

Bel Pemanggil Perawat Rumah Sakit Berbasis Wireless Menggunakan Android

Roby¹, Pono Budi Mardjoko², Meirista Wulandari³

^{1,2,3}Jurusan Teknik Elektro, Universitas Tarumanagara

Jl. Letjen. S. Parman no. 1, Grogol, Jakarta Barat

¹roby1409@outlook.com

³meiristaw@ft.untar.ac.id

Abstract— Sometimes patient needs any helps from nurses who are staying in nurse's room. However, nurse call system is too conventional. By technology advances, Internet connection can be build by combining hardware and software. This connection is used to increase efficiency of nurse call system. This research discuss about a new nurse call system by Android application which is supported by Internet connection. This system can tell the nurse the information of patient's bedroom location and patient's bed position. This system can also tell the nurse about the type of assistance that patient's need (such as emergency or not). By using an Android smartphone which has jelly bean operating system, this system can be built. By using wireless technology, cable's restriction can be minimized.

Intisari— Dalam suatu rumah sakit, pasien yang menginap sering memerlukan bantuan dari para tenaga kesehatan yaitu perawat atau dokter. Namun pada umumnya sistem pemanggil perawat yang digunakan masih sangat konvensional. Dengan kemajuan teknologi, koneksi Internet dapat dibuat dengan dukungan perangkat keras dan perangkat lunak. Koneksi Internet ini digunakan untuk meningkatkan efisiensi pelayanan perawat terhadap pasiennya. Penelitian ini membahas tentang pembuatan sistem dan aplikasi Android untuk pemanggilan perawat dengan memanfaatkan Internet. Sistem ini dapat memberitahukan posisi kamar pasien dan juga letak tempat tidur pasien. Sistem ini juga dapat memberitahu jenis bantuan yang dibutuhkan oleh pasien darurat atau tidak. Dengan memanfaatkan smartphone Android bersistem operasi Jelly bean. Sistem ini dapat bekerja secara wireless dengan bantuan koneksi Internet sehingga dapat mengurangi resiko yang terjadi yang diakibatkan oleh kendala kabel.

Kata Kunci— pemanggil rumah sakit, Android, jelly bean, Internet

I. PENDAHULUAN

Rumah sakit adalah gedung tempat menyediakan dan memberikan pelayanan kesehatan yang meliputi berbagai masalah kesehatan. Rumah sakit juga memberikan fasilitas berupa kamar untuk menginap bagi pasien yang memerlukan perawatan lebih lanjut. Setiap kamar dari rumah sakit dilengkapi dengan fasilitas-fasilitas yang baik untuk memantau perkembangan kesehatan pasien.

Dalam suatu rumah sakit, pasien yang menginap memerlukan bantuan dari para tenaga kesehatan baik perawat maupun dokter ahli. Pada saat pasien membutuhkan bantuan, seringkali orang yang sedang menjaga pasien tersebut meminta bantuan perawat secara

manual dengan memanggil perawat yang sedang berada di ruang jaga perawat atau melalui telepon yang disediakan pada tiap-tiap kamar. Pemanggilan secara manual atau melalui telepon mempunyai kelemahan yaitu seringkali mendapat tanggapan yang lambat dari pihak rumah sakit terhadap pasien. Tanggapan yang lambat dapat berakibat buruk bagi pasien itu sendiri. Pada beberapa rumah sakit, pemanfaatan kabel masih banyak digunakan untuk sistem pemanggil perawat. Penggunaan kabel juga dapat memberikan masalah, seperti data yang tidak dapat dikirimkan dengan baik akibat putusnya kabel dan kabel yang tidak terpasang dengan baik [1].

Permasalahan ini dapat diatasi dengan memanfaatkan kemajuan di bidang teknologi. Suatu software dan rangkaian elektronik dapat dibuat agar pelayanan perawat terhadap pasiennya menjadi lebih efisien [2]. Nurse call ini merupakan suatu sistem pemanggilan perawat yang setiap tombol yang terdapat pada sisi tempat tidur pasien akan terhubung pada Android yang terletak pada ruang jaga perawat sehingga dapat memberitahu perawat apa yang dibutuhkan oleh pasien. Pengiriman data tersebut akan melalui Internet.

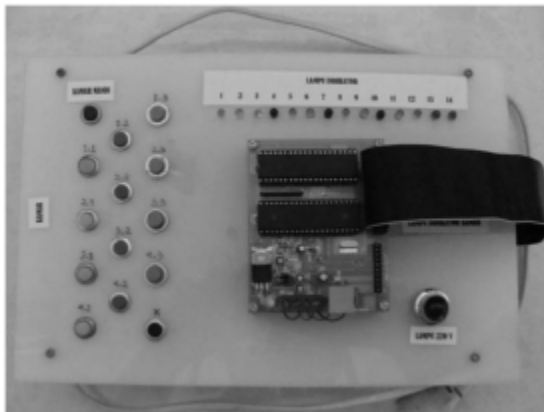
Sekarang ini, penggunaan Internet sudah menjadi hal yang biasa bagi banyak orang. Penggunaan internet semakin terkenal ketika telepon pintar (smartphone) muncul. Smartphone dikendalikan oleh sebuah sistem operasi. Sistem operasi mengatur banyak hal, seperti pengiriman data, komunikasi, dan cara kerja dari alat tersebut. Pada smartphone, terdapat beberapa jenis sistem operasi seperti sistem operasi Android dan sistem operasi IOS. Smartphone yang banyak dipakai adalah smartphone Android. Pada smartphone Android, sudah mulai muncul pengendalian melalui smartphone atau yang sering disebut dengan Internet of Things (IoT).

Smartphone Android dapat dimanfaatkan untuk membuat suatu sistem yang dapat membantu pelayanan rumah sakit, dalam hal ini adalah pelayanan perawat terhadap pasiennya. Suatu sistem pemanggilan perawat ini dapat dilakukan dengan menekan tombol-tombol pada aplikasi smartphone yang telah dirancang khusus. Smartphone pada sisi tempat tidur pasien akan terhubung pada suatu perangkat Android yang terletak pada ruang jaga perawat.

Survei dilakukan sebanyak dua kali. Survei pertama dilakukan melalui studi pustaka berupa jurnal ilmiah Politeknik Negeri Semarang yaitu Jurnal Teknik Elektro Terapan (JTET) yang berjudul Bel Pemanggil Perawat

Berbasis *Wireless* Menggunakan Xbee yang disusun oleh Ilham Sayekti [1]. Sistem pemanggil perawat tersebut menggunakan modul Xbee untuk melakukan pengiriman data dan juga sebagai penerima. Sistem ini menggunakan dua tombol pada bagian pengirim yaitu tombol *Call* dan *Stop*. Setiap kali tombol *Call* ditekan, modul pengirim akan mengirimkan data. Data yang dikirimkan akan diterima oleh modul penerima. Data tersebut akan mengaktifkan LED dan pemanggil. Alat akan terus aktif bila tombol *Stop* belum ditekan. Komunikasi dilakukan antara modul pengirim dan penerima dengan menggunakan frekuensi radio. Sistem ini sangat dipengaruhi oleh jarak jangkauan frekuensi radio. Sistem dirancang hanya untuk melakukan komunikasi 1 arah yaitu dari pihak pengirim ke penerima saja.

