**IMPLEMENTASI VISUALISASI PENDETEKSI TANK BERBASIS ANDROID**

Indar Jaya Ismail, Mayor Inf Imam Azhar, S.ST.,M.T.1, Letkol Arh Desy Derius Minggu, S.T, M.T.2

Diploma 4 Angkatan Pertama Jurusan Teknik Telekomunikasi Militer Politeknik Kodiklatad

Desa Pendem, Kecamatan Junrejo, Kota Batu, Jawa Timur, Indonesia

[kommil2509@gmail.com,](mailto:lukyrenaldi@gmail.com) indarjaya94@gmail.com

Abstrak — Tank merupakan kendaraan tempur lapis baja beroda rantai yang digunakan untuk mendukung pasukan Kavaleri dalam operasi tempur. Manuver tank dimedan operasi selalu diwaspadai, karena tank memiliki daya kejut dan daya tembak yang dapat merusak konsentrasi lawan. Oleh karena itu, arah datangnya tank selalu menjadi faktor dominan untuk diketahui atau dideteksi sejak awal. Dalam melakukan pengintaian masih menggunakan personel untuk mengawasi daerah datangnya tank. Kondisi dalam pengintaian ini membutuhkan konsentrasi, tenaga personel dan logistik yang lebih banyak. Analisis alat pendeteksi yang berupa sensor biasanya diimplementasikan pada media komputer saja. Namun dengan perkembangan teknologi yang bersifat *mobile*, dalam implementasi alat pendeteksi tersebut membutuhkan aplikasi visualisasi guna menampilkan sebuah data informasi objek. Aplikasi ini dapat diimplementasikan dengan menggunakan *smartphone,* dikarenakan dalam penggunaan *smartphone* aplikasi dapat berjalanpadaandroid. Dengan menggunakan metode KNN

Kata Kunci –*Tank, Android, KNN*

*The tank is a chain-armored armored fighting vehicle used to support Cavalry troops in combat operations. Tank maneuvers in the operating area are always on the lookout, because tanks have shock and firepower which can damage the opponent's concentration. Therefore, the direction of the tank's arrival is always a dominant factor to be known or detected early. In conducting surveillance, still use personnel to monitor the area where the tanks come from. Conditions in this surveillance require more concentration, personnel and logistics. Analysis of detection devices in the form of sensors is usually implemented on computer media only. However, with the development of technology that is mobile, in the implementation of the detection device requires visualization applications to display an object information data. This application can be implemented using a smartphone, because in smartphone usage the application can run on Android.* *By using the KNN method*

*Keywords –Tank, Android*

**I. PENDAHULUAN**

* 1. **Latar Belakang.** Telekomunikasi merupakan salah satu bidang dari teknik elektro yang saat ini berkembang pesat. Perkembangannya memberikan dampak yang sangat pesat dalam dunia teknologi pada umumnya dan teknologi alutsista Militer pada khususnya. Penggunaan media sensor suara dan getaran sebagai media pengintaian dan pengamatan. Dalam hal ini sensor tersebut memerlukan adanya media sebagai alat untuk mengimplementasikan hasil dari informasi yang didapat mengenai objek pengintaian. Penggunaan kamera dan alat sensor sebagai alat untuk menangkap gambar objek dan variable lainnya sangat diperlukan, guna melengkapi data pengintaian yang dilakukan oleh pasukan Militer terhadap pergerakan suatu objek, khususnya untuk pendeteksian tank.

Tank merupakan kendaraan tempur lapis baja beroda rantai yang digunakan untuk mendukung pasukan Kavaleri dalam operasi tempur. Manuver tank dimedan operasi selalu diwaspadai, karena tank memiliki daya kejut dan daya tembak yang dapat merusak konsentrasi lawan. Oleh karena itu, arah datangnya tank selalu menjadi faktor dominan untuk diketahui atau dideteksi sejak awal. Dalam melakukan pengintaian masih menggunakan personel untuk mengawasi daerah datangnya tank. Kondisi dalam pengintaian ini membutuhkan konsentrasi, tenaga personel dan logistik yang lebih banyak. Maka hal ini merugikan satuan dalam aspek waktu, logistik dan tenaga. Disamping itu pengintaian dengan personel memiliki resiko yang cukup besar, mulai dari personel yang mengalami kelelahan, hal ini dapat menyebabkan kelalaian sehingga menyebabkan terjadinya korban jiwa. Peneliti memiliki ide untuk menggantikan tugas personel tersebut dengan menggunakan alat pengintai untuk mendeteksi tank secara otomatis. Dalam hal ini yang digunakan dalam objek penelitian adalah tank Leopard 2.

Seiring dengan berkembangnya zaman, penggunaan alat pengintai dapat menjadi salah satu solusi untuk mendeteksi adanya tank. Dalam hal ini, analisis alat pendeteksi yang berupa sensor biasanya diimplementasikan pada media komputer saja. Namun dengan perkembangan teknologi yang bersifat *mobile*, dalam implementasi alat pendeteksi tersebut membutuhkan aplikasi visualisasi guna menampilkan sebuah data informasi objek. Aplikasi ini dapat diimplementasikan dengan menggunakan *smartphone,* dikarenakan dalam penggunaan *smartphone* aplikasi dapat berjalanpadaandroid. *OS* android menyediakan *platform* terbuka bagi para pengembang sehingga dapat membuat aplikasi baru didalamnya tanpa harus membayar lisensi apapun karena sifatnya yang *open source* sehingga lebih mudah untuk memodifikasi programnya.

Berdasarkan permasalahan diatas maka dalam penelitian tugas akhir ini penulis membangun aplikasi yang digunakan untuk mengimplementasikan informasi yang didapat guna mendukung pengunaan alat pendeteksi tank, berjudul “**IMPLEMENTASI VISUALISASI PENDETEKSI TANK BERBASIS ANDROID”**. Aplikasi tersebut diimplementasikan sebagai visualisasi dari informasi yang sudah didapat dan diolah oleh alat pendeteksi untuk mengetahui dan menampilkan posisi, bentuk dan arah pergerakan dari suatu objek melalui *image,* suara, dan getaranyang telah terdeteksi oleh alat tersebut.

Dengan pembuatan tugas akhir ini diharapkan dapat mempermudah dalam pelaksanaan tugas prajurit dimedan pertempuran dan lebih mengetahui kekuatan serta strategi musuh yang akan dihadapi demi tercapainya suatu tugas operasi dalam pertempuran.

* 1. **Rumusan Masalah.** Adapun rumusan masalah dalam merancang dan membuat alat tersebut adalah :
     1. Bagaimana proses mengimplementasikan informasi yang dihasilkan oleh alat pendeteksi, pada smartphone androiddengan 3 variabel yaitu *image,* suara, dan getaran tank ?
     2. Bagaimana mengetahui jenis tank beserta jarak posisi objek ke alat pendeteksi melalui variabel suara dan getaran menggunakan algoritma KNN (*K-Nearest Neighbour*) ?

* 1. **Tujuan Penelitian.** Adapun tujuan dari penelitian ini adalah sebagai berikut :
     1. Mengimplementasikan informasi yang didapat alat pendeteksi berupa menampilkan *image*, suara, dan getaran tank.
     2. Mengetahui jenis objek melalui variabel suara dan getaran menggunakan algoritma KNN.
     3. Mengetahui posisi jarak objek tank dari alat pendeteksi melalui suara dan getaran menggunakan algoritma KNN.
     4. Dapat membangun aplikasi visualisasi pendeteksi tank pada smartphone berbasis android.
     5. Diharapkan dapat mempermudah dalam pelaksanaan tugas prajurit dimedan pertempuran, khususnya dalam pendeteksian tank.
  2. **Batasan Masalah.** Adapun batasan masalah dalam penulisan tugas akhir ini sebagai berikut :
     1. Membahas tentang proses implementasi visualisasi dari data masukan yang diterima oleh alat pendeteksi.
     2. Mengetahui jenis objek melalui suara dan getaran menggunakan algoritma KNN.
     3. Menampilkan jarak objek dari alat pendeteksi melaui suara dan getaran menggunakan algoritma KNN.
     4. Aplikasi hanya dapat berjalan pada smartphone android.
     5. Pengolahan data diterima oleh firebase secara realtime untuk diolah dan dikirimkan ke smartphone android sehingga dapat menampilkan visualisasi dari alat pendeteksi tersebut.
     6. Hasil visualisasi alat pendeteksi menggunakan 3 variabel utama yaitu image, suara, dan getaran tank.
     7. Objek penelitian yaitu tank Leopard 2.
  3. **Metode Penulisan.** Untuk menyelesaikan tugas akhir penulis melaksanakan pendekatan dan metode penulisan yaitu :
     1. Deduktif. Proses pengamatan secara studi kepustakaan dan observasi dilapangan untuk mendapatkan data kualitatif sampai dengan hipotesa.
     2. Induktif. Menggunakan metode eksperimen untuk mengambil data kuantitatif dan melaksanakan pengujian alat di Laboratorium Poltekad.

