

# Aplikasi *Caring Assistance* Untuk Manula Berbasis Android Dan Raspberry Pi

Fenny Destasia Chan, Andreas Handoyo, Justinus Andjarwirawan  
Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Kristen Petra  
Jl. Siwalankerto 121 – 131 Surabaya 60236  
Telp. (031) – 2983455, Fax. (031) – 841765

E-mail: fennydestasia@gmail.com, handoyo@petra.ac.id, justin@petra.ac.id

## ABSTRAK

Kehadiran panti werdha merupakan tempat yang dipilih oleh sebagian keluarga manula yang memiliki aktivitas yang padat. Panti werdha dianggap sebagai rumah baru atau lingkungan baru bagi para manula. Untuk itu, perlu perhatian khusus untuk manula yang tinggal di panti werdha. Layanan 24 jam oleh para staf pendamping dan sistem pencatatan yang terjadwal merupakan cara untuk memantau kesehatan manula.

Namun pada kenyataannya, tidak semua perawat manula dapat terjaga selama 24 jam. Oleh karena keterbatasan kemampuan manusia yaitu tertidur ataupun tidak selalu berjaga didekat manula Hal ini bisa berbahaya apabila manula membutuhkan bantuan di tengah malam. Melihat permasalahan yang ada maka dibutuhkan sebuah teknologi yang dapat menjadi perantara antara manula dan perawat manula. Dan berfungsi sebagai asisten dalam menjaga kesehatan manula.

Aplikasi *Caring Assistance* menggunakan Android dan Raspberry Pi. Manula yang membutuhkan bantuan, dapat menekan *switch* kemudian akan muncul notifikasi di *smartphone* perawat dan menampilkan video *live streaming*. Perawat manula bisa lebih tanggap dalam memberikan bantuan secara darurat. Aplikasi ini dapat membantu mengingatkan perawat manula mengenai jadwal obat manula, dan juga menyimpan daftar rumah sakit dan apotek terdekat dengan menggunakan fasilitas Google Places API pada *smartphone* pengguna. Dari hasil pengujian aplikasi dan Raspberry Pi dapat disimpulkan bahwa aplikasi dapat menjawab kebutuhan perawat manula dalam merawat kesehatan manula dan memantau keadaan manula menggunakan Raspberry Pi.

**Kata Kunci:** Manula, Panti Werdha, Android, Raspberry Pi, Firebase Real Time, *Live Streaming*, Google Place, Google Cloud Messaging

## ABSTRACT

*The presence of Nursing Home is a place chosen by some elderly families who have a solid activity. The Nursing Home is considered a new home or new environment for the elderly. For that it needs special attention to the elderly living in the Panti Werdha. A 24-hour service by accompanying staff and scheduled listing systems is a way to lead an elderly health. But in the belief, not all elderly nurses can be maintained for 24 hours. Because of the limitations of human ability that is sleeping or not, always guard near the elderly It can be dangerous seniors need help in the middle of the night.*

*Seeing the problem, then needed a technology that could be between seniors and elderly nurses. And serves as a helper in maintaining the health of the elderly. Caring Assistance app using Android and Raspberry Pi. Seniors who need help, can press the switch later will appear notification to mobile run and display live streaming video. Elderly nurses can be more responsive in providing emergency assistance. This application can help seniors with the schedule of elderly drug, as well as make a list of the nearest hospital and pharmacy using the Google Places API facility on the user's mobile phone.*

*From the results of application testing and Raspberry Pi can be concluded the application can answer the needs of elderly nurses in the elderly health care and the state of the elderly using Raspberry Pi.*

**Keywords:** Elderly, Nursing Home, Android, Raspberry Pi, Firebase Real Time, *Live Streaming*, Google Place, Google Cloud Messaging

## 1. PENDAHULUAN

Manula merupakan kelompok umur pada manusia yang telah memasuki tahapan akhir dari fase kehidupannya [9]. Menurut [2] kelompok yang dikategorikan manula ini akan mengalami penurunan fungsi sistem motorik (otot dan rangka), antara lain berkurangnya daya tumbuh dan regenerasi, kemampuan mobilitas dan kontrol fisik, dan semakin lambatnya gerakan tubuh. Menurut ICN (*International Council of Nurses*) jumlah manula di dunia terus bertambah sebanyak satu juta manula tiap bulannya.