Survei kedua dilakukan melalui studi pustaka berupa jurnal ilmiah Universitas Budi Luhur yaitu Jurnal *TELEMATIKA MKOM* yang berjudul *Sistem Pengendali Nurse Station Menggunakan PPI 8255* yang disusun oleh Nurul Jamal [2]. Sistem ini menggunakan Serial PPI 8255 untuk mengirimkan data. Pengiriman data ini merupakan komunikasi antara bagian modul *interface* alat dan komputer. Komunikasi ini dilakukan secara serial dengan menggunakan kabel serial DB9 ke *port* USB komputer. Data yang masuk ke komputer akan ditampilkan pada layar sesuai dengan jenis tombol yang ditekan pada modul *interface*. Tampilan dari modul *interface* dengan menggunakan PPI 8255 dapat dilihat pada Gambar 1.



Gambar. 1. Modul *Interface* dengan Menggunakan PPI 8255

Tombol-tombol tersebut digunakan untuk memberitahu kebutuhan pasien. Setiap tombol memiliki fungsi yang berbeda. Tombol-tombol tersebut memiliki fungsi untuk memberitahu ketika obat habis, pasien membutuhkan tindakan medis, dan pasien memiliki keluhan fasilitas. Tombol-tombol tersebut tersedia pada setiap kamar pasien dan memberi informasi kamar tempat pasien berada. Modul ini juga mempunyai indikator lampu yang berfungsi sebagai indikator letak kamar pasien.

Pada penelitian ini sistem pemanggil perawat dengan menggunakan *smartphone Android*. Pemanggilan perawat dilakukan dengan menggunakan modul *wi-fi* yang terhubung ke Internet. Setiap kamar pasien memiliki tombol-tombol yang membedakan kebutuhan pasien darurat atau tidak. Internet digunakan untuk melakukan pengiriman data yang akan diterima oleh *smartphone* yang terhubung ke Internet juga. Komunikasi dapat terus

dilakukan selama terdapat koneksi internet pada kedua modul.

Selain pengiriman data menggunakan Internet, setiap kali pasien menekan tombol virtual untuk memanggil perawat di ruang jaga.

Pada bagian *smartphone*, akan muncul notifikasi setiap kali terjadi panggilan dan perawat dapat menekan *button* pada tampilan *smartphone* menyalakan lampu pada atas tempat tidur pasien sebagai indikator. Indikator ini berguna untuk memberitahu pasien bila perawat sudah dalam perjalanan menuju kamar untuk memberikan bantuan.

Tujuan dari sistem ini adalah membuat sistem dan aplikasi *Android* dalam pemanggilan perawat dengan memanfaatkan Internet. Sistem ini dapat memberitahukan posisi kamar pasien dan juga letak tempat tidur pasien. Sistem ini juga dapat memberitahu jenis bantuan yang dibutuhkan oleh pasien darurat atau tidak.

II. LANDASAN TEORI

A. *Internet of Things*

Konsep IoT sebenarnya sudah ditemukan di *Massachusetts Institute of Technology* (MIT) sejak tahun 1999 [3]. Secara umum konsep IoT diartikan sebagai sebuah kemampuan untuk menghubungkan objek-objek cerdas dan memungkinkannya untuk berinteraksi dengan objek lain, lingkungan maupun dengan peralatan komputasi cerdas lainnya melalui jaringan Internet [4]. Pemakaian IoT sudah banyak digunakan di Amerika, Asia dan eropa dalam berbagai bidang dan masih terus dikembangkan sampai sekarang [5]. Dengan terus berkembangnya IoT, di masa depan penggunaan IoT diperkirakan akan dikombinasikan dengan teknologi dan konsep yang terkait seperti *Cloud computing*, *Future Internet*, *Big Data*, robotika dan semantic. Penggunaan IoT pada sistem pemanggilan perawat menggunakan *Android* ini dapat memudahkan pengiriman data. Pengiriman suatu data dapat dikirimkan menuju beberapa *device* dalam waktu yang bersamaan.

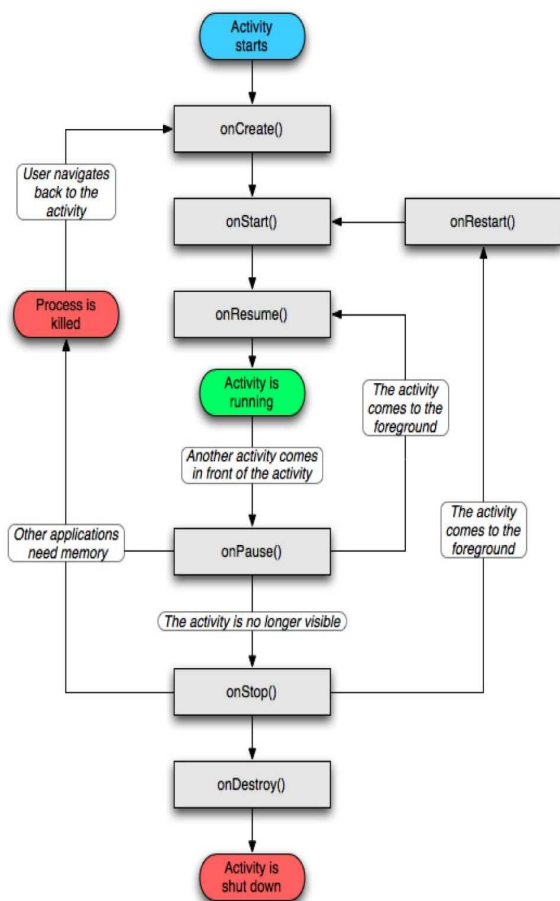
B. *Android*

Ada beberapa pengertian mengenai *Android*. *Android* adalah sebuah sistem operasi untuk perangkat *mobile* berbasis Linux yang mencakup sistem operasi, *middleware* dan aplikasi [6]. *Android* adalah sistem operasi untuk telepon seluler yang berbasis Linus. *Android* menyediakan *platform* terbuka bagi para pengembang untuk membuat suatu aplikasi. Pada awalnya dikembangkan oleh *Android Inc*, sebuah perusahaan baru yang membuat perangkat lunak untuk ponsel yang kemudian dibeli oleh *Google Inc*. Untuk pengembangannya, dibentuklah *Open Handset Alliance* (OHA), konsorsium dari 34 perusahaan perangkat keras, perangkat lunak dan telekomunikasi termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile dan Nvidia [7]. Selain pada *smartphone*, sistem operasi *Android* juga sudah dipakai pada komputer tablet, televisi pintar (*smart television*), dan masih banyak alat-alat yang kini sudah mulai menggunakan sistem operasi ini. Google menyediakan beberapa program yang dapat digunakan oleh para *developer* untuk membuat

aplikasi *Android* mereka sendiri secara gratis. Aplikasi tersebut dapat dibuat dengan beberapa *software*, seperti *Android Studio* dan *B4A* [8].