* 1. **Sistematika Penulisan.** Adapunpenyusunan dalam penulisan tugas akhir ini dengan sistematika penulisan sebagai berikut :
     1. BAB I Pendahuluan.
     2. BAB II Tinjauan Pustaka.
     3. BAB III Metode Penelitian.
     4. BAB IV Hasil dan Pembahasan .
     5. BAB V Kesimpulan dan Saran.
     6. BAB VI Penutup.
  2. **Manfaat Penelitian.** Adapun manfaat yang didapat dalam penelitian tugas akhir ini, yaitu **:** 
     1. Manfaat Bagi Lembaga.
        1. Hasil penelitian ini dapat digunakan oleh satuan Poltekad sebagai referensi sistem mengajar.
        2. Hasil penelitian dapat dikembangkan untuk penelitian selanjutnya.
     2. Manfaat bagi peneliti. Untuk menambah wawasan dalam bidang perancangan aplikasi berbasis android guna mendukung implementasi visualisasi pada alat pendeteksi yang akan dibangun, diharapkan peneliti dapat mengimplementasikan ilmu yang didapat selama perkuliahan.
     3. Manfaat bagi TNI AD.
        1. Membantu pasukan TNI AD dalam pelaksanaan tugas didaerah pertempuran terutama pada saat pengintaian tank musuh, sehingga personel yang melakukan pengintaian akan digantikan oleh alat teknologi yang semakin canggih dan efisien dalam penggunaannya.
     4. Membantu pasukan TNI AD dalam menampilkan sebuah objek tank Leopard 2 dan mengetahui poisisi jarak objek dari alat pendeteksi melalui android dengan input data berupa image, suara, dan getaran tank dengan menggunakan sensor serta kamera.

**II. DASAR TEORI**

* 1. **Penelitian Terdahulu.**
     1. Remote *Monitoring* Berbasis *Smartphone* Android, peneliti Denish Novenda, Agung Nugroho Jati, Erwin Susanto, Ph.D, tahun 2015. Dengan kesimpulan bahwa penggunaan CPU dan RAM pada perangkat Raspberry Pi ketika mengeksekusi program streaming sebesar 4.723% untuk CPU dan penggunaan Random Acces Memory sebesar 0.417%. Hasil pengujian perangkat Raspberry Pi dipengaruhi oleh hardware dari webcam dan driver yang digunakan. Selain itu dipengaruhi oleh pengaturan *Frame Per Second* dan resolusi kamera webcam pada *software streaming* yang digunakan. Selanjutnya adalah pengujian penggunaan CPU dan RAM pada android ketika mengeksekusi program Remo tank sebesar 59.886% untuk CPU dan penggunaan RAM sebesar 8.535%. Hasil pengujian ini dipengaruhi oleh spesifikas*i smartphone* android.

Setelah hasil pen wifigecekkan koneksi *wifi* dilakukan, diperoleh hasil bahwa koneksi dari perangkat *android* ke *server* yang terdapat pada *Raspberry Pi berhasil*. Hasil ini menyatakan dari tes *ping* dengan jumlah *ping* berbeda dan tiap tes *ping* diperoleh 0% paket *loss*. Pengujian ini dipengaruhi salah satunya dari jarak antara perangkat *android* dengan *server* yang berbeda.

* + 1. Aplikasi Pengontrolan Robot Pemantau Dari Udara Berbasis Android, peneliti, Jacquline M.S. Waworunden, Hendra Sayago, pengontrolan robot dari udara berbasis android maka diperoleh kesimpulan :
       1. Pemantauan dilakukan dengan memodifikasi Helikopter *Remote Control* sebagai robot pantau dari udara.
       2. Robot pantau dari udara dapat dibuat dengan memodifikasi Helikopter Remote Control secara wireless dengan menggunakan WiFi yang dapat melakukan pemantauan serta pengambilan gambar.
       3. Robot pantau dari udara dapat di kontrol melalui aplikasi berbasis Android dengan menggunakan modul *Arduino Ethernet Shield* untuk *Arduino Uno* yang menghubungkan *Arduino Uno* dengan *wireless router*. Apabila menggunakan *Arduino Ethernet Shield*, di kendalikan secara *wireless* menggunakan WiFi maka terdapat kendala dalam proses pengiriman data dari android ke Arduino. Hal ini Arduino memproses instruksi tersebut satu detik atau lebih setelah aksi dilakukan, dan selain itu jika hal tersebut di implementasikan pada helikopter maka akan sangat berbahaya untuk helikopter tersebut.
    2. Sistem Pendeteksi Posisi dalam Ruangan Menggunakan Kekuatan Sinyal Wi-Fi dengan Penerapan Algoritma *Cluster Filtered* KNN, peneliti Rizky Ichsan Parama Putra, Waskitho Wibisono, dan Hudan Studiawan. Tahun 2013. Dengan hasil pengamatan selama perancangan, implementasi, dan proses uji coba perangkat lunak yang dilakukan, penulis mengambil kesimpulan sebagai berikut:
       1. Pencocokan sidik jari dengan algoritma K-NN dan penyaringan menggunakan algoritma Cluster Filtered KNN dapat mendeteksi posisi pengguna berdasarkan pencocokan sidik jari dengan memanfaatkan kekuatan sinyal Wi-fi.
       2. Pengiriman data dari computer client ke server sebesar 105 ms per data.
       3. Besarnya dataset yang digunakan berpengaruh terhadap hasil. Untuk 10 data dihasilkan keakuratan 92,16%, 5 data menghasilkan 88,71%, dan untuk 1 data menghasilkan 84,06%. Rata-rata dari dataset juga berpengaruh terhadap keakuratan pendeteksian posisi. Dataset yang dicari nilai rata-ratanya menghasilkan keakuratan yang lebih kecil. Dari uji coba yang dilakukan dengan 10 dataset, diperoleh keakuratan sebesar 80,12%.
       4. Energi dari baterai yang digunakan aplikasi mengalami penurunan yang tidak terlalu signifikan. Dari hasil uji coba selama 120 menit, terjadi penurunan penggunaan energi baterai sebesar 21%.
       5. Jumlah data dapat berpengaruh terhadap kecepatan pengolahan data di server. Semakin banyak data yang diolah, maka kecepatan pengolahan tiap data semakin tinggi. Dari hasil pengujian kecepatan untuk mengolah 1 data sebesar 0,004854 data/ms, 10 data sebesar 0,006527 data/ms dan 50 data sebesar 0,007883 data/ms.
  1. **Landasan Teori.**
     1. ***Main Battle* tank Leopard 2.**

[[1]](#footnote-1)*Main Battle* tank Leopard 2 adalah tank tempur utama *Main Battle* Tankdari Jerman yang dikembangkan oleh *Krauss-Maffei* pada sekitar awal 1970-an dan mulai digunakan pada tahun 1979. Leopard 2 menggantikan Leopard 1 sebagai tank tempur utama Angkatan Darat Jerman Barat (*Bundeswehr*). Pertama kali digunakan Angkatan Darat Jerman pada Perang Kosovo serta pasukan Kanada dan Denmark yang tergabung dalam ISAF di medan tempur Afghanistan. Gambar senjata Tank Leopard 2 tampak seperti gambar dibawah ini :



Gambar 2.1. *Main Battle Tank Leopard 2*

( Sumber : Batalyon Kavaleri 8 Divisi Infantri 2 Kostrad Pasuruan )

Pengembangan utama pada tank ini, memiliki kubah tembak vertikal berlapis baja dan model yang lebih maju serta memiliki kubah tembak menyudut seperti panah dengan *applique armour*. Seluruh model dilengkapi dengan sistem pengontrol penembakan digital dan laser penjejak jarak, meriam utama 120 mm dengan kestabilan tinggi, senapan mesin koaksial, serta kelengkapan untuk melihat dan membidik dalam kegelapan (*night vision*) yang lebih maju.

Bagian depan dan samping turret dipasangi dengan lapis baja tambahan yang berbentuk baja, dan dapat dengan mudah diganti oleh mekanik jika tertembak atau, dapat diganti dengan lapis baja yang lebih canggih. Panel samping dari baja tambahan ini tergantung dan dapat diayunkan ke depan, dibutuhkan apabila mesin akan diganti. Mantlet meriam telah didesain ulang, dan *boks stowage* tambahan dipasang di belakang dan samping *turret*. Bagian dalam *turret* sekarang dipasangi dengan sebuah *spall liner* untuk meningkatkan proteksi awak terhadap serpihan. Sistem stabilisasi dan kendali meriam hidrolik diganti dengan sistem elektrik. Teleskop tambahan FERO Z-18 optik dipindah posisinya di atas mantlet meriam, dan pembidik panoramik PERI-R 17 milik komandan dipindah ke bagian belakang kiri ruang komandan. Pembidik independen komandan juga dikembangkan dengan *chanel thermal* yang ditampilkan dalam sebuah monitor di dalam ruang komandan. *Processesor* data jarak laser dimodifi hingga Leopard 2 sekarang dapat menyerang helikopter dengan amunisi APFSDS-T, dan sistem navigasi GPS dibuat dengan antena GPS terpasang dibelakang atap turret.

Tambahan lapis baja membuat berat tempur Leopard 2 menjadi 59.500 kg, yang sama sekali tidak berpengaruh pada mobilitasnya, karena memang tank ini memang sudah didesain untuk menerima perubahan seperti ini.