Populasi manula semakin meningkat setiap tahunnya dan sebagai akibat peningkatan tersebut tersebut adalah kebutuhan manula akan tempat tinggal. Kehadiran Panti Werdha kini sering dipilih sebagai *alternative* tempat tinggal manula, pilihan tersebut dilakukan terutama oleh keluarga dengan aktivitas yang padat. Panti Werdha merupakan panti yang di dalamnya ada *personal care professional*, dan hanya usia lanjut usia yang lemah yang tidak mampu mengurus dirinya sendiri serta mempunyai kondisi ketergantungan dapat diterima atau dirawat.

Di dalam Panti Werdha, manula harus memperoleh berbagai fasilitas penunjang, layanan 24 jam oleh para staf pendamping, rutinitas yang terjadwal, dan hiburan yang dibutuhkan sesuai keinginan dari para manula. Namun pada realitanya, tidak semua staf pendamping menjaga manula selama 24 jam penuh, oleh karena keterbatasan kemampuan manusia yaitu tertidur ataupun tidak berjaga didekat manula. Hal tersebut bisa berbahaya karena apabila manula membutuhkan bantuan secara *urgent*, perawat tidak mengetahui kondisi manula tersebut. Dan juga

saat terjadi kondisi *urgent* untuk menghubungi rumah sakit atau apotek terdekat, perawat manula kebingungan untuk mencari kontak nomor *telephone* atau alamat rumah sakit dari *contant handphone*. Oleh karena itu dibutuhkan teknologi yang dapat digunakan untuk membantu perawat dalam memberikan bantuan perhatian bagi manula. Teknologi yang dibutuhkan manula adalah “*assistive technologies*” atau teknologi pembantu yang tidak sepenuhnya digunakan oleh manula akan tetapi lebih diperuntukkan kepada perawat dari manula tersebut.

Melihat permasalahan yang ada maka dibutuhkan teknologi yang dapat membantu perawat dalam memberikan bantuan perhatian bagi manula. Aplikasi *Caring Assistance* untuk perawat manula berbasis Android dan Raspberry PI, berguna untuk membantu perawat untuk memantau kesehatan manula, mengingatkan perawat manula mengenai jadwal obat manula, perawat bisa lebih tanggap saat memberikan bantuan secara darurat secara langsung saat manula menekan *panic button*, melihat kondisi manula yang membutuhkan bantuan secara *live streaming*, dan juga menyimpan daftar rumah sakit dan apotek terdekat dengan menggunakan fasilitas Google Places API dan GPS (*Global Positioning System*) pada *smartphone*.

## 2. TINJAUAN PUSTAKA

Saat ini Android adalah sistem operasi yang paling banyak digunakan di perangkat *mobile*. Gartner mengatakan penjualan *smartphone* tumbuh 20% pada kuartal ketiga 2014. Berdasarkan analisis Gartner, 83,1% perangkat *mobile* menggunakan sistem operasi Android. Android banyak digunakan oleh pengembang untuk membuat aplikasi mereka karena sistem operasi *platform* yang bebas dan terbuka. Dalam konteks penyiaran informasi menggunakan aplikasi Android, masih banyak penelitian yang telah dilakukan seperti pemandu wisata [1, 2], pendidikan [4, 5, 6], dan kesehatan [7, 8]. Yang menggunakan ponsel untuk menyiarkan informasi spesifik dan penting ke / dari pengguna. Biasanya aplikasi *handphone* mereka berbasiskan pada sistem operasi Android. Karena pertumbuhan sistem operasi Android, *developer* kini bisa membuat aplikasi Android dengan mudah dengan *cross platform framework* [9].