Pada alur kerja tersebut terdapat beberapa kondisi yaitu *onCreate*, *onStart*, *onResume*, *onPause*, *onStop*, *onDestroy*, dan *onRestart*. Kondisi *onCreate* bertugas untuk melakukan perintah-perintah pada saat awal menjalankan suatu aplikasi tersebut. Kondisi *onStart* bertugas untuk melakukan perintah-perintah pada saat awal suatu aplikasi berjalan. Kondisi *onResume* bertugas untuk melakukan perintah-perintah ketika suatu aplikasi berjalan. Kondisi *onPause* bertugas untuk melakukan perintah-perintah pada saat aplikasi berada pada *background*. Kondisi *onStop* bertugas untuk menjalankan perintah-perintah pada saat aplikasi berhenti bekerja. Kondisi *onDestroy* bertugas untuk menghapus aplikasi pada *task manager* sehingga memori dapat digunakan kembali oleh aplikasi lain. Kondisi *onRestart* bertugas untuk menjalankan perintah-perintah pada saat aplikasi kembali dijalankan dari *background*.

Sistem operasi *Android* memiliki suatu alur kerja untuk setiap aplikasinya. Alur kerja tersebut dapat dilihat pada Gambar 2.



Gambar 2. Alur Kerja Aplikasi Sistem Operasi *Android*

C. Versi *Android*

Android telah mengalami sejumlah pembaruan sejak pertama kali dirilis. Rata-rata versi terbaru dari *Android* dirilis setiap 6 bulan. Berikut beberapa jenis *Android* dan penjelasannya:

1) *Cupcake*

Android 1.5 (API level 3) yang dikenal dengan nama kode *Cupcake* in dirilis pada April 2009. API level 3 membawa perbaikan dari versi sebelumnya dan tentu saja fitur baru. Misalnya dukungan *On-screen keyboard*, *video recording*, *home screen widget*, fitur auto *pairing* pada *Bluetooth*, kernel Linux versi 2.6.7 dan kemampuan memperbaiki file *system SD Card* yang rusak.

2) *Donut*

Android 1.6 (API Level 4) yang dirilis pada Oktober 2009 ini merupakan rilis minor dengan fitur utama berupa dukungan teknologi *CDMA*, *gesture*, *text-to-speech engine* dan fitur pencarian cepat yang memungkinkan pencarian semua hal seperti kontak, riwayat jelajah Internet, *bookmark* dan lainnya.

3) *Éclair*

Android 2.1 (API Level 7) yang dirilis pada Januari 2010 ini mengusung fitur baru seperti *live wallpaper* dan beberapa fitur baru seperti *Android 2.0*, misalnya menggunakan banyak akun kontak cepat, *bluetooth 2.1* dan profil *Bluetooth* baru yaitu *Object Push Profile* (OPP) dan *Phone Book Access Profile* (PBAP).

4) *Froyo*

Android 2.2 (API Level 8) dirilis pada Mei 2010. Fitur baru yang dibawa adalah dukungan *OpenGL ES 2.0*, instalasi aplikasi penyimpanan eksternal *SD card* dan *Android Cloud to Device Mesagging* yang memungkinkan aplikasi melakukan *push mesagging* dan *portable hotspot* yang memungkinkan perangkat *Android* menjadi *hotspot wifi* untuk berbagi koneksi Internet.

5) *Gingerbread*

Android 2.3 (API Level 9 dan 10) mengusung beberapa fitur baru antara lain dukungan banyak kamera, *Near Field Communication* (NFC), *download manager service*, dukungan terhadap sensor lain seperti giroskop dan barometer.

6) *Honeycomb*

Android 3.0 (API Level 11) membawa perubahan besar terutama pada tampilan UI yang berubah drastis supaya optimal untuk perangkat layar besar seperti tablet. Pada versi ini diperkenalkan *fragment*, *action bar*, *system clipboard* dan *cursor leader*.

7) *Ice Cream Sandwich*

Android 4.0 (API Level 14) membawa perubahan besar terutama pada tampilan UI *Android 3.0* supaya cocok pada layar kecil sehingga memungkinkan aplikasi tampak konsisten di tablet maupun ponsel. Fitur baru seperti *Android Beam* dan *wifi direct* juga ditambahkan.

8) *Jellybean*

Android 4.1 (API Level 16) adalah rilis minor yang membawa perbaikan fungsionalitas dan performa rendering UI. Versi 4.2 (API Level 17) mengusung fitur baru seperti *gesture typing* dan dukungan multi user di perangkat tablet. Versi 4.3 (API Level 18) membawa perbaikan berupa dukungan *bluetooth low energy* dan *OpenGL ES 3.0*. Beberapa fitur baru antara lain dukungan terhadap bahasa internasional dan penulisan teks dua arah (kiri ke kanan atau kanan ke kiri untuk bahasa-bahasa tertentu seperti Bahasa Arab).

9) Kitkat

Android 4.4 (API Level 19) ini mengusung sejumlah perbaikan dan fitur baru, terutama dukungan teknologi NFC melalui *host card emulation*, pencetakan ke *printer* nirkabel, WebView dengan rendering *engine chromium* dan dukungan yang lebih baik bagi perangkat yang menggunakan RAM rendah.

D. Arsitektur Android

Secara umum, arsitektur *Android* terdiri dari lapisan perangkat lunak yaitu lapisan aplikasi, lapisan *framework*, *Android run-time* sebagai lapisan perantara (*middleware*) serta kernel Linux yang membungkus perangkat keras di bawahnya. Berikut penjelasan arsitektur dari *Android*:

1) Aplikasi

Aplikasi yang dibuat berada pada level tertinggi dalam hierarki struktur perangkat lunak di *Android*. Termasuk di dalamnya aplikasi utama bawaan *Android* seperti telepon, pembaca surel, program sms, kontak, *browser* dan kalender. Aplikasi umumnya ditulis dalam bahasa pemrograman Java.

2) Framework Aplikasi

Di bawah aplikasi terdapat sejumlah perangkat lunak pendukung, meliputi layanan pengelola *activity*, *view*, sumber daya, pemberitahuan (notifikasi) dan lain-lain.

3) Pustaka

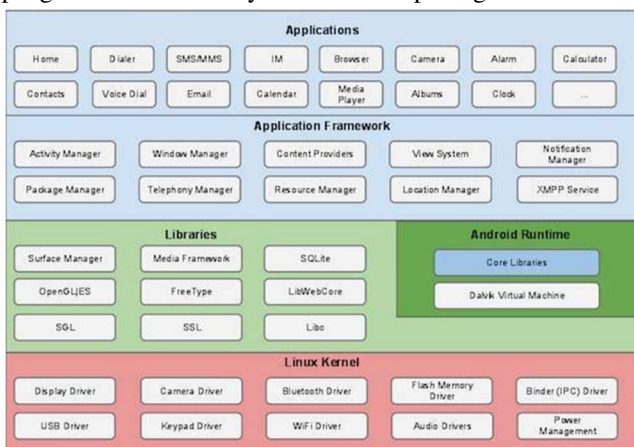
Android menyediakan sejumlah pustaka dasar seperti pustaka grafis 2D dan 3D, multimedia *playback*, *browser engine*, pencetakan huruf, *database* dan lain-lain. Aplikasi yang memanfaatkan pustaka fungsi ini melalui lapisan *framework* aplikasi

4) Android Runtime

Android menyediakan pustaka inti bagi pemrograman Java. Dibawahnya terdapat Dalvik *Virtual Machine* (Dalvik VM) yang akan menjalankan aplikasi. Tiap aplikasi akan dijalankan pada proses terpisah dengan VM yang berbeda sehingga terisolasi satu sama lain. Aplikasi masih dapat berkomunikasi dengan aplikasi lain melalui mekanisme yang disediakan *framework* aplikasi. Dalvik VM bergantung pada lapisan di bawahnya (kernel Linux) untuk *multi-threading* dan pengelolaan memori tingkat rendah.