Adapun spesifikasi dari tank Leopard 2 dapat dilihat pada tabel dibawah ini sebagai berikut :

Tabel 2.1 Spesifikasi Tank Leopard 2

( Sumber : Batalyon Kavaleri 8 Divisi Infantri 2 Kostrad Pasuruan )

|  |  |  |
| --- | --- | --- |
| **NO** | **Spesifikasi** | **Keterangan** |
| **1** | **2** | **3** |
| 1. | Senjata pokok | 120 mm Rheinmetall smoothbore gun L/44. |
| 2. | Senjata tambahan | Senjata mesin 1 x 12,7 mm dan 1 x 7,62 mm. |
| 3. | Elevasi | – 9 sampai + 20°. |
| 4. | Kecepatan putaran kubah | 360° dalam 9 detik. |
| 5. | Berat kubah | 16 ton. |
| 6. | Alat bidik penembak | RDE EMES 15 dengan thermal channel  dan laser range finder. |
| 7. | Alat bidik | RDE PERI-R17. |
| 8. | Macam munisi | Heat DM12 Kal. 38. |
| 9. | ATGM | Ada. |
| 10. | Pelontar granat asap | 2 x 8 pelontar granat asap. |
| 11. | Alat bidik | VERO Z-18. |
| 12. | Kec. maksimum | 68 km/jam, mundur 31 Km/jam. |
| 13. | Mengarung | 1.200 mm. |
| 14. | Menyelam | 4.000 mm dengan snorkel. |
| 15. | Melangkah | 2.500 mm. |
| 16. | Rintangan tegak | 1.000 mm. |
| 17. | Tanjakan maks | 31º (60%). |
| 18. | Aksi radius | Di jalan 340 km, di medan 220 km, di medan rata-rata 280 km. |

* + 1. **Algoritma KNN (*K-Nearest Neighbour*)** adalah sebuah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut **(Widiarsana, O et al., 2011)**. Algoritma *K-Nearest Neighbor (KNN)* adalah metode yang digunakan untuk mengelompokkan objek berdasarkan contoh pelatihan terdekat di ruang fitur. *K-Nearest Neighbor* merupakan jenis yang paling dasar dari contoh based learning atau lazy learning juga termasuk kelompok instance-based learning. *K-Nearest Neighbor* dilakukan dengan mencari kelompok objek dalam data training yang paling dekat (mirip) dengan objek pada data baru atau data testing. **(Krati,2014)**.

Algoritma *K-Nearest Neighbor* bersifat sederhana, bekerja dengan berdasarkan pada jarak terpendek dari sampel uji (testing sample) ke sampel latih (training sample) untuk menentukan *K-Nearest Neighbor*nya. Setelah mengumpulkan *K-Nearest Neighbor*, kemudian diambil mayoritas dari *K-Nearest Neighbor* (KNN) untuk dijadikan prediksi dari sample uji. *K-Nearest Neighbor* memiliki beberapa kelebihan yaitu tangguh terhadap training data yang noise dan efektif apabila data latihnya besar. Pada fase training, algoritma ini hanya melakukan penyimpanan vektor-vektor fitur dan klasifikasi data training sample. Pada fase klasifikasi, fitur-fitur yang sama dihitung untuk testing data atau yang klasifikasinya tidak diketahui. Jarak dari vektor baru yang ini terhadap seluruh vektor training sample dihitung dan sejumlah k buah yang paling dekat diambil. Titik yang baru klasifikasinya diprediksikan termasuk pada klasifikasi terbanyak dari titik-titik tersebut. Ketepatan algoritma *K-Nearest Neighbor* sangat dipengaruhi oleh ada atau tidaknya fitur-fitur yang tidak relevan atau jika bobot fitur tersebut tidak setara dengan relevansinya terhadap klasifikasi. **(Alfian,2014)**.

Algoritma *K-Nearest Neighbor* (KNN) beberapa karakteristik K-NN adalah sebagai berikut :

K-NN merupakan algoritma yang menggunakan seluruh data latih untuk melakukan proses klasifikasi. Hal ini mengakibatkan proses prediksi yang sangat lama untuk data dalam jumlah yang sangat besar.

Algoritma K-NN tidak membedakan setiap fitur dengan suatu bobot.

Karena K-NN termasuk lazy learning yang menyimpan sebagian atau semua data, K-NN sangat cepat dalam proses pelatihan tetapi sangat lambat dalam proses prediksi.

Hal yang rumit adalah menentukan nilai K yang paling sesuai

Karena K-NN pada prinsipnya memilih tetangga terdekat, parameter jarak juga penting untuk dipertimbangkan sesuai dengan kasus datanya.

Algoritma K-NN menggunakan klasifikasi ketetanggaan sebagai nilai prediksi untuk query data yang baru atau data uji. Dekat atau jauhnya tetangga biasanya dihitung berdasarkan *Euclidean Distance :*

**Rumus 2.1** *Euclidean Distance*

Keterangan :

D(a,b) : jarak skalar dari a dan b

ak, bk : vektor a dan b yang berupa matrik berukuran d dimensi

Algoritma :

Langkah 1 : Tentukan parameter K

Langkah 2 : Hitung jarak antara data uji dengan data latih dengan rumus Euclidean Distance.

Langkah 3 : Urutkan hasil jarak tersebut secara ascending dan tetapkan tetangga terdekat berdasarkan jarak minimum ke-K

Langkah 4 : Dengan menggunakan klasifikasi K-Nearest Neighbor yang paling mayoritas, maka dapat diprediksikan klasifikasi dari data uji.

* + 1. Androidmenurut **Nazaruddin Safaat H. (2012)** Android adalah sebuah sistem operasi untuk perangkat mobile berbasis Linux yang mencakup sistem operasi, middleware dan aplikasi. Android dikembangkan bersama antara Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, NVIDIA yang tergabung dalam OHA (Open Hedset Alliance) dengan tujuan membuat standar terbuka untuk perangkat bergerak (mobile device). Fungsinya sama seperti sistem operasi Symbian di Nokia, iOS di Apple dan BlackBerryOS. Android tidak terikat ke satu merek handphone saja, beberapa vendor terkenal yang sudah memakai Android antara lain Samsung, Sony Ericsson, HTC, Nexus, Motorolla, dan lain-lain.

Android merupakan *platform mobile* yang memiliki tiga hal sebagai berikut.

Lengkap (*complete platform*), karena Android menyediakan banyak *tools* dalam membangun perangkat lunak dan memiliki peluang untuk mengembangkan aplikasi.

Terbuka (open source platform), karena pengembang secara bebas dapat mengembangkan aplikasi dan platform Android disediakan melalui lisensi open source.

Free (free platform), karena Android merupakan platform yang bebas untuk mengembangkan tidak ada royalty untuk pengembangan pola platform Android.

* + 1. Android 6.0 *Marshmalow* tanggal rilis 10 Agustus 2015. Menu aplikasi pada android *Marshmallow* benar-benar dibuat baru. Desainnya membuat *user* merasa naik kelas dari versi sebelumnya dikarenakan desain yang lebih dinamis. Selain itu, ada juga fitur *memory manager* yang memungkinkan *user* mengecek penggunaan *memory* pada tiap aplikasi. Rentan waktu pengecekannya bisa disetel dari tiga jam yang lalu hingga 24 jam sebelumnya. Pembaruan kedua dapat dilihat dari pengaturan *volume*. Pada *Marshmallow*, pengguna bisa mengontrol *volume* yang berbeda-beda pada panggilan, media, dan alarm. Keamanan juga mendapat peningkatan pada versi ini. Google memungkinkan vendor menambahkan sensor pemindai sidik jari sudah didukung *Marshmallow.*



Gambar 2.15 Tamplian Android Marshmallow

( Sumber : Android 6.0 Marshmalow )

Android yang digunakan pada penelitian kali ini adalah android versi 6 dan beberapa fasilitas yang dimiliki oleh Android versi 6 atau *Nougat* ini adalah :

1. ***Memory Manager*.** Fitur ini berfungsi untuk mengatur penggunaan memori atau kapasitas suatu *smarphone*. Apabila *memory manager* membaca ada data yang terlalu besar, maka akan ada notifikasi untuk mengingatkan pihak *user smartphone.*
2. Sensor Sidik Jari (fingerprint). Salah satu fitur terbaik bawaan android 6.0 marshmallow yang berfungsi untuk meningkatkan keamanan privasi para pemilik smartphone.
3. Doze fitur ini mampu mengemat daya batrai dan pemakaian aplikasi. Tentunya ini sangat bermanfaat bagi para user android Marshmallow.
4. Pesan Pada Lock Screen. User bisa membentuk pesan yang majemuk yg akan ditampilkan di waktu smartphone terkunci.
5. Mudah dalam Un-Install aplikasi android bawaan. Perangkat lunak bawaan bisa diuninstall dengan melakukan root di smartphone. Metode lama ini diklaim cukup efektif namun cara ini dianggap relatif berbahaya. Root akan memberi kesempatan di virus buat menyerang karena sistem keamanannya sudah dibongkar. Selain itu apabila root terjadi ketika masa garansi, maka garansi yang berlaku atas smartphone tersebut eksklusif hangus.
6. App Permissions setiap kali install sebuah pop-up akan ada. Pop up ini berisi perjanjian bahwa selesainya pemasangan, pengguna sepakat bahwa perangkat lunak dapat melakukan akses terhadap penyimpanan, hubungan serta beberapa hal langsung lainnya. Pada fitur terbaik bawaan android 6.0 Marshmallow user dapat mengubah pengaturan ini. Sebagai akibatnya akses yang dilakukan hanya akan terjadi saat perangkat lunak dipergunakan.
7. Pencarian Otomatis menggunakan OK GOOGLE. user smartphone dapat menemukan sensasi baru dalam pemakaian smartphone. Dengan mengandalkan sistem Google Now, user smartphone dapat mengtahui beragam isu hanya menggunakan suara kode OK GOOGLE.
8. USB Type-C yang memiliki kelebihan lebih cepat dalam **proses datanya** daripada pendahulunya–USB 2.0 dengan kecepatan transfer data di angka 480 mbps–USB 3.1 sudah mencapai 10 gbps. USB Type-C bisa mengisi daya HP lebih cepat, hingga 2,20V pada arus 5A.