Firestore adalah BaaS (*Backend as a Service*), yang saat ini dimiliki oleh Google. Firestore ini merupakan solusi yang ditawarkan oleh Google untuk mempermudah pekerjaan *Mobile Apps Developer*. Firestore menyediakan library untuk berbagai *client platform* yang memungkinkan integrasi dengan Android, iOS, JavaScript, Java, Objective-C dan Node aplikasi Js dan dapat juga disebut sebagai layanan DbaaS (*Database as a Service*) dengan konsep realtime. Dengan adanya Firestore, *apps developer* bisa fokus mengembangkan aplikasi tanpa harus memberikan *effort* yang besar untuk urusan *backend*. Firestore Cloud Messaging dan Notifications: FCM (*Firestore Cloud Messaging*) adalah layanan yang diberikan oleh Firestore untuk menggantikan GCM (*Google Cloud Messaging*).

## 3. ANALISA DAN DESAIN SISTEM

Langkah pertama untuk membuat aplikasi *Caring Assistance* adalah melakukan *survey* di Panti Werdha Bhakti Ibu. *Survey* tersebut dilakukan guna untuk mengetahui beberapa hal sebagai berikut: *detail* jadwal kegiatan manula seperti jam manula bangun pagi, mandi, berkumpul di ruang kumpul, dan tidur. Kemudian dilanjutkan dengan wawancara mengenai jumlah pekerja, jumlah manula, dan jumlah *shift* pekerja yang yang

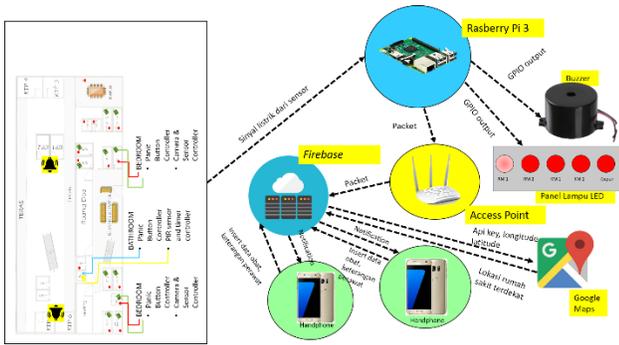
dimiliki. Setelah itu, penelitian ini dilanjutkan dengan menganalisa beberapa permasalahan dari hasil *survey* untuk mendapatkan ide pembuatan *design database*. Penelitian ini dibangun karena kebutuhan perawat manula dalam memantau dan menjaga kesehatan manula. Penting bagi penelitian ini untuk mengetahui konsep dengan struktur database *Caring Assistance* sebagai kerangka kerja pertama, karena merupakan fondasi awal sebelum memulai pembuatan aplikasi.

Dilanjutkan dengan tahap konsultasi dengan *coordinator* Panti Werdha untuk memastikan sistem informasi yang diperlukan untuk membantu perawat manula sesuai dengan kebutuhan, sehingga dapat digunakan untuk koordinator dan penjaga manula di Panti Werdha dalam jangka lama. Desain ERD (*Entity Relationship Diagram*) dan relasinya dibuat berdasarkan analisis konsultasi dengan *coordinator* dan penjaga manula.

Setelah selesai melakukan desain *database*, peneliti mulai membuat desain *interface*. Desain *interface* adalah pembuatan halaman dasar dan relasi desain halaman. Setelah menyelesaikan desain antarmuka, penelitian berlanjut dengan mengadakan pertemuan dengan *coordinator* dan penjaga manula untuk memberikan beberapa pilihan alternatif desain *interface mobile* dari aplikasi *Caring Assistance*, dan *coordinator* panti untuk memilih desain yang paling sesuai untuk kebutuhan *Caring Assistance*. Setelah itu, revisi *interface* dilakukan dan dilanjutkan dengan pertemuan dengan *client* untuk menunjukkan *interface* yang diharapkan paling sesuai untuk kebutuhan *Caring Assistance*, dan juga menunjukkan kemajuan dan menerima kritik dan saran.