5) Kernel Linux

Kernel Linux bertanggung jawab menyediakan layanan dasar seperti keamanan, pengelolaan proses, pengelolaan file, pengelolaan sumber daya memori dan perangkat keras.



Gambar 3. Arsitektur *Android* [8]

E. Komponen Aplikasi Android

Aplikasi *Android* ditulis dalam bahasa pemrograman Java. Semua file kode *intermediate* dan aset disatukan dalam satu paket berupa file berekstensi *.APK*, sebuah file yang dapat didistribusi. Tiap file *.APK* adalah sebuah aplikasi tunggal. Komponen aplikasi *Android* terdiri dari beberapa jenis antara lain:

1) Activity

Activity adalah istilah yang digunakan dalam pemrograman *Android* untuk mengacu pada satuan interaksi dengan pengguna melalui antarmuka grafis (*graphical user-interface*, GUI). Sebagai satuan interaksi, *activity* adalah tampilan yang dilihat di layar seperti *Windows* atau kotak dialog pada pemrograman aplikasi *desktop*. Tiap aplikasi dapat terdiri dari nol atau lebih *activity*. Selain sebagai satuan interaksi dengan pengguna, *activity* juga satuan eksekusi. Sebagai satuan eksekusi, *activity* selalu memiliki paling tidak satu buah *Thread*, yakni *Thread* utama yang digunakan untuk memperbarui tampilan *user-interface* UI *Thread*.

2) Intent

Intent adalah istilah yang digunakan dalam pemrograman *Android* untuk mengacu pada mekanisme berbagi pesan pemberitahuan atau bertukar data *activity* atau untuk menjalankan aplikasi lain.

3) Service

Service adalah komponen aplikasi yang berjalan di belakang layar tanpa UI untuk menyediakan layanan tertentu seperti mengecek RSS *feed* secara kontinu atau memainkan musik. *Service* tetap berjalan meski *activity* yang mengendalikannya telah berhenti. Media *player* adalah contoh aplikasi yang menggunakan *service*.

4) Content Provider

Content Provider membuat suatu aplikasi dapat berbagi sejumlah data tertentu kepada aplikasi lain. Jika membutuhkan data nama-nama kontak, aplikasi meminta data tersebut.

5) Broadcast Receiver

Broadcast Receiver adalah komponen yang memantau, menerima dan bereaksi terhadap pesan yang disebarkan, baik oleh sistem maupun oleh aplikasi lain. Misalnya, ketika baterai lemah, *Android* akan mengirim pesan “Baterai Lemah” kepada semua *broadcast receiver* yang ingin diberitahu pesan ini. Untuk menggunakan *broadcast receiver*, pada dasarnya hanya perlu membuat turunan tipe *broadcast receiver*, melengkapi metode *onReceive()*, dan mendaftarkannya di *AndroidManifest.xml* atau dengan metode *Context.registerReceiver()* sehingga seseorang tidak boleh menyimpan referensi ke *instance* ini. Satu hal yang dapat dilakukan ketika *onReceive()* dipanggil adalah mewakilkannya ke komponen lain, misalnya dengan memanggil metode *startActivity()* atau *startService()* milik *Context*.

F. Width Fidelation (WIFI)

Wireless Fidelity (Wi-Fi) merupakan teknologi *Wireless Local Area Network* (WLAN) yang distandarisasi dalam standar IEEE 802.11. Wi-Fi digunakan untuk transmisi data secara *wireless* [9]. Wi-Fi menggunakan jaringan radio dengan jangkauan frekuensi 2,4 GHz dan 5 GHz. Jaringan Wi-Fi bermula dari *access point*, yaitu perangkat yang menjalankan komunikasi dengan frekuensi

radio dan memungkinkan perangkat *wireless* untuk terkoneksi jika memiliki konfigurasi dan kualifikasi keamanan yang sesuai [10]. Wi-Fi memiliki standar-standar dengan spesifikasi tertentu yang dapat dilihat pada Tabel I.

TABEL I
SPESIFIKASI STANDAR WI-FI [11]

Standard	Frequency range	Theoretical maximum throughput	Effective throughput (approximate)	Average geographic range
802.11b	2.4 GHz	11 Mbps	5 Mbps	100 meter
802.11a	5 GHz	54 Mbps	11-18 Mbps	20 meter
802.11g	2.4 GHz	54 Mbps	20-25 Mbps	100 meter
802.11n	2.4 GHz or 5 GHz	65 to 600 Mbps	65 to 600 Mbps	Up to 400 meter

G. Android Studio

Android Studio merupakan IDE resmi untuk mengembangkan aplikasi pada *platform Android* yang diumumkan dalam acara *Google I/O* pada tahun 2013. *Android Studio* tersedia secara gratis di bawah lisensi *Apache 2.0*, versi *stable* pertama dari *Android Studio* dirilis pada Desember 2014 dan dimulai dari versi 1.0. *Android Studio* telah menggantikan *Eclipse Android Development Tools (Eclipse ADT)* sebagai IDE utama untuk pengembangan aplikasi *Android* [12].

Aplikasi *Android* dikembangkan menggunakan bahasa pemrograman *Java*. Desain *user interface (UI)* dari aplikasi *Android* dilakukan dengan menggunakan bahasa (*Extensible Markup Language (XML)*). Sebuah aplikasi *Android* pada umumnya terdiri dari beberapa *Javafile (.java)* dan *layout file (.xml)*, serta berbagai *file* lain yang dibutuhkan untuk menjadikannya sebuah proyek aplikasi *Android* yang utuh. Aplikasi *Android* dapat didistribusikan dan dipasang pada perangkat *Android* setelah di-*package* menjadi hanya sebuah *Android Application Package (APK) file (.apk)*. *Android build system* melakukan kompilasi *resource* dan *source code* dari sebuah proyek aplikasi *Android*, kemudian mem-*package*-nya menjadi *APK file*. *Android Studio* menggunakan *Gradle*, yang merupakan *advanced build toolkit* untuk mengotomatisasi dan mengelola proses *build*, sembari memungkinkan para pengembang untuk mengkustomisasi konfigurasi *build* secara fleksibel.

III. METODE PENELITIAN

Penelitian ini dilakukan dengan diawali studi literatur, survei dilanjutkan dengan perancangan *software* dan *hardware*. Survei yang dilakukan mencakup survei karya ilmiah dan survei ke rumah sakit. Adapun hasil survei yang dibandingkan dengan penelitian yang dilakukan dapat dilihat pada Tabel II.