**III. METODE PENELITIAN**

* 1. **Variabel Penelitian.**

1. Variabel bebas/independen (*independent variable*) atau variabel pengaruh adalah kondisi-kondisi untuk menerangkan hubungannya dengan fenomena yang diobservasi. Variabel bebas adalah variabel yang mempengaruhi atau sebab perubahan timbulnya variabel terikat. Dengan kata lain, variabel bebas adalah variabel yang mempengaruhi variabel terikat. Dalam penelitian ini yang berperan sebagai variabel bebas (*predictor*) adalah *image*, suara dan getaran.
   * + 1. *Image*. Digunakan untuk mengidentifikasi objek tank Leopard 2 sebagai tolak ukur. Gambar akan diambil oleh alat pendeteksi secara digital menggunakan kamera dari alat tersebut, kemudian data diolah firebase dan akan dikirimkan ke aplikasi visualisasi android melalui internet, kemudian ditampilkan berupa data *image* serta spesifikasi tank tersebut.
       2. Suara. Sama halnya dengan variable sebelumnya, aplikasi akan menerima masukan data dari alat pendeteksi yang nantinya besaran gelombang suaranya akan divisualisasikan/ditampilkan serta akan diolah kembali menggunakan algoritma K-NN sehingga aplikasi dapat mengetahui objek tersebut adalah tank Leopard 2 serta dapat mengetahui jarak dari objek ke alat pendeteksi yang akan ditampilkan melalui aplikasi android.
       3. Getaran. Data variabel getaran yang didapatkan dari alat pendeteksi akan diolah sebagai masukan dan ditampikan menjadi grafik, serta akan diolah kembali menggunakan algoritma K-NN sehingga aplikasi dapat mengetahui objek tersebut adalah tank Leopard 2 serta dapat mengetahui jarak dari objek ke alat pendeteksi yang akan ditampilkan melalui aplikasi android.
2. Variabel Terikat/tergantung (*dependent variabel*) atau variabel terpengaruh adalah kondisi atau karateristik yang berusaha atau muncul ketika penelitian mengintroduksi, pengubahan atau mengganti variabel bebas. Dalam hal ini variable terikat yang dapat dipengaruhi oleh tiga variable diatas adalah pendeteksian adanya objek berupa tank Leopard 2.
   * + 1. Persentase kecocokan objek tank Leopard 2 terhadap data training yang ada dalam sistem pendeteksi, memberikan kalkulasi tingkat persentase kesamaan bahwa apakah objek yang terdeteksi tersebut tank Leopard 2.
       2. Jarak antara objek dengan alat pendeteksi, memberikan estimasi jarak antara alat pendeteksi dan posisi tank berdasarkan getaran dan suara tank yang terdeteksi.
   1. **Desain Sistem *Software.***
3. Blok diagram *hardware* dan *sofware* dijelaskan pada prinsip kerja rangkaian pada alat yang akan dibuat :
   * + 1. Blok Diagram. Diagram blok mencakup cara kerja dari alat yang dibuat, kemudian karakteristik komponen yang dipergunakan. keseluruhan sistem visualisasi alat pendeteksi tank dapat dilihat pada gambar 3.1.



Gambar 3.1 Blok Diagram

( Sumber : Perancangan )

Pada gambar 3.1 Dijelaskan tentang cara kerja implementasi visualisasi pendeteksi tank dengan memanfaatkan data *image*, suara, dan getaran masukan yang dikirim dari alat pendeteksi, kemudian diterima dan diolah oleh firebase serta dikirim ke smartphone android. Kemudian data suara dan getaran diolah menggunakan metode K-NN untuk mengetahui persentase kecocokan objek tank Leopard 2 serta mengetahui jarak objek tank dari alat pendeteksi.

* + - 1. Cara Kerja Alat Pendeteksi Tank, sebagai berikut :
         1. Identifikasi dimulai saat menjalankan alat.
         2. Raspberry mulai menginisialisasi I/O.
         3. Raspberry on, mulai menginisialisasi servo, kamera Pi, sound sensor dan piezo element.
         4. Raspberry akan memerintahkan kamera Pi untuk mendeteksi data berupa image tank.
         5. Raspberry akan menjalankan sound sensor untuk mendeteksi data suara tank.
         6. Raspberry akan menjalankan piezo element untuk mendeteksi data getaran tank.
         7. File image diolah dalam tensorflow untuk pemodelan data traning.
         8. Opencv berungsi untuk pengolahan citra menggunakan metode ANN, yang akan diolah dengan cara mencocokan citra template dengan data yang telah ditetapkan sebagai data training.
         9. Apabila masih terdapat kesalahan maka data akan kembali dipisahkan hingga tidak terdapat kesalahan, sehingga tank akan dikenali.
         10. Selanjutnya mengambil data suara dan getaran menggunakan sound sensor dan piezo element.
         11. Data suara dan getaran akan dikonversi dari data analog menjadi data digital oleh MCP 3008, sehingga dapat dikirim melaui modem ke firebase.
         12. Sehingga raspberry Pi akan mengirim data image, suara dan getaran melalui modem ke firebase database.
      2. Cara Kerja Implementasi Visualisasi Pendeteksi Tank, sebagai berikut :
         1. Aplikasi visualisasi akan terus berjalan sampai alat pendeteksi mengirim data *image*, suara, dan getaran melalui modem ke *firebase database* bahwa terdeteksi objek tank Leopard 2.
         2. *Firebase* menerima dan mengolah data *image*, suara, dan getaran.
         3. Aplikasi visualisasi mengambil data dari firebase dan diolah di dalam android guna menampilkan data image dan spesifikasi tank Leopard 2.
         4. Data suara ditampilkan di aplikasi visualisasi berupa grafik suara.
         5. Data getaran ditampilkan di aplikasi visualisasi berupa grafik getaran.
         6. Selanjutnya data suara dan getaran akan diolah menggunakan metode KNN untuk mengetahui tingkat persentase kecocokan objek tersebut adalah tank Leopard 2.
         7. Serta data suara dan getaran diolah kembali menggunakan metode KNN untuk mengetahui jarak objek tank dari alat pendeteksi.
      3. Diagram Alir Sistem Alat Pendeteksi Tank, dapat dilihat pada gambar dibawah ini.



Gambar 3.2 Diagram Alir Sistem Alat pendeteksi Tank

( Sumber : Perancangan )

* + - 1. Cara kerja diagram alir sistem alat pendeteksi tank, sebagai berikut :
         1. Menginisialisasi input dan output.
         2. Raspberry Pi on.
         3. Servo on, agar dapat memutar alat pendeteksi 1800 kearah kiri dan kanan, pada saat pendeteksian objek.
         4. Kamera Pi, sound sensor, dan piezo element dalam posisi on,
         5. Raspberry Pi mulai memerintahkan kamera Pi untuk mendeteksi image objek.
         6. Raspberry Pi akan menjalankan sound sensor untuk mendeteksi data suara tank.
         7. Raspberry Pi akan menjalankan piezo element untuk mendeteksi data getaran tank.
         8. Apabila tidak di temukan objek maka sensor akan kembali mendeteksi objek disekitar alat pendeteksi, apabila terdeteksi objek maka akan diolah ke dalam Raspberry Pi.
         9. Tensorflow melakukan pengolahan data training image tank.
         10. Dilanjutkan pemprosesan dalam OpenCV menggunakan metode ANN untuk melakukan pencocokan data image yang terdeteksi dengan data training.
         11. Apabila objek teridentifikasi sebagai tank Leopard 2 maka akan lanjut ke proses selanjutnya, apabila tidak maka alat, kembali mendeteksi image objek.
         12. MCP 3008 mengkonversi data input sensor suara dan piezo element dari data analog ke data digital.
         13. Raspberry Pi mengirim data image, suara dan getaran mengunakan modem ke Firebase guna diterima dan diolah didalam aplikasi visualisasi pendeteksi tank Leopard 2.
      2. Diagram alir sistem implementasi visualisasi pendeteksi tank, dapat dilihat pada gambar dibawah ini.