Setelah *interface* mengalami fase revisi, penelitian ini berlanjut dengan fase *coding*. Setelah melakukan tahap *coding*, maka akan dilakukan proses pengujian dan kemudian setelah menemukan *error*, maka *coding* akan di revisi. Selanjutnya, penelitian ini lebih difokuskan pada tahap *finishing* karena pada tahap ini akan dilakukan *testing* untuk mencoba semua fitur yang ada, dan tetap melanjutkan tugas penulisan *manual analytical writing*. *Client*, untuk tahap finishing kali ini adalah perawat manula, karena tim ingin mengkonfirmasi informasi yang ada dalam aplikasi apakah sesuai dengan kebutuhan karena kemungkinan selama proses pembuatan aplikasi *Caring Assistance* ada tambahan informasi yang dibutuhkan. Aplikasi *Caring Assistance* menggunakan Android Studio sebagai *Platform* karena Android Studio memiliki banyak fitur yang memudahkan fitur kerja, Android Studio juga memiliki banyak perpustakaan yang dijalankan membuat aplikasi ini. Selain memiliki banyak perpustakaan untuk digunakan.

Selain itu, kemudahan membangun aplikasi berbasis Android adalah alasannya, aplikasi ini dibangun menggunakan Android Studio. Dan saat kita mengkompilasi dan menjalankan aplikasi kita, file APK secara otomatis terbentuk sehingga kita tidak perlu bekerja dua kali untuk membangun aplikasi. Pengguna menggunakan aplikasi *Caring Assistance* yang bisa diakses melalui *smartphone* berbasis Android. Aplikasi *mobile Caring Assistance* dibuat dengan menggunakan Android Studio. Penggunaan aplikasi dimulai dengan pemasangan aplikasi. Setelah itu, pengguna bisa langsung masuk ke halaman utama (*main activity*) tampilan yang berisi halaman untuk menambahkan data *master* seperti kamar, obat, manula dan jadwal minum obat.



Gambar 1. Design Arsitektur

## 4. IMPLEMENTASI SISTEM

### 4.1 Application Testing

Pengujian pertama kali dilakukan dengan menggunakan jaringan internet yang sama dengan laptop. Pada pengujian ini, menggunakan wifi IP address dari laptop yang berfungsi sebagai server local. IP laptop diperoleh dengan mengetik ipconfig pada command prompt. Untuk pengguna aplikasi yang belum memiliki account harus mendaftarkan diri terlebih dahulu ke halaman registrasi, pengguna diminta untuk mengisi username, email, password, nama dan nomor handphone. Data pengguna, akan disimpan kedalam Firebase Database. Pengujian untuk halaman registrasi dapat dilihat pada Gambar 2.

Gambar 2. Halaman Registrasi

Untuk pengguna yang sudah memiliki akun dapat melakukan login. Data yang dimasukkan berupa email, password, ip address untuk camera activity dan ip address untuk akses local database. Setelah pengguna berhasil melakukan login, sistem akan menyimpan token handphone pengguna yang berfungsi sebagai autentikasi account. Kemudian sistem akan mengirimkan token dari smartphone pengguna. Dengan menggunakan ip address local untuk database, contoh: 192.168.1.100/skripsi/token.php. Data token disimpan di phpmyadmin. Tampilan pengujian pengambilan token dapat dilihat pada Gambar 3.

| id | user_id                      | user_token  |
|----|------------------------------|---|
| 1  | rp5y6ZASgihViAnzmK5ME3i1ZQ73 | el3q6YKyFPs:APA91bFc16z2zu25N-nuP5btLuxf40NdgMciQ...  |
| 2  | r4Tawc64R5dYx91YFU4U7K6VLQ82 | fkdq-e7AnSE:APA91bEWhC3YyEKDarCfbW0McAEuoRVzBa5Vtq... |