Hasil survei dilihat tentang jarak komunikasi antara kamar pasien dengan ruang jaga perawat terbatas. Hasil survei pertama menyatakan jaraknya kurang dari 20 meter, hasil survei menyatakan jaraknya terbatas pada panjang kabel yang digunakan dan pada penelitian ini jaraknya terbatas sepanjang mendapatkan koneksi Internet. Hasil survei dilihat mengenai kemampuan untuk membedakan kebutuhan pasien. Hasil survei pertama menyatakan bahwa perbedaan jenis kebutuhan pasien tidak ada, hasil survei kedua menyatakan bahwa ada perbedaan jenis kebutuhan pasien dan penelitian ini dilakukan dengan membedakan jenis kebutuhan pasien antara kebutuhan yang darurat atau kebutuhan yang tidak darurat. Hasil survei dilihat pada kategori kemampuan membedakan kasur pasien yang

memanggil pada kamar yang sama. Hal ini diperlukan menginggit kelas rumah sakit yang bervariasi ada yang 1 kamar 1 pasien namun ada juga kelas dengan 1 kamar pasien tidak hanya 1 pasien melainkan bisa 2 sampai 6 pasien. Pada survei pertama dan survei kedua tidak dijelaskan mengenai lokasi kasur pasien namun pada penelitian ini lokasi kasur pasien dapat diketahui.

TABEL II.
PERBANDINGAN HASIL SURVEI DENGAN PENELITIAN

No	Parameter	Hasil Survei		Penelitian
		Hasil Survei Pertama [1]	Hasil Survei Kedua [2]	
1.	Jarak Komunikasi	<20 meter	Terbatas pada Panjang Kabel yang Digunakan	Selama Mendapatkan Koneksi Internet
2.	Kemampuan Membedakan Kebutuhan Pasien	Tidak Ada	Ada	Ada
3.	Kemampuan untuk Membedakan Posisi Kamar Pasien	Ada	Ada	Ada
4.	Kemampuan Membedakan Pasien yang Memanggil pada Kamar yang Sama	Tidak Ada	Tidak Ada	Ada

A. Jenis Penelitian

Jenis penelitian ini adalah perancangan aplikasi menggunakan *Android*. Aplikasi yang dimaksud adalah aplikasi pemanggilan perawat berbasis wi-fi menggunakan *Android*.

B. Lokasi dan Waktu Penelitian

Penelitian ini dilakukan di *workshop* Teknik Elektro Universitas Tarumanagara dan Laboratorium Komputer Universitas Tarumanagara. Penelitian ini dimulai dari Bulan Oktober 2016 sampai dengan Januari 2017.

C. Alat dan Bahan Penelitian

Alat dan bahan penelitian ini meliputi perangkat keras dan perangkat lunak. Perangkat keras yang digunakan dalam penelitian ini adalah *smartphone* dengan sistem operasi *Android Jelly bean*. Perangkat lunak yang digunakan dalam penelitian ini adalah pemrograman dengan *Android studio*.

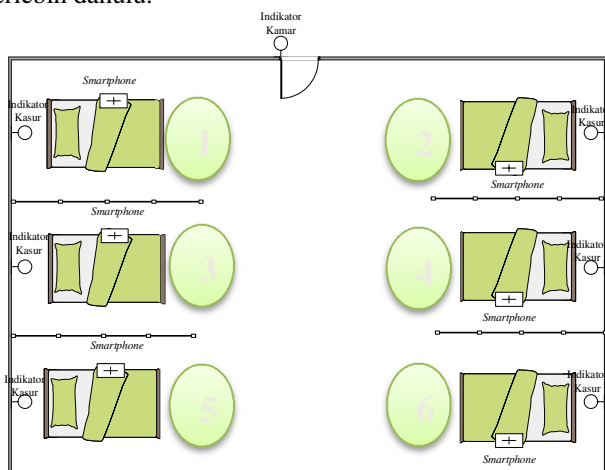
D. Diagram Blok

Penelitian ini terbagi menjadi 2 bagian yaitu pada bagian kamar dan ruang jaga perawat. Setiap kamar diberikan fasilitas berupa tombol virtual untuk memanggil perawat pada setiap tempat tidur. Penempatan *smartphone* dan kasur pasien telah ditentukan sebelumnya. Hal ini dapat ditunjukkan pada Gambar 4. *Smartphone* diletakkan di samping tempat tidur. Setiap kasur mempunyai nomornya masing-masing. Nomor tersebut ditentukan dari kiri ke kanan, dilanjutkan ke kiri bawah dan seterusnya. Penomoran ini diperlukan agar terjadi kesesuaian dengan tampilan notifikasi pada *smartphone* yang diletakkan di ruang jaga perawat sehingga perawat dapat mengetahui pasien dari kasur nomor tertentu yang melakukan panggilan. Setiap tempat tidur diberikan 2 buah tombol virtual pada layar *smartphone*. Kedua tombol virtual tersebut adalah tombol virtual untuk memanggil perawat dengan jenis bantuan biasa dan tombol virtual untuk memanggil perawat dengan jenis

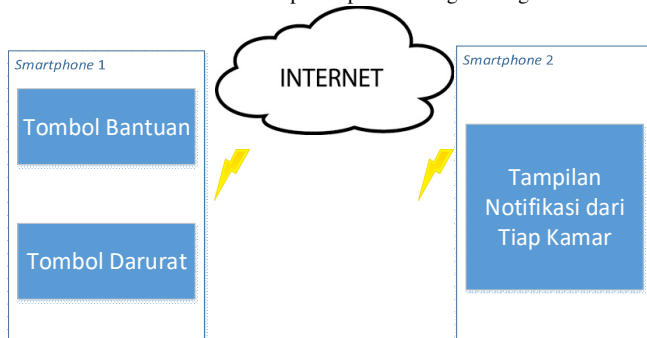
bantuan darurat. Tombol virtual pemanggil perawat dengan jenis bantuan biasa digunakan untuk memanggil perawat yang sedang berjaga bila pasien sedang membutuhkan bantuan sederhana seperti membutuhkan makanan atau ingin ke toilet. Tombol virtual pemanggil perawat dengan jenis bantuan darurat digunakan untuk memberikan pemberitahuan kepada ruang jaga perawat bila pasien pada kamar tersebut sedang membutuhkan suatu hal yang penting atau sedang berada pada kondisi yang kritis. Setiap kali tombol pemanggil perawat dan darurat ditekan, panggil perawat dan tombol darurat berada di dekat kasur pasien sehingga dapat memudahkan pasien saat melakukan pemanggilan.

Setiap ada tombol virtual yang ditekan, data kamar akan berubah sesuai dengan jenis tombol virtual yang ditekan. Setiap terjadi perubahan pada data kamar, data kamar akan dikirimkan melalui modul *wifi*. Data kamar yang dikirim dapat dilihat pada layar *smartphone Android* di ruang jaga perawat.

Tampilan *user interface* dirancang pada *smartphone Android* agar perawat dapat melihat dimana posisi pasien yang membutuhkan bantuan. Dengan demikian, perawat dapat melakukan pelayanan yang prioritas lebih tinggi terlebih dahulu.