Gambar 3.3 Diagram Alir Visualisasi Pendeteksi Tank

( Sumber : Perancangan )

* + - 1. Cara kerja diagram alir sistem aplikasi visualisasi pendeteksi tank, sebagai berikut :
         1. *Firebase* menerima dan mengolah data kemudian dikirimkan ke Aplikasi visualisasi pendeteksi tank.
         2. Aplikasi visualisasi pendeteksi tank mengambil data *image,* suara dan getaran dari *Firebase.*
         3. Aplikasi mengenali objek berupa tank Leopard 2 atau bukan.
         4. Apabila tidak mengenali maka kembali ke proses pengambilan data image, suara dan getaran. Dan apabila dikenali maka akan menuju ke proses selanjutnya.
         5. Aplikasi visualisasi pendeteksi tank menampilkan image dan spesifikasi tank serta menampilkan grafik suara dan grafik getaran.
         6. Menghitung data suara dan getaran untuk mengetahui persentase kecocokan objek tank Leopard 2 menggunakan metode KNN.
         7. Tahap selanjutnya adalah mendeteksi jarak posisi tank dari alat pendeteksi menggunakan metode KNN.
         8. Menampilkan hasil visualisasi persentase kecocokan objek serta mengetaui jarak objek tank Leopard 2 dari alat pendeteksi.
      2. DFD (Data Flow Diagram) level 0,keseluruhan alur data sistem pendeteksi tank Leopard 2.



Gambar 3.4 *Data Flow Diagram* level 0

( Sumber : Perancangan )

* + - 1. Keterangan proses DFD (*Data Flow Diagram)* level 0, sebagai berikut :
         1. Inisialisasi I/O dimulai dari memasukkan data inputan ke Raspberry Pi.
         2. Proses pengambilan data image menggunakan kamera Pi. Dilanjutkan dengan pengambilan data suara menggunakan sound sensor, serta data getaran menggunakan piezo element.
         3. Selanjutnya data image, suara dan getaran akan akan dilolah dalam Raspberry Pi.
         4. Untuk data suara dan getaran akan dikonversi dari data analog ke data digital menggunakan MCP 3008.
         5. Data image diolah dalam Raspberry Pi untuk menentukan apakah objek tersebut tank Leopard 2 atau bukan.
         6. Raspberry Pi akan mengirim data image, suara dan getaran menggunakan modem ke firebase database.
         7. Selanjutnya data dari firebase database akan dikirim ke proses visualisasi guna menampilkan data output berupa image dan spesifikasinya, grafik suara, grafik getaran, beserta presentase kecocokan objek tank dan jarak objek tank Leopard 2 dari alat pendeteksi.

Adapun proses data yang terjadi dalam masing-masing alat dan aplikasi visualisasi dapat dilihat pada DFD (Data Flow Diagram) level berikutnya, seperti dibawah ini.

* + - 1. DFD (Data Flow Diagram) level 1, proses inisialisasi alat pendeteksi tank Leopard 2.



Gambar 3.5 *Data Flow Diagram* level 1 alat pendeteksi tank

( Sumber : Perancangan )

* + - 1. Keterangan proses DFD (*Data Flow Diagram)* level 1 alat pendeteksi tank, sebagai berikut :
         1. Inisialisasi I/O dimulai dari memasukkan data inputan ke Raspberry Pi.
         2. Proses pengambilan data image menggunakan kamera Pi. Dilanjutkan dengan pengambilan data suara menggunakan sound sensor, serta data getaran menggunakan piezo element.
         3. Selanjutnya data image akan akan dilolah dalam Raspberry Pi yang menggunakan Tensorflow sebagai media pengolahan data training.
         4. Dilanjutkan pengolahan data image menggunakan OpenCV untuk pengolahan citra image.
         5. Selanjutnya data image akan diolah untuk mendapatkan pendeteksian objek sebagai tank Leopard 2 menggunakan metode ANN.
         6. Setelah objek dikenali sebagai tank Leopard 2 maka data image, suara dan getaran akan dikirimkan menggunakan modem ke firebase database guna pemprosesan selanjutnya di dalam aplikasi visualisasi.
      2. DFD *(Data Flow Diagram)* level 1, proses aplikasi visualisasi pendeteksi tank Leopard 2.

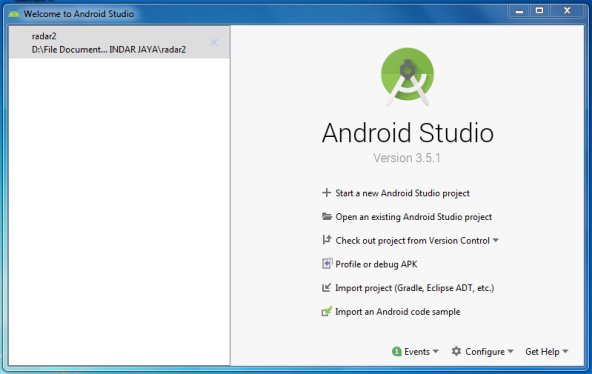


Gambar 3.6 *Data Flow Diagram* level 1 aplikasi pendeteksi tank

( Sumber : Perancangan )

* + - 1. Keterangan proses DFD (*Data Flow Diagram)* level 1 aplikasi visualisasi pendeteksi tank :
         1. Data dari *firebase database* akanditerima oleh aplikasi visualisasi.
         2. Didalam aplikasi visualisasi data image tank akan ditampilkan beserta spesifikasinya secara langsung tanpa melalui proses lebih lanjut.
         3. Kemudian data suara dan getaran akan ditampilkan secara langsung oleh aplikasi visualisasi berupa grafik suara dan grafik getaran.
         4. Data suara dan getaran tank kembali diproses menggunakan metode K-NN, guna mengetahui persentase tingkat kecocokan objek serta mengetahui jarak objek tank Leopard 2 dari alat pendeteksi.
  1. **Perancangan Perangkat Lunak *(Software)*.** Dalam perancangan alat ini, untuk mendukung kinerja perangkat keras dibutuhkan aplikasi/*software* yang berfungsi untuk menampilkan hasil visualisasi tangkapan alat pendeteksi tank, guna menampilkan semua data yang sudah diidentifikasi. Berikut proses pembuatan aplikasi visualisasi alat pendeteksi tank.

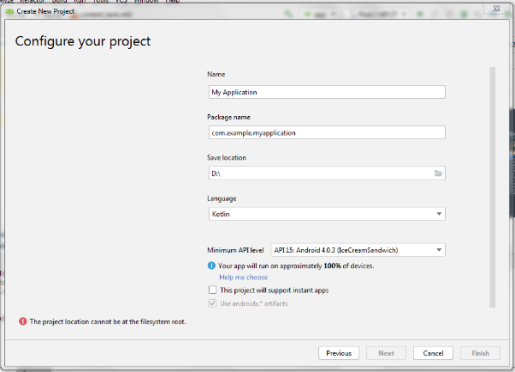
1. Pembuatan project android. Pertama buka android studio yang akan digunakan untuk membuat aplikasi android. Disini peneliti menggunakan android studio versi 3.3. adapun tampilan awal android studio adalah sebagai berikut :
   * + 1. Buat project baru.



Gambar 3.7 Tampilan awal IDE Android Studio

( Sumber : Perancangan )

* + - 1. Berikan nama project yang akan dibuat.

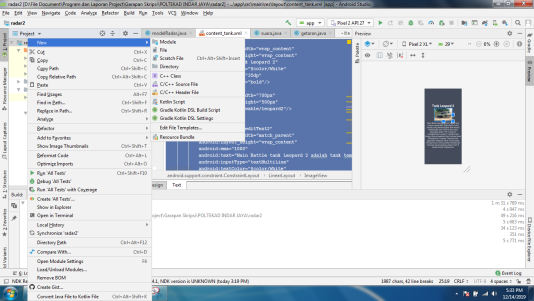


Gambar 3.8 Membuat Nama Project

( Sumber : Perancangan )

Kemudian setelah selesai proses awal mulailah untuk melakukan perancangan aplikasi yang akan dibuat, umumnya dalam pembuatan aplikasi android menggunakan bahasa java untuk mobile.

1. Membuat new file java android**.** Kemudian mulai untuk perancangan class java yang dibutuhkan dengan pilih folder java, kemudian klik kanan, new, pilih class java.

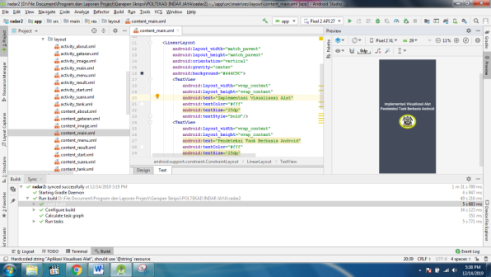


Gambar 3.9 Membuat Java Class

( Sumber : Perancangan )

Kemudian mulai membangun tampilan awal dari aplikasi yang akan dibuat.

* + - 1. Perancangan halaman muka.