Gambar 3. Database Token PHP

Setelah pengguna berhasil melakukan login, sistem akan menampilkan halaman utama aplikasi yang berisikan navbar dengan menu home, manula, bantuan dan profil. Di halaman menu home terdapat beberapa fitur yang dapat dijalankan oleh pengguna yaitu menambah kamar, menambah obat, menambah jadwal obat, dan melihat list obat sesuai dengan parameter waktu (pagi, siang, sore, dan malam). Tampilan untuk halaman home aplikasi dapat dilihat pada Gambar 4.

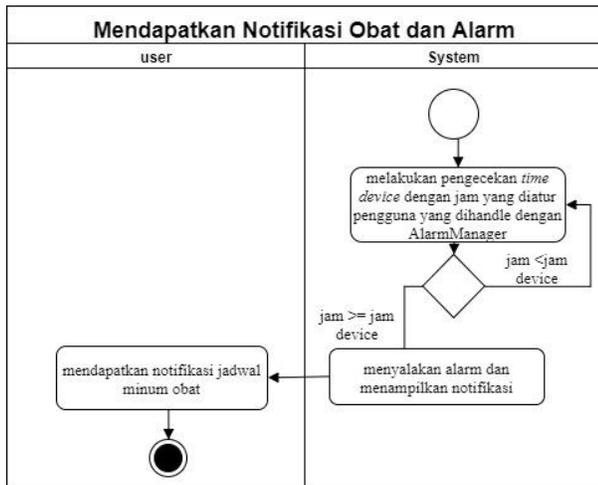


Gambar 4. Tampilan Halaman Home

Pengguna dapat melakukan pengelolaan penjadwalan minum obat untuk manula. Pengguna dapat menekan tombol Tambah Jadwal di halaman home. Kemudian pengguna dapat memasukkan data sesuai dengan form yaitu berupa nama manula, nama obat, interval jadwal minum obat (contoh: satu kali sehari, dua kali sehari, dan lain-lain), tanggal mulai minum obat, dan tanggal akhir minum obat. Kemudian pengguna dapat menambahkan data jadwal obat dengan menekan tombol Save. Jadwal jam yang dimasukkan oleh pengguna akan menyala sesuai dengan jam yang telah dipesan oleh pengguna. Tahap pertama pengujian untuk notifikasi alarm medication yaitu pengguna menginputkan jam dan parameter jangka waktu minum obat.

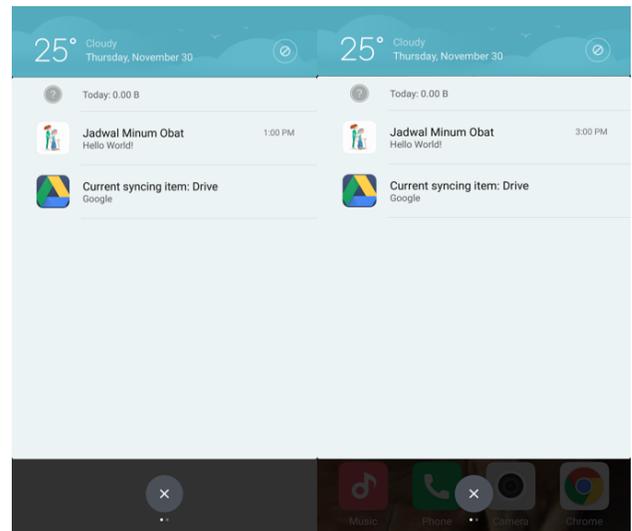
Gambar 5. Tampilan Penambahan Jadwal Minum Obat

Design flowchart dapat untuk mendapatkan notifikasi obat dan alarm dapat dilihat pada Gambar 6.



