras Samsung Galaxy Tab 2 disebutkan bahwa tampilan menu terlihat kurang simetris. Hal ini disebabkan karena perangkat memiliki dimensi layar lebih besar dibanding *smartphone* pada umumnya yaitu 10.1 inci, sehingga menyebabkan tampilan menu terlihat lebih kecil.

4. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan dapat ditarik kesimpulan bahwa selain dapat mengisi presensi dosen dan mahasiswa, aplikasi juga dapat menampilkan daftar presensi dosen, daftar peserta kuliah beserta jumlahnya, dan presentase kehadiran mahasiswa. Aplikasi telah berhasil diterapkan pada beberapa perangkat keras dengan merek dan spesifikasi yang berbeda. Dengan menerapkan teknologi NFC, sistem mampu melakukan pengisian kehadiran mahasiswa dengan rata-rata durasi waktu sebesar 2208,6 ms. Pada penelitian selanjutnya, penulis berharap pengembang dapat membuat antar muka untuk pengguna mahasiswa berdasarkan fitur-fitur yang telah tersedia pada aplikasi ini.

Referensi

- [1]. A. Cavoukian, "Mobile Near Field Communications (NFC) " Tap ' n Go " Keep it Secure & Private," *PbD*, 2011.
- [2]. V. Subtil, *Near Field Communication with Android Cookbook*. Brimingham: Packt Publishing, 2014.
- [3]. A. H. Setyawan, K. I. Satoto, and R. Isnanto, "Perancangan Aplikasi Sistem Presensi Mahasiswa Menggunakan QR Code pada Sistem Operasi Android," *Tek. Elektro Univ. Diponegoro*, 2012.
- [4]. M. . Vazquez-Briseno, F. I. . Hirata, J. . de Dios Sanchez-Lopez, E. . Jimenez-Garcia, C. . Navarro-Cota, and J. I. Nieto-Hipolito, "Using RFID/NFC and QR-Code in Mobile Phones to Link the Physical and the Digital World," *Interact. Multimed.*, pp. 219–242, 2012.
- [5]. J. Pardede, A. Nana, and P. Endo, "Implementasi NFC pada Aplikasi Layanan Informasi Benda Museum Berbasis Android," *J. Inform.*, vol. 5, no. 2, pp. 30–40, 2014.
- [6]. Y. Bassil, "A Simulation Model for the Waterfall Software Development Life Cycle," vol. 2, no. 5, 2012.
- [7]. D. M. Kroenke, *Dasar-Dasar, Desain dan Implementasi Database Processing*. Penerbit Erlangga, 2015.