Gambar 3.10 Perancangan Halaman Muka

( Sumber : Perancangan )



Gambar 3.11 Tampilan Halaman Muka

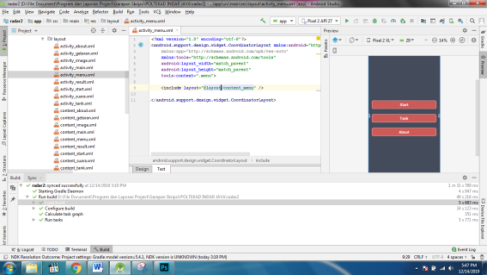
( Sumber : Perancangan )

Dalam perancangan ini mengunakan file xml untuk membuat tampilan yang nantinya berfungsi sebagai *User Interface* pada aplikasi. Perancangan *User Interface* aplikasi dapat dilakukan dengan menggunakan *Pallette Preview* yang ada di *android studio,* listing kode dapat dilihat di Lampiran 1.

Kemudian dilanjutkan dengan halaman berikutnya yaitu masuk ke menu awal.

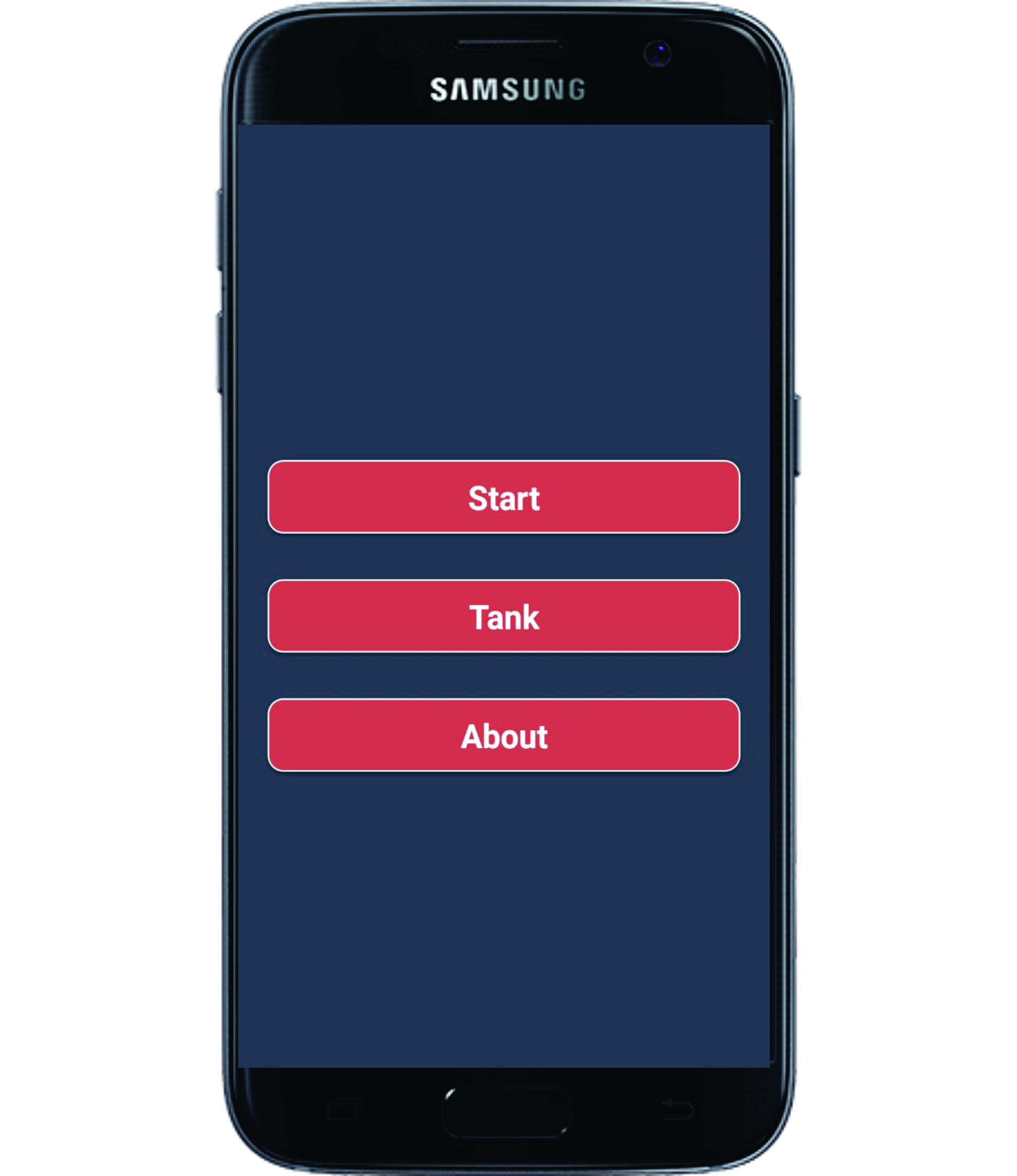
1. Perancangan menu awal.

Sama halnya dengan perancangan halaman sebelumnya yaitu tetap menggunakan xml. Listing tampilan dapat dilihat di Lampiran 2, dan tampilannya dapat dilihat di gambar dibawah ini.



Gambar 3.12 Perancangan Menu

( Sumber : Perancangan )



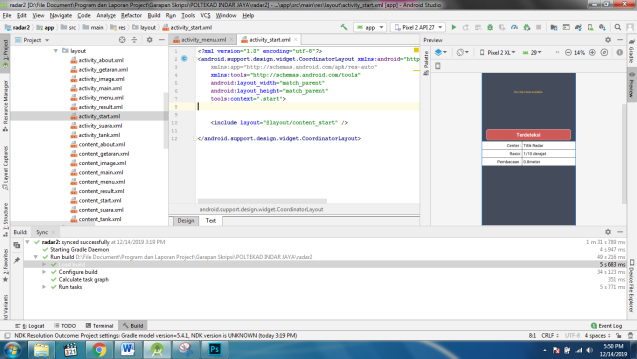
Gambar 3.13 Tampilan Menu

( Sumber : Perancangan )

Menu tersebut berfingsi sebagai berikut :

1. Start berfungsi untuk menjalankan aplikasi pendeteksi tank.
2. Tank berfungsi sebagai tombol untuk menuju halaman spesifikasi tank.
3. About berisi tentang biodata peneliti.
4. Perancangan pendeteksi tank.

Dalam perancangan halaman ini selain perancangan dari tampilan User Interface juga ada perancangan untuk menambahkan fungsi system agar dapat mendeteksi adanya tank Leopard 2 menggunakan Java Class. Berikut Listing dan tampilannya.



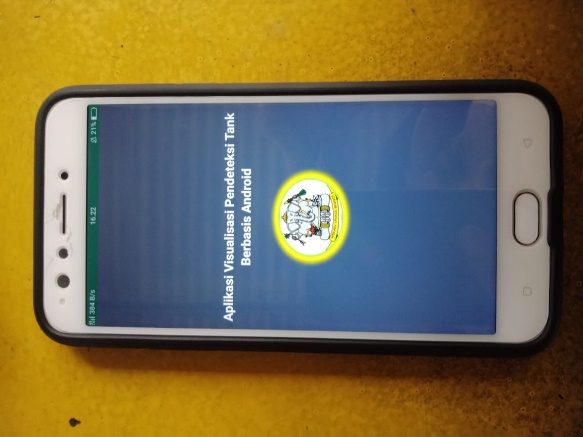
Gambar 3.14 Perancangan Tampilan alat Pendeteksi

( Sumber : Perancangan )

**IV HASIL DAN PEMBAHASAN**

* 1. **Hasil Pengujian Aplikasi.** Pengujian implementasi visualisasi pendeteksi tank diperlukan sedikitnya ada 3 tahap yaitu tahap pengujian penerimaan data dari *firebase*, kemudian pengujian *black box testing* dimana dalam pengujian ini adalah mengecek seluruh menu dan tombol yang ada di aplikasi secara keseluruhan, hasil dapat dinyatakan baik apabila menu dan tombol berfungsi sesuai dengan sekenario yang diharapkan, selanjutnya adalah pengujian langsung aplikasi terhadap objek yaitu tank leopard 2.

1. Hasil penerimaan data dari *firebase***.** Pengecekan *firebase* data yang akan diolah apakah sudah terkoneksi dengan internet, kemudian pastikan smartphone sudah terkoneksi dengan internet guna pengambilan data dari *firebase database.*

**

Terkoneksi

internet

Gambar 4.1 Tampilan Terkoneksi internet

(Sumber : Pengujian)

Gambar diatas menunjukkan bahwa *smartphone* yang akan digunakan untuk menampilkan hasil visualisasi pendeteksi tank sudah terkoneksi dengan internet, ditandai dengan adanya *up down* data yang telah digunakan. Dengan demikian *smartphone* siap untuk menerima masukkan data dari *firebase*

Pengecekan *firebase* dimulai dari, database yang akan diakses sudah terdapat data masukkan dari alat pendeteksi tank, apabila pengiriman data dari alat pendeteksi tank tidak terkendala maka data dapat dilihat pada *firebase*, tampilan dapat dilihat seperti tabel 4.1 dibawah. Apabila data masukkan dari alat pendeteksi sudah ditemukan dalam *firebase* maka secara otomatis data tersebut dapat ditampilkan ke aplikasi visualisasinya

Tabel 4.1 Tampilan *Firebase database*

(Sumber : Pengujian)

|  |  |  |
| --- | --- | --- |
| **No** | **Pengujian** | **Data *Firebase*** |
| **1** | **2** | **3** |
| 1. | Data *image* tampak depan | 10 |
|  |  |  |
|  | Data suara dan  getaran tampak depan | 11 |

1. Hasil pengujian *blackbox* (*blackbox* *testing*). Pengujian *blackbox* adalah salah satu metode pengujian perangkat lunak yang berfokus pada sisi fungsionalitas, khususnya pada *input* dan *output* aplikasi, hal ini berguna untuk memastikan kesesuaian hasil dengan rancangan.