Gambar 6. Activity Diagram Notifikasi

Apabila jam jadwal minum obat sesuai dengan jam yang ada pada *smartphone* pengguna maka, aplikasi akan memberikan notifikasi dengan informasi mengenai nama obat dan nama manula. Pada Gambar 4, dilakukan pembuatan jadwal minum obat dengan jangka waktu 2 kali sehari yaitu pada jam 13:00 dan 15:00. Pengujian notifikasi untuk jadwal minum obat dengan jangka waktu 2 kali sehari dan jam yang ditentukan jam 13:00 dapat dilihat pada Gambar 7.



Gambar 7. Notifikasi dan Alarm Jadwal Minum Obat

Pengujian aplikasi, dilakukan dengan menampilkan notifikasi *emergency call* ke *smartphone* ketika manula menekan *panic button* dengan sasaran menyalakan LED dan *buzzer*. Kemudian pengujain dengan mendapatkan notifikasi dan pengguna mendapatkan *alarm* ketika jadwal minum obat sesuai dengan jam yang ada pada *smartphone* pengguna.

Pengujian aplikasi dapat dilihat pada Tabel 1.

Tabel 1. Hasil Percobaan Aplikasi

| Deskripsi  | Sasaran   | Pengamatan   | Kesimpulan |
|--|---|--|------------|
| Mendapatkan notifikasi ketika manula menekan tombol atau pir sensor mendeteksi objek   | <ul style="list-style-type: none"> <li>Menyalakan LED dan buzzer</li> <li>Mengirimkan notifikasi ke <i>mobile</i> pengguna</li> </ul> Dengan data masukkan: Nilai true dari Raspberry Pi  | Menampilkan notifikasi <i>emergency call</i> ke android ketika terjadi <i>action</i> pada Raspberry Pi | Berhasil   |
| Mendapatkan notifikasi dan mendapatkan alarm ketika alarm untuk jadwal minum obat sesuai dengan jam yang ada pada <i>device</i> pengguna | <ul style="list-style-type: none"> <li>Mengirimkan notifikasi ke <i>mobile</i> pengguna</li> </ul> Dengan data masukkan: <ul style="list-style-type: none"> <li>Parameter waktu minum obat</li> <li>Jam minum obat</li> <li>Tanggal mulai minum obat</li> <li>Tanggal akhir minum obat</li> </ul> | <ul style="list-style-type: none"> <li>Menampilkan notifikasi jadwal minum obat</li> </ul>             | Berhasil   |

Pengujian dilanjutkan dengan Google Place dan Google Place Detail menggunakan dua parameter untuk menampilkan lokasi.

Pengguna ingin menampilkan rumah sakit dan dokter terdekat dapat memilih *checkbox* rumah sakit dan dokter maka sistem akan menampilkan *Toast* yang berisikan informasi bahwa sistem sedang melakukan *load* untuk menampilkan lokasi rumah sakit dan dokter terdekat sesuai dengan radius parameter yang disesuaikan. Tampilan untuk rumah sakit terdekat dapat dilihat pada Gambar 8.



**Gambar 8. Menampilkan Tempat Darurat Terdekat Dengan Dua Parameter**

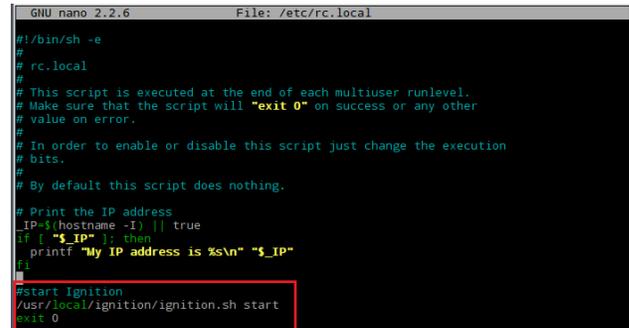
Pengujian melalui jaringan *local*. Langkah ini adalah menguji kamera sudah dapat diakses secara *online* melalui jaringan *local*. Cara pengujian yaitu dengan mengetik IP Address dari Raspberry Pi pada browser: 192.168.1.105 merupakan IP Address dari Raspberry Pi dan 8081 merupakan *port* yang digunakan untuk kamera. Tampilan *live stream* dapat dilihat pada Gambar 9.