Gambar. 4. Denah Kamar Pasien beserta Lokasi Kasur Pasien dengan Letak Tombol Virtual dari Smartphone pada Masing-Masing Kasur Pasien

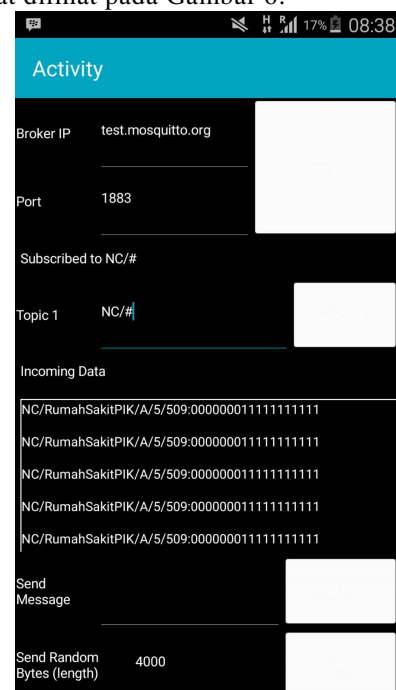


Gambar. 5. Diagram Blok Sistem Nurse Call Berbasis Android

Diagram blok dari rancangan sistem pemanggilan perawat berbasis *Android* dapat dilihat pada Gambar 5. *Smartphone 1* adalah *smartphone* yang diletakkan di kamar pasien. *Smartphone 2* adalah *smartphone* yang diletakkan di ruang jaga perawat. Sistem ini dapat bekerja ketika terdapat *input* dari pasien. *Input* tersebut

berasal dari tombol virtual yang ditekan oleh pasien. Tombol tersebut akan mengirimkan sinyal ke pemroses. Sinyal akan diolah untuk menyesuaikan data kamar. Data kamar akan berubah sesuai dengan tombol yang ditekan. Data kamar akan dikirimkan melalui modul *wifi* ke *smartphone Android* setiap kali terjadi perubahan. Pengiriman data dilakukan dengan menggunakan bantuan *Message Queuing Telemetry Transport (MQTT)*. MQTT adalah sebuah sistem pengiriman pesan yang menggunakan metode *subscribe/publish*. MQTT dapat mengirim berbagai jenis data seperti data biner, text, dan bahkan XML. MQTT dapat digunakan sebagai sarana pertukaran data.

MQTT memiliki beberapa *command* untuk mengaksesnya, seperti *subscribe*, *unsubscribe*, *connect*, *disconnect*, dan *publish*. Perintah *subscribe* dan *unsubscribe* berguna untuk mendapatkan data dari suatu *topic* yang ingin diterima/dikirim. Perintah *connect* dan *disconnect* berfungsi untuk mendapatkan koneksi atau memutuskannya dari MQTT. Pada penggunaan MQTT, terdapat beberapa data penting yang harus dimasukkan oleh pengguna seperti broker IP, port, *topic*. Broker IP memperlihatkan alamat IP dari jaringan yang digunakan. Port menunjukkan port yang dipakai. *Topic* menunjukkan topik yang dibaca dan dikirimkan. Ketika suatu *device* melakukan proses *subscribe* pada suatu *topic*, *device* tersebut akan menerima segala data yang dikirimkan (*publish*) pada *topic* tersebut. Penggunaan tanda pagar (#) pada penulisan *topic*, akan membuat suatu *device* melakukan *subscribe* pada semua *topic* yang ada pada MQTT sehingga akan menerima seluruh data yang masuk pada MQTT. Contoh tampilan aplikasi MQTT dapat dilihat pada Gambar 6.



Gambar. 6. Tampilan MQTT

Smartphone Android pada kasur pasien akan melakukan proses *connect* pada MQTT. Bila proses berhasil, *smartphone* akan menerima segala data yang masuk sesuai dengan *topic* dan akan mengirimkan data sesuai dengan *topic*. *Topic*, IP broker, dan port akan

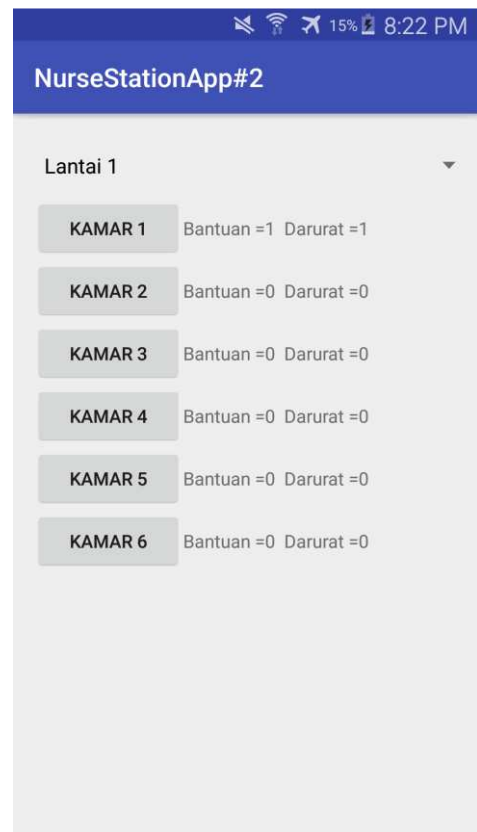
disesuaikan dengan data yang terdapat pada aplikasi pemanggil sehingga *smartphone Android* tidak akan menerima data yang tidak diinginkan atau dibutuhkan oleh sistem keseluruhan. *Smartphone Android* yang terletak pada ruang akan melakukan proses *connect* terhadap MQTT. Setelah berhasil, *smartphone Android* pada ruang jaga perawat akan menerima data yang dikirimkan pada MQTT sesuai dengan *topic* yang sudah ditentukan. Data tersebut akan diolah oleh *smartphone*. Data yang telah diterima akan diolah menjadi beberapa data yaitu posisi kamar pasien dan posisi kasur pasien yang memerlukan bantuan. *Smartphone* akan menampilkan posisi kamar pasien yang membutuhkan bantuan dan jenis bantuan yang dibutuhkan sesuai dengan tombol yang ditekan. Data akan dikirimkan ke *smartphone* di ruang jaga perawat dengan tingkatan lantai yang sama dengan kamar pasiennya (mengingat pada umumnya rumah sakit merupakan gedung bertingkat). Setelah menerima notifikasi, perawat akan menuju ke kamar pasien. Setelah perawat berada di kamar pasien, perawat akan menekan *reset button* pada *smartphone* yang akan mengubah data kamar tersebut. Hal ini berguna untuk memberitahu perawat lain bila kamar tersebut sudah dilayani dan perawat yang lain dapat menuju kamar pasien lain membutuhkan bantuan.. Setelah selesai membantu pasien, perawat akan menekan tombol *Reset*. Tombol *Reset* akan mengubah data kamar kembali seperti semula dan notifikasi pada *smartphone* akan hilang.