Tabel 4.2 Pengujian *Whitebox*

(Sumber : Pengujian)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Skenario Pengujian | Hasil yang diharapkan | Hasil Uji | Kesimpulan |
| 1 | 2 | 3 | 4 | 5 |
| 1. | Logo awal | Digunakan untuk masuk ke menu utama aplikasi visualisasi | Sesuai | Valid |
| 2. | Menu start | Digunakan untuk masuk ke halaman visualisasi pendeteksi tank | Sesuai | Valid |
| 3. | Menu tank | Digunakan untuk masuk ke halaman data spesifikasi tank | Sesuai | Valid |
| 4. | Menu about | Digunakan untuk masuk ke halaman biodata peneliti | Sesuai | Valid |
| 5. | Tombol terdeteksi | Untuk masuk kedalam hasil pendeteksian objek tank | Sesuai | Valid |
| 6. | *Image* tank | Gambar tangkapan *image* tank, dan menampilkan spesifikasi tank | Sesuai | Valid |
| 7. | Tombol suara | Berfungsi menampilkan grafik suara | Sesuai | Valid |
| 8. | Tombol getaran | Berfungsi menampilkan grafik getaran | Sesuai | Valid |
| 9. | Persentase kecocokan tank | Menampilkan Hasil persentase kecocokan tank | Sesuai | Valid |
| 10. | Jarak Posisi Objek tank | Menampilkan Hasil posisi tank dari alat pendeteksi. | Sesuai | Valid |

1. Pengujian pada tahap ini berfokus pada sisi hasil yang menampilkan visualisasi aplikasi pendeteksi tank.
2. Pengujian pendeteksian objek tank tampak depan, sebagai berikut :

Tabel 4.3 Pengujian Tampak Depan

(Sumber : Pengujian)

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Uraian Pengujian** | **Keterangan** | **Hasil Pengujian** |
| **1** | **2** | **3** | **4** |
| 1. | Posisi objek tank | Informasi ditampilkan adalah sebagai berikut :   * Arah posisi tank berada pada 100 dari posisi alat. * Perkiraan jaraknya adalah 8 m. | 8 radar |
| 2. | Hasil pendeteksian tank | Informasi yang ditampilkan adalah sebagai berikut :   * *Image* objek yang ditangkap oleh alat. * Persentase kecocokan tank sebesar 97%. * Perkiraan jarak objek tank adalah 8 m. | 8 |
| 3. | Spesifikasi tank | Informasi yang ditampilkan adalah sebagai berikut :   * *Image* objek yang ditangkap oleh alat. * Spesifikasi mengenai tank Leopard 2. | Screenshot_2019-11-07-16-11-53-28b |
| 4. | Grafik suara | Informasi yang ditampilkan adalah sebagai berikut :  Besaran frekuensi suara yang tertangkap oleh sensor suara diambil yang paling terakhir masuk yaitu 39 Hz. |  |
| 5. | Grafik getaran | Informasi yang ditampilkan adalah sebagai berikut :  *Besaran* intensitas getaran yang tertangkap oleh *piezo element* di ambil yang paling terakhir masuk yaitu 679 Hz. | 679 |

Hasil dari perhitungan KNN diatas dapat dilihat di tabel 4.4.

Tabel 4.4 Hasil KNN Tampak Depan

(Sumber : Pengujian)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | **Suara (Hz)** | **Getaran (Hz)** | **Euclidean distance (39,679)** | **Rangking** | **Kategori Ya untuk KNN** | **Jarak yang termasuk**  **20-NN** |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| 1 | 36 | 684 | 5.83 | 1 | Ya | 19.39 |
| 2 | 36 | 684 | 5.83 | 2 | Ya | 19.39 |
| 3 | 36 | 684 | 5.83 | 3 | Ya | 19.39 |
| 4 | 36 | 684 | 5.83 | 4 | Ya | 19.39 |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| 5 | 36 | 684 | 5.83 | 5 | Ya | 19.39 |
| 6 | 36 | 684 | 5.83 | 6 | Ya | 19.39 |
| 7 | 36 | 684 | 5.83 | 7 | Ya | 19.39 |
| 8 | 36 | 684 | 5.83 | 8 | Ya | 19.39 |
| 9 | 36 | 684 | 5.83 | 9 | Ya | 19.39 |
| 10 | 36 | 684 | 5.83 | 10 | Ya | 19.39 |
| 11 | 36 | 684 | 5.83 | 11 | Ya | 19.39 |
| 12 | 36 | 684 | 5.83 | 12 | Ya | 19.39 |
| 13 | 36 | 684 | 5.83 | 13 | Ya | 19.39 |
| 14 | 36 | 684 | 5.83 | 14 | Ya | 19.39 |
| 15 | 36 | 684 | 5.83 | 15 | Ya | 19.39 |
| 16 | 36 | 684 | 5.83 | 16 | Ya | 19.39 |
| 17 | 36 | 684 | 5.83 | 17 | Ya | 19.39 |
| 18 | 36 | 684 | 5.83 | 18 | Ya | 19.39 |
| 19 | 36 | 684 | 5.83 | 19 | Ya | 19.39 |
| 20 | 36 | 684 | 5.83 | 20 | Ya | 19.39 |
| 21 | 36 | 684 | 5.83 | 21 | Tidak | - |
| 22 | 36 | 684 | 5.83 | 22 | Tidak | - |
| 23 | 36 | 684 | 5.83 | 23 | Tidak | - |
| 24 | 36 | 684 | 5.83 | 24 | Tidak | - |
| 25 | 36 | 684 | 5.83 | 25 | Tidak | - |
| 26 | 36 | 684 | 5.83 | 26 | Tidak | - |
| 27 | 36 | 684 | 5.83 | 27 | Tidak | - |
| 28 | 36 | 684 | 5.83 | 28 | Tidak | - |
| 29 | 36 | 684 | 5.83 | 29 | Tidak | - |
| 30 | 36 | 684 | 5.83 | 30 | Tidak | - |
| 31 | 36 | 684 | 5.83 | 31 | Tidak | - |
| 32 | 36 | 684 | 5.83 | 32 | Tidak | - |
| 33 | 36 | 684 | 5.83 | 33 | Tidak | - |
| 34 | 36 | 684 | 5.83 | 34 | Tidak | - |
| 35 | 36 | 684 | 5.83 | 35 | Tidak | - |
| 36 | 36 | 684 | 5.83 | 36 | Tidak | - |
| 37 | 36 | 684 | 5.83 | 37 | Tidak | - |
| 38 | 36 | 684 | 5.83 | 38 | Tidak | - |
| 39 | 36 | 684 | 5.83 | 39 | Tidak | - |
| 40 | 36 | 684 | 5.83 | 40 | Tidak | - |
| 41 | 36 | 684 | 5.83 | 41 | Tidak | - |
| 42 | 36 | 684 | 5.83 | 42 | Tidak | - |
| 43 | 36 | 684 | 5.83 | 43 | Tidak | - |
| 44 | 36 | 684 | 5.83 | 44 | Tidak | - |
| 45 | 36 | 684 | 5.83 | 45 | Tidak | - |
| 46 | 36 | 684 | 5.83 | 46 | Tidak | - |
| 47 | 36 | 684 | 5.83 | 47 | Tidak | - |
| 48 | 36 | 684 | 5.83 | 48 | Tidak | - |
| 49 | 36 | 684 | 5.83 | 49 | Tidak | - |
| 50 | 36 | 684 | 5.83 | 50 | Tidak | - |

Berdasarkan hasil perhitungan KNN maka nilai jaraknya adalah 8 meter.

* 1. **Pembahasan.** Pada poin ini berisi tentang pembahasan dan analisa dari hasil pengujian aplikasi visualisasi pendeteksi tank.

1. **Pembahasan penerimaan data dari firebase.**

Hasil penyimpanan data dari *firebase database* dapat dilihat pada tabel 4.1, data tersebut yang akan ditampilkan pada aplikasi visualisasinya, berdasarkan data yang diperoleh maka dapat disimpulkan bahwa data masukan dari alat pendeteksi dapat diterima dengan baik oleh *smartphone*, karena itu dapat dinyatakan bahwa hasil penerimaan data dari *firebase* berjalan dengan baik dikarenakan data *image*, suara, dan getaran yang dikirim oleh alat pendeteksi sama dengan data yang ditampilkan oleh aplikasi visualisasi pendeteksi tank Leopard 2.