**Gambar 9. Tampilan Live Streaming**

## 4.2 Hardware Testing

Tahapan awal melakukan pengujian pada *hardware* yaitu menggunakan sebuah *computer* yang berfungsi sebagai perantara antara dengan *mini PC* Raspberry Pi. Apabila *mini PC* Raspberry Pi sudah tersambung dengan wifi, maka pengguna dapat menjalankan *script* tanpa menggunakan komputer, dan *script run on startup* dengan mengubah sudo nano /etc/rc.local. Gambar untuk perintah *syntax* dapat *run on startup* dapat dilihat pada Gambar 10.



**Gambar 10. Syntax run on startup Raspberry Pi**

Kemudian dilanjutkan dengan instalasi rangkaian dengan *mini PC* Raspberry Pi di Panti Werdha. Proses instalasi dapat dilihat pada Gambar 11.



**Gambar 11. Implementasi Raspberry Pi**

Pengujian rangkaian *mini PC* Raspberry Pi di bagi menjadi dua bagian yaitu pada gambar sebelah kiri keadaan dimana manula belum menekan *switch* dan keadaan disebelah kanan yaitu manula menekan *switch* dan pir sensor mendeteksi ada pergerakannya. Hasil pengujian dapat dilihat pada Gambar 12.



**Gambar 12. Pengujian button emergency call**

### 4.3 Pengujian Kuesioner

Uji kuesioner ditunjukkan kepada penjaga manula yang ada Panti Werdha. Pengujian kuesioner adalah untuk memberikan kuesioner tentang kepuasan pengguna aplikasi Caring Assistance kepada koordinator dan penjaga manula. Uji kuesioner diberikan dengan memberikan tingkat kepuasan berupa angka. Perhitungan dilakukan dengan mengkorelasikan antara nilai item dan nilai total item. Standar kategori dibagi menjadi 5 kategori dengan nilai: tidak ada korelasi, korelasi sangat lemah, korelasi cukup, korelasi kuat, korelasi sangat kuat, korelasi sempurna.

Untuk menilai performa aplikasi Caring Assistance, pertanyaan yang digunakan adalah sebagai berikut:

- Apakah desain aplikasi sudah sesuai dengan yang diharapkan?
- Apakah fungsi sistem tersebut sudah sesuai dengan kebutuhan?
- Informasi yang dihasilkan sangat diperlukan untuk menunjang kegiatan di Panti Werdha sehari-hari?
- Apa yang Anda lakukan sehari-hari berhubungan dengan sistem tersebut?
- Apa anda bisa dengan mudah mengimplementasikan atau menggunakan sistem tersebut?

### 5. KESIMPULAN

- Berdasarkan hasil kuesioner 11 dari 12 responden menjawab aplikasi ini sesuai dengan kebutuhan. Tabel penilaian terhadap penggunaan aplikasi dapat dilihat pada Tabel 2.

**Tabel 2. Penilaian Terhadap Penggunaan Aplikasi**

| Indikator                      | 1 | 2 | 3 | 4 | 5 |
|--------------------------------|---|---|---|---|---|
| Tampilan Aplikasi Sudah Sesuai |   |   | 1 | 2 | 9 |
| Fungsi sistem                  |   |   |   | 8 | 4 |
| Fitur informasi                |   |   |   | 3 | 9 |
| Aplikasi menjawab kebutuhan    |   |   | 1 | 4 | 7 |
| Kemudahan menggunakan aplikasi |   |   |   | 8 | 4 |