IV. HASIL DAN ANALISIS

Tampilan virtual dibuat pada sistem *Android*. Pada tampilan ini ditampilkan informasi mengenai lantai tempat pasien berada, nomor kamar pasien dan nomor kasur. Nomor kasur ini penting karena umumnya untuk kelas kamar yang bukan VIP, satu kamar rumah sakit bisa diisi oleh beberapa pasien (6-8 orang pasien). Nomor kasur disesuaikan dengan denah kamar pasien. Terdapat 2 buah tombol virtual yaitu panggil perawat dan darurat. Panggilan darurat hanya diperkenankan untuk keadaan-keadaan yang sangat mendesak yang mempengaruhi keselamatan pasien. Gambar tampilan dapat dilihat pada Gambar 7. Gambar 7 menunjukkan tampilan program pemanggil pada *Android* di samping kasur pasien. Tampilan memberikan informasi tentang posisi tingkatan lantai kamar pasien, nomor kamar dan nomor kasur pasien. Tingkatan lantai ini ditampilkan karena pada umumnya gedung rumah sakit bertingkat sehingga diperlukan informasi yang jelas. Tingkatan lantai yang ditunjukkan pada Gambar 7 adalah lantai 1 dengan nomor kamar 1 dan nomor kasur 1. Masing-masing *Android* di samping kasur pasien menampilkan informasi yang berbeda-beda sesuai dengan tingkatan lantai, nomor kamar dan nomor kasur pasien itu sendiri.

Pada Gambar 8 menunjukkan tampilan notifikasi di ruang jaga perawat. Tampilan ini memberi informasi tentang tingkatan lantai ruang jaga perawat di suatu gedung rumah sakit bertingkat dan memberi informasi tentang nomor kamar pasien. Pada baris pertama tampilan ini menunjukkan bahwa ruang jaga perawat berada di lantai 1 suatu gedung rumah sakit. Tampilan ini juga berisi pemberitahuan tentang jenis panggilan yang dilakukan

pasien kepada perawat. Jenis panggilan dibedakan menjadi 2 yaitu bantuan dan darurat. Pada kamar 1, terdapat 2 orang pasien yang memanggil perawat di ruang jaga perawat. Seorang pasien melakukan jenis panggilan bantuan dan seorang lainnya melakukan jenis panggilan darurat. Pada kamar 2, kamar 3, kamar 4, kamar 5 dan kamar 6 tidak ada pasien yang melakukan panggilan kepada perawat. Setiap tombol virtual kamar pada tampilan ini dapat ditekan. Setelah tombol virtual “Kamar” ditekan, tampilan pada *Android* di ruang jaga perawat akan berubah seperti yang ditunjukkan pada Gambar 9. Gambar 9 menunjukkan tampilan notifikasi jika tombol virtual “Kamar 1” pada Gambar 8 ditekan. Tampilan ini berisi tentang informasi nomor kasur pasien dalam suatu kamar. Terdapat 6 kasur dalam 1 kamar.



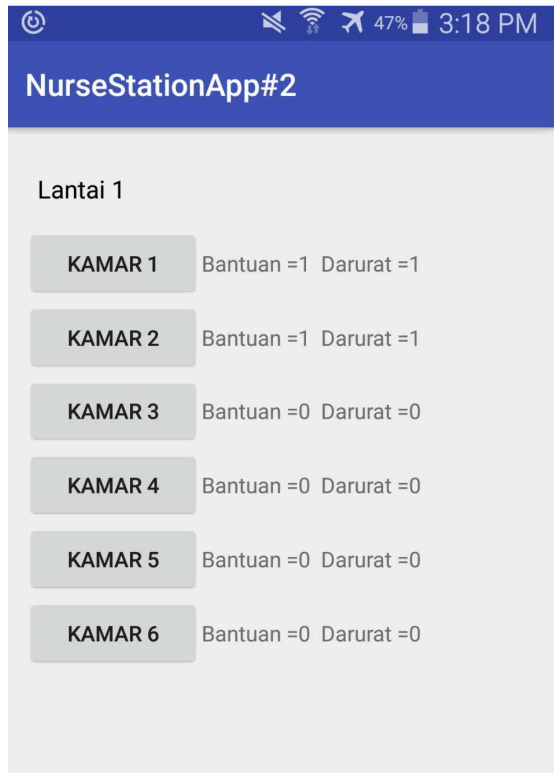
Gambar. 8. Tampilan Notifikasi di Ruang Jaga Perawat pada Lantai 1 Gedung Rumah Sakit

Pada Gambar 4 telah diinformasikan bahwa terdapat 2 orang pasien yang melakukan panggilan kepada perawat. Seorang pasien melakukan jenis panggilan bantuan dan seorang pasien lainnya melakukan jenis panggilan darurat. Gambar 5 menunjukkan bahwa pasien yang melakukan panggilan bantuan adalah pasien dengan nomor “Kasur 5” dan pasien yang melakukan panggilan darurat adalah pasien dengan nomor “Kasur 2”. Selain pada kamar 1, pengujian sistem dilakukan terhadap kamar lainnya. Gambar 10 menunjukkan informasi bahwa terdapat 4 orang pasien yang membutuhkan bantuan. Dua pasien dari kamar 1 dan 2 pasien dari kamar 2. Pada kamar 1, seorang pasien membutuhkan jenis bantuan biasa dan seorang pasien lainnya membutuhkan jenis bantuan

darurat. Hal yang sama terjadi pada kamar 2. Ada dua pasien yang membutuhkan bantuan dengan tipe bantuan dan jumlah pasien yang sama dengan kamar 1.



Gambar. 9. Tampilan Notifikasi di Ruang Jaga Perawat pada Kamar 1



Gambar. 10. Tampilan Notifikasi pada Ruang Jaga Perawat di Lantai 1



Gambar. 11. Tampilan Notifikasi berisi Informasi pada Kamar 1



Gambar. 12. Tampilan Notifikasi berisi Informasi pada Kamar 1

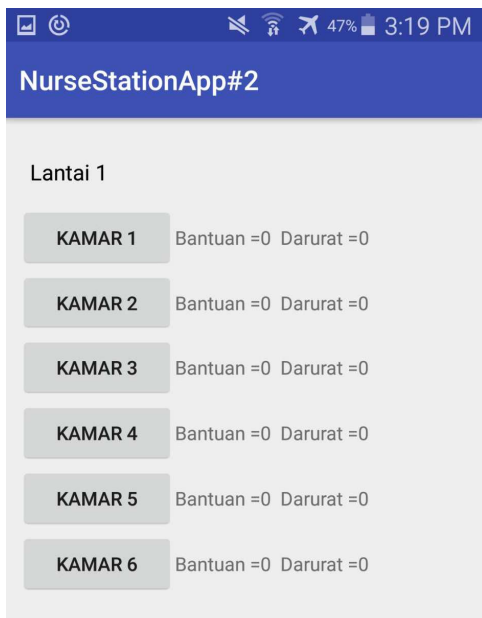
TABEL III.
HASIL PENGUJIAN PENEKANAN TOMBOL VIRTUAL DI SETIAP KASUR PASIEN PADA KAMAR 1 – KAMAR 6

Kamar 1			Kamar 4		
Kasur	Jenis Tombol Virtual		Kasur	Jenis Tombol Virtual	
	Bantuan	Darurat		Bantuan	Darurat
Kasur 1	Berhasil	Berhasil	Kasur 1	Berhasil	Berhasil
Kasur 2	Berhasil	Berhasil	Kasur 2	Berhasil	Berhasil
Kasur 3	Berhasil	Berhasil	Kasur 3	Berhasil	Berhasil
Kasur 4	Berhasil	Berhasil	Kasur 4	Berhasil	Berhasil
Kasur 5	Berhasil	Berhasil	Kasur 5	Berhasil	Berhasil
Kasur 6	Berhasil	Berhasil	Kasur 6	Berhasil	Berhasil