1. **Pembahasan Blackbox testing.**

Hasil Blackbox testing dinyatakan berhasil dengan baik apabila seluruh menu dan tombol dapat bekerja dengan baik tanpa ada kendala apapun, baik dari segi fungsionalitas perangkat keras dan perangkat lunak, ataupun error yang ditimbulkan karena kesalahan perancangan. Dapat dilihat dari hasil *blackbox testing* seperti yang dapat dilihat pada tabel 4.2 dimana dari 10 sekenario pengujian menu dan tombol yang ada pada aplikasi seusai atau

berjalan dengan baik menurut fungsionalitas yang diharapkan. Maka dapat dinyatakan bahwa tingkat kesesuaian fungsionalitas menu dan tombol pada aplikasi dapat berjalan dengan baik hingga 100% tanpa error.

1. **Pembahasan hasil visualisasi pendeteksi tank Leopard 2.**
2. Pembahasan hasil pengujian tank tampak depan.

Tabel 4.19 Nilai Visualisasi Tampak Depan

(Sumber : Pengujian)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Percobaan Pengujian** | **Hasil Visualisasi** | | **Hasil KNN** | |
| **1** | **2** | | **3** | |
| Objek Tampak Depan | Gambar | Sesuai dengan data *firebase* | Persentase | 97% |
| Spesifikasi | Sesuai dengan data *firebase* | Jarak | 8 m |
| Grafik Suara | 39 Hz |  |  |
| Grafik Getaran | 679 Hz |  |  |

Dari hasil pengujian tank tampak depan didapatkan data seperti tabel diatas. Untuk penerimaan data dari firebase sesuai dengan data yang telah diterima dari alat pendeteksi, maka pengujian kali ini dapat dinyatakan berjalan dengan baik berdasarkan perbandingan antara hasil yang ditunjukkan oleh visualisasi pendeteksi tank dan menurut perhitungan yang menggunakan sampel data yang diambil dari 50 record teratas dari 1047 data training didapatkan hasil jumlah data training jarak 19.39 = 20 kali muncul, sehingga ditetapkan bahwa perkiraan jaraknya adalah 19 meter, dari keseluruhan menu fitur yang ada dalam aplikasi berjalan dengan baik maka dapat dinyatakan pengujian tampak depan berhasil 100%.

Dari hasil pengujian tank tampak samping kiri didapatkan data seperti tabel diatas. Untuk penerimaan data dari firebase sesuai dengan data yang telah diterima dari alat pendeteksi, maka pengujian kali ini dapat dinyatakan berjalan dengan baik berdasarkan perbandingan antara hasil yang ditunjukkan oleh visualisasi pendeteksi tank dan menurut perhitungan yang menggunakan sampel data yang diambil dari 50 record teratas dari 1047 data training didapatkan hasil jumlah data training jarak 4.12 = 20 kali muncul, sehingga ditetapkan bahwa perkiraan jaraknya adalah 4 meter, dari keseluruhan menu fitur yang ada dalam aplikasi berjalan dengan baik maka dapat dinyatakan pengujian tampak depan berhasil 100%

Adapun hasil dari pengujian aplikasi visualisasi pendeteksi tank yang dimulai dari tahap persiapan yaitu penginstalan apk visulaisasi pendeteksi tank dapat berjalan dengan baik. Dilanjutkan dengan pengujian seluruh tombol yang ada di aplikasi untuk setiap halaman dalam tabel *black box testing* berjalan sesuai yang diharapkan, dilanjutkan dengan pengujian aplikasi terhadap objek tank menghasilkan data dengan tingkat akurasi kecocokan tank yang cukup memuaskan antara 50%-98% maka dapat dinyatakan pengujian aplikasi visualisasi pendeteksi tank Leopard 2 berjalan dengan baik.

**V. KESIMPULAN DAN SARAN**

* 1. **Kesimpulan.**  Dari analisa secara teoritis, analisa perhitungan dan hasil pengujian alat yang telah dilakukan, maka pada akhir penulisan Tugas Akhir yang berjudul Implementasi Visualisasi Pendeteksi Tank Berbasis Android ini dapat diambil kesimpulan sebagai berikut :

1. Hasil dari penelitian ini menunjukkan bahwa Implementasi Visualisasi alat pendeteksi tank sangat terpengaruh pada 3 variabel yaitu *image*, suara, dan getaran. Dimana untuk variabel *image* diperoleh hasil tertinggi tingkat kecocokan tank adalah 97%, Untuk variabel suara diperoleh frekuensi terendah yaitu 35 Hz dan tertinggi adalah 39 Hz, sedangkan untuk variabel getaran diperoleh frekuensi terendah yaitu 37 Hz, dan frekuensi getaran tertinggi mencapai 1.023 Hz.
2. Penggunaan metode KNN yang digunakan untuk mendeteksi jenis tank sudah dapat berjalan dengan baik, dimana untuk frekuensi suara yang ditangkap sebesar 39 Hz, sedangkan untuk vfrekuensi getaran sebesar 679 Hz dengan tingkat kecocokan 97%.
   1. **Saran.** Untuk lebih menyempurnakan Implementasi Visualisasi Pendeteksi Tank Berbasis Android, maka peneliti menyarankan, sebagai berikut :
3. Perlu pengembangan aplikasi visualisasi pada system operasi mobile yang lainnya seperti Apple IOS dan lain sebagainya.
4. Perlu penambahan data tank jenis lainnya.
5. Perlu adanya penambahan variabel lain seperti variabel temperatur mesin, dan arus tegangan tank guna menyempurnakan data visualisasi yang akan diterima.

**VI. PENUTUP**

* 1. **Penutup.** Demikianlah penulisan Tugas Ahir dengan judul“**IMPLEMENTASI VISUALISASI PENDETEKSI TANK BERBASIS ANDROID”**.ini dibuat, semoga bermanfaat bagi jajaran TNI AD untuk meningkatkan profesionalisme dalam bidang teknologi. Penulis menyadari bahwa tulisan ini masih banyak kekurangan dan jauh dari sempurna, oleh karena itu penulis mengharapkan saran dan kritik yang membangun untuk menambah pengetahuan dan wawasan agar lebih baik dimasa depan. Penulis mengucapkan terimakasih yang setinggi-tingginya kepada semua pihak yang telah membantu kelancaran dalam penulisan Tugas Ahir.

DAFTAR PUSTAKA

[1] Anthony, Howard Mark.2012. “Close Combat Vehicle And Leopard 2 Main Battle Tank : Back In The Heavyweight Fight.”

[2] Hermawan S, Stephanus. 2011. “Mudah Membuat Aplikasi Android”. Andi Offset. Yogyakarta.

[3] Indrajani. 2011. “Perancangan Basis Data dalam All in 1”, PT. Elex Media Komputindo, Jakarta.

[4] Kusrini, Emha T. Luthfi, 2009, Algoritma Data Mining. Andi, Yogyakarta.

[5] Nazarudin Safaat Harahap. 2012. “Pemograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android”. Informatika. Bandung.

[6] Saxena, Krati. 2014. Diagnosis of Diabetes Mellitus using K Nearest Neighbor Algorithm. Internasional Journal of Computer Science Trends and Technology (IJCST).

[7] Sukma, Alfian et-al . 2014. K-Nearest Neighbor (Information Retrieval). Fakultas Sains dan Teknologi Universitas Airlangga.

[8] Widiarsana, O., Putra, N.W., Budiyasa, P.G.I., Bismantara, A.N.I., Mahajaya, S.N. 2011. Data Mining: Metode Clasification K-Nearest Neighbor (KNN). Bali: Program Studi Teknologi Informasi Universitas Udayana.

[9] Zainal Arifin. 2012. “Hacking & Programming dengan Android SDK untuk Advanced”. Elex Media. Jakarta.

[10] *Sukma, Alfian et-al . 2015. K-Nearest Neighbor (Information Retrieval). Fakultas Sains dan Teknologi Universitas Airlangga.*

[11] *Widiarsana, O., Putra, N.W., Budiyasa, P.G.I., Bismantara, A.N.I., Mahajaya, S.N. 2011. Data Mining: Metode Clasification K-Nearest Neighbor (KNN). Bali: Program Studi Teknologi Informasi Universitas Udayana.*

[12] *K G Sudiartha, I N E Indrayana, I W Suasnawa, Jurnal Ilmu Komputer VOL. XI No. 2 p-ISSN: 1979-5661 e-ISSN: 2622-321X, "Membangun Struktur Realtime Database Firebase Untuk Aplikasi Monitoring Pergerakan Group Wisatawan".*

[13] *Andi Juansyah, Jurnal Ilmiah Komputer dan Informatika (KOMPUTA) Edisi. 1 Volume. 1 Agustus 2015 ISSN : 2089-9033 “Pembangunan Aplikasi Child Tracker Berbasis Assisted – Global Positioning System (A-GPS) Dengan Platform Android”*

1. Anthony, Howard Mark, “Close Combat Vehicle And Leopard 2 Main Battle Tank : Back In The Heavyweight Fight.”, 2012 [↑](#footnote-ref-1)