- Fitur pengelolaan data dapat berjalan dengan baik, dibuktikan dengan pengujian untuk menambah, mengubah dan menghapus data
- Fitur notifikasi untuk *emergency call* dan *alarm medication*, dapat tampil di *handphone* pengguna sesuai dengan waktu yang di atur
- Pengujian untuk rangkaian Raspberry PI di uji menggunakan beberapa panjang kabel yang berbeda dan dari hasil pengujian tersebut dapat disimpulkan pengiriman data dengan menggunakan panjang kabel 20 meter dapat berjalan dengan baik. Dibukti dengan lampu *led*, *buzzer* dan pengiriman data ke Firebase yang
- Alarm untuk penjadwalan obat dengan jangka waktu tertentu berjalan dengan sangat baik. Oleh karena pengujian alarm

dengan jangka waktu tanggal dan interval waktu misalnya 2 kali sekali sehari, 3 kali sehari, dll.

- Fitur menampilkan tempat darurat terdekat berjalan dengan baik dengan menambahkan detail lokasi
- Fitur *live streaming* dapat berjalan dengan baik, dibuktikan dengan pengujian manula yang menekan *switch* untuk menjalankan perintah *emergency call*. Dan hasil video live streaming tersebut dapat disimpan di dalam *handphone* pengguna.

### 6. DAFTAR PUSTAKA

- [1] A. Handoyo, J. Andjarwirawan, S Sunaryo and R. Lim, *Heroic Battle of Surabaya Application Based on Android*, *Journal of Engineering and Applied Sciences*, Vol. 9, No. 12. 2014. pp. 2396-2403
- [2] A.Handoyo, D.S. Frederick, J. Andjarwirawan, *Broadcast learning system using multicast address*, Proc. 2nd International Conference on Education Technology and Computer. 2010. doi: 10.1109/ICETC.2010.5529504
- [3] Azizah, A.N . 2016. *Desain Panti Sosial Tresna Wredha Abiyoso Sleman, Yogyakarta*. Panti Sosial Tresna Werdha di Kabupaten Magelang Dengan Pendekatan Konsep Home, Semarang: Universitas Negeri Semarang, 2.
- [4] Downey, A. B. 2012. *Think Python*. Sebastopol, California: O'REILLY.Google Developers (18 Apr. 2017.).
- [5] The Google Places API Web Service | Google Places API Web Service | Google Developers. Google Developers. Retrieved from <https://developers.google.com/places/web-service/intro>
- [6] F. Jia, X. Cheng, Z. Duan, "Analyzing the Activity Areas of Non-resident Tourists of Shanghai Expo using Cellular Phone Data" Proc. COTA International Conference of Transportation Professionals (CICTP), 2013, pp. 1136 – 1145, doi: 10.1016/j.sbspro.2013.08.130
- [7] Lee, Wei-Meng. (2012). *Beginning Android 4 Application Development*. Retrived October 20. 2017. from <http://www3.ul.ie/ictl/Android.pdf>
- [8] Murtiwiayati dan Glenn. 2013. *Rancang Bangun Aplikasi Pembelajaran Budaya Indonesia Untuk Anak Sekolah Dasar Berbasis Android*.Depok: Universitas Gunadarma
- [9] J. Cinnamon, S.K. Jones, W.N.Adger, *Evidence and future potential of mobile phone data for disease disaster management*, *Geoforum*, Vol. 75, 2016, pp. 253–264
- [10] R. Lim, P.S. Wijaya, J. Andjarwirawan, A. Handoyo, R. Intan, *Web services extension for accessing quiz on Moodle Mobile application*, *Jurnal of Engineering Applied Sciences* Vol. 9, No. 12. 2014. pp. 2912-2915
- [11] S. Phithakitnukoon, T. Horanont, A. Witayangkurn, R. Siri, Y. Sekimoto, R. Shibasaki, "Understanding tourist behavior using large-scale mobile sensing approach: A case study of mobile phone users in Japan" *Pervasive and Mobile Computing* 18. 2015. pp. 18–39, doi: 10.1016/j.pmcj.2014.07.003