Kamar 2			Kamar 5		
Kasur	Jenis Tombol Virtual		Kasur	Jenis Tombol Virtual	
	Bantuan	Darurat		Bantuan	Darurat
Kasur 1	Berhasil	Berhasil	Kasur 1	Berhasil	Berhasil
Kasur 2	Berhasil	Berhasil	Kasur 2	Berhasil	Berhasil
Kasur 3	Berhasil	Berhasil	Kasur 3	Berhasil	Berhasil
Kasur 4	Berhasil	Berhasil	Kasur 4	Berhasil	Berhasil
Kasur 5	Berhasil	Berhasil	Kasur 5	Berhasil	Berhasil
Kasur 6	Berhasil	Berhasil	Kasur 6	Berhasil	Berhasil

Kamar 3			Kamar 6		
Kasur	Jenis Tombol Virtual		Kasur	Jenis Tombol Virtual	
	Bantuan	Darurat		Bantuan	Darurat
Kasur 1	Berhasil	Berhasil	Kasur 1	Berhasil	Berhasil
Kasur 2	Berhasil	Berhasil	Kasur 2	Berhasil	Berhasil
Kasur 3	Berhasil	Berhasil	Kasur 3	Berhasil	Berhasil
Kasur 4	Berhasil	Berhasil	Kasur 4	Berhasil	Berhasil
Kasur 5	Berhasil	Berhasil	Kasur 5	Berhasil	Berhasil
Kasur 6	Berhasil	Berhasil	Kasur 6	Berhasil	Berhasil

Apabila tombol virtual kamar 1 ditekan, maka akan muncul tampilan pada Gambar 10 untuk kamar 1 dan Gambar 12 untuk kamar 2. Peneakan tombol virtual pada setiap kasur di setiap kamar diuji. Hasil pengujian peneakan tombol dapat dilihat pada Tabel 3.



Gambar. 13. Tampilan Notifikasi Hasil Peneakan Tombol *Reset* di Ruang Jaga Perawat

Sesaat setelah perawat datang ke kamar pasien, perawat menekan tombol *reset* yang terletak di samping tempat tidur pasien. Tombol *reset* ini berfungsi untuk mengembalikan tampilan status kamar pasien seperti

semula. Hasil peneakan tombol *reset* pada tampilan *Android* di ruang jaga perawat dapat dilihat pada Gambar 13. Gambar 13 menunjukkan bahwa status kamar pasien dari kamar 1 sampai dengan kamar 6 di lantai 1 kembali seperti semula. Keadaan bantuan = 0 menunjukkan bahwa tidak ada pasien yang menekan tombol panggil perawat. Keadaan darurat=0 menunjukkan bahwa tidak ada pasien yang menekan tombol darurat. Hasil peneakan tombol *reset* pada tampilan status kamar pasien dapat dilihat pada Gambar 14.



Gambar. 14. Tampilan Status Kasur Pasien pada Kamar 2 Setelah Tombol *Reset* Telah Ditekan

Gambar 14 menunjukkan bahwa di kamar 2 tidak ada pasien pada nomor kasur tertentu menekan tombol bantuan kepada perawat. Warna setiap kasur pada tampilan ini kembali ke *default*-nya.

V. KESIMPULAN

Kesimpulan yang diperoleh dari penelitian ini adalah *Android* dapat digunakan untuk sistem pemanggilan perawat di rumah sakit gedung bertingkat. Dua buah *Android* digunakan dalam sistem ini. Sebuah perangkat *Android* diletakkan di ruang jaga perawat dan satu perangkat *Android* lainnya di samping kasur pasien. Sistem pemanggilan perawat diwakili dengan merancang suatu tombol virtual dari suatu aplikasi *Android*. Komunikasi antara kamar pasien dan ruang jaga perawat dapat dilakukan melalui Internet atau *wifi* sehingga mengurangi kebutuhan kabel instalasi di gedung rumah sakit. Namun, biaya yang dibutuhkan lebih besar karena memerlukan sebuah *smartphone Android* untuk setiap kasur. Untuk selanjutnya, penelitian akan dikembangkan dengan tujuan untuk mengurangi biaya pengadaan *smartphone* di masing-masing kasur pasien. Pengurangan biaya dilakukan dengan mengganti *smartphone* tersebut dengan sebuah tombol fisik (*push button*) yang terhubung dengan kabel pada suatu modul mikrokontroler. Setiap kamar pasien memiliki sebuah modul mikrokontroler. Mikrokontroler ini berfungsi sebagai perantara komunikasi *wireless* antara kamar pasien dengan *Android* di ruang jaga perawat.

UCAPAN TERIMA KASIH

Ucapan terima kasih disampaikan kepada pihak-pihak yang telah mendukung berjalannya penelitian ini khususnya para pembimbing dan laboran Universitas Tarumanagara, rekan-rekan angkatan 2013 Teknik Elektro Universitas Tarumanagara.

DAFTAR PUSTAKA

- [1] I. Sayekti, "Bel Pemanggil Perawat Berbasis *Wireless* Menggunakan *Android*," *Jurnal Teknik Elektro Terapan*, vol. 2, pp. 174-180, 2013.
- [2] N. Jamal, "Sistem Pengendali *Nurse Station* Menggunakan PPI 8255," *Jurnal TELEMATIKA MKOM*, vol. 4, pp. 133-140, September 2012.
- [3] D. Evans, "The Internet of Things How the Next Evolution of The Internets is Changing Everything: Cisco Internet Business Solutions Group (IBSG).", ed. San Jose: CISCO, 2011.
- [4] E. D. Meutia, "Internet of Things - Keamanan dan Privasi," *Seminar Nasional dan Expo Teknik Elektro Tahun 2015*, 2015.
- [5] O. Vermesan and P. Friess, *Internet of Things - From Research and Innovation to Market Deployment*. Aalborg: River Publisher, 2014.
- [6] N. S. H., "Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis *Android*," ed Bandung: Informatika Bandung, 2011.
- [7] M. Ichwan and F. Hakiky, "Pengukuran Kinerja *Goodreads Application Programming Interface (API)* pada Aplikasi *Mobile Android* (Studi Kasus Untuk Pencarian Data Buku)," *Jurnal Informatika*, vol. 2, pp. 13-21, Mei - Agustus 2011.
- [8] Z. P. Juhara, *PANDUAN LENGKAP PEMROGRAMAN ANDROID*. Yogyakarta: Penerbit Andi Yogyakarta, 2016.
- [9] S. V. Kartalopoulos, *Free Space Optical Networks For Ultra-Broad Band Service*. New Jersey: John Wiley & Sons, Inc.
- [10] P. Gregory, *CISSP Guide to Security Essentials 2nd Edition*. Boston: Cengage Learning, 2015.
- [11] T. Dean, *Network+ Guide to Networks 6th ed*. Boston: Course Technology, 2013.
- [12] J. F. DiMarzio, *Android: A Programmer's Guide*. New York: McGraw-Hill, 2008.