

BAB II

TINJAUAN PUSTAKA

2.1. Panti Asuhan

Dalam suatu panti asuhan berada di bawah pengawasan dinas sosial. Untuk itu, sesuai dengan Amanat UUD 1945 pasal 34 ayat, fakir miskin dan anak terlantar dijamin oleh Negara. Tanggung jawab sosial membuat masyarakat turut berpartisipasi untuk mengemban tanggung jawab tersebut salah satunya melalui pembangunan dan pengelolaan panti asuhan sebagai keluarga alternatif bagi anak fakir dan terlantar.

Panti asuhan merupakan suatu lembaga kesejahteraan sosial anak yang memberikan pelayanan pada anak terlantar dengan melaksanakan penyantunan, sebagai pengganti orang tua/wali anak dalam memenuhi kebutuhan fisik, mental dan sosial kepada anak asuh untuk membentuk perkembangan anak-anak yang tidak memiliki keluarga ataupun yang tidak tinggal bersama dengan keluarga (Departemen Sosial RI., 2004).

Berdasarkan pengertian diatas, dapat disimpulkan bahwa panti asuhan merupakan salah satu lembaga perlindungan anak yang berfungsi memberikan perlindungan terhadap hak anak – anak sebagai wakil orang tua dalam memenuhi kebutuhan mental dan sosial pada anak asuh agar mereka memiliki kesempatan untuk mengembangkan diri sampai mencapai tingkat kedewasaan yang matang serta mampu melaksanakan peranya sebagai individu dan warga negara di dalam kehidupan masyarakat.

2.2. Android

Android merupakan *Operating System(OS) Mobile* yang berkembang dewasa ini.OS lainnya seperti *Windows Mobile, Symbian OS, iOS*, dan masih banyak lagi juga menawarkan keoptimalan berjalan di atas *hardware* yang ada. Akan tetapi OS yang ini berjalan dengan memprioritaskan aplikasi inti yang dibangun sendiri

tanpa melihat potensi yang cukup besar dari aplikasi pihak ketiga (Kurniawan, 2017). Berikut fitur-fitur yang ada pada Android adalah :

1. *Framework* Aplikasi : Memungkinkan penggunaan dan pemindahan dari komponen yang tersedia.
2. Dalvik *Virtual Machine* : *Virtual Machine* untuk pengoptimalan perangkat *mobile*.
3. Grafik : Grafik 2D dan 3D yang menggunakan *library OpenGL*.
4. *SQLite* : Untuk penyimpanan data.
5. Mendukung Media : Audio, video, dan berbagai format gambar (MPEG, MP3, H.264, AAC, AMR, JPG, GIF, PNG).
6. GSM, Bluetooth, EDGE, 3G dan Wi-Fi(tergantung *hardware*).
7. Kamera, *Global Positioning System* (GPS), kompas dan *accelerometer* (tergantung *hardware*).Lingkungan pengembangan yang kaya seperti emulator, *debugging*, dan *plugin Eclipse IDE*.

2.3. Basis Data

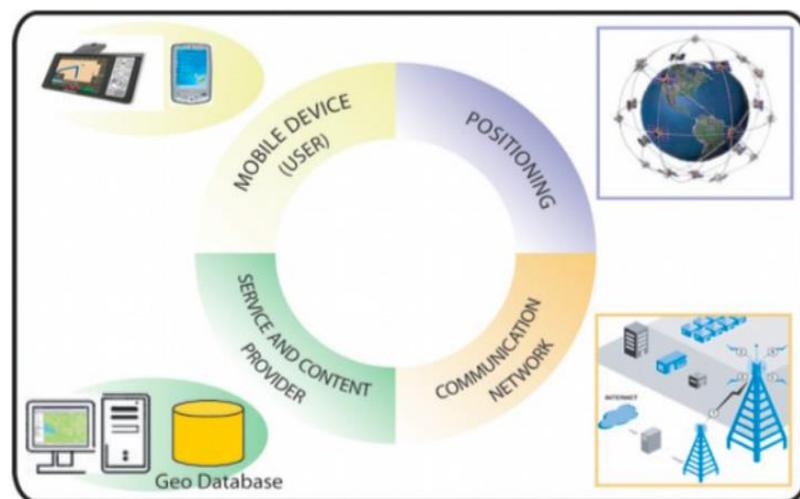
Basis data atau database merupakan kumpulan data satu dengan data lainnya yang tersimpan dalam satu tempat penyimpanan luar dan membutuhkan suatu perangkat lunak untuk menjalankannya (Edhy, 2012). Elemen-elemen basis data antara lain :

1. Entitas kumpulan dari beberapa objek yang memiliki karakter sama namun bisa di bedakan satu dengan lain nya. Contoh objek nya berupa orang, barang atau tempat.
2. Atribut unit terkecil dalam data yang mempunyai relasi dan makna bagi pengguna atau *user*.
3. Data *value* data atau informasi yang tersimpan pada tipe data, atribut, dan elemen. Contoh nilai data yaitu atribut nama siswa.
4. *Field*/tabel kumpulan karakter yang di bentuk dalam satu arti. Jika field itu terdapat seperti nama barang atau nomor barang, maka *field* yang terdapat harus berhubungan dengan atau berkaitan dengan nama dan nomor barang tersebut.
5. *Record* ialah kumpulan dari beberapa *field* yang lengkap, dan di hitung dalam bentuk satuan baris.

2.4. Location Based Service (LBS)

Location based service adalah layanan informasi yang di akses menggunakan piranti *mobile* melalui jaringan internet dan seluler serta memanfaatkan kemampuan penunjuk lokasi pada piranti *mobile*. Konsep dari metode *Location Based Service* ini sendiri menggunakan database informasi geografis yang digabungkan dengan teknologi *Global Positioning System* (GPS) yang tertanam di *smartphone* pengguna untuk melacak suatu pergerakan *device* pengguna dan mengirimkan informasi yang dibutuhkan oleh *device* pengguna. (Susanty, Astari, & Thamrin, 2019).

1. *Mobile Device* yaitu sebuah alat yang digunakan oleh pengguna untuk meminta informasi yang dibutuhkan. Perangkat memungkinkan yaitu perangkat yang memiliki fasilitas navigasi seperti PDA, *mobile phone*, laptop dan lainnya.
2. *Communication Network* adalah jaringan selular yang mengirimkan data pengguna dan permintaan layanan.
3. *Positioning Component* biasanya posisi pengguna harus ditentukan untuk pengolahan layanan. Posisi pengguna dapat diperoleh menggunakan jaringan komunikasi atau dengan menggunakan *Global Positioning System* (GPS).
4. *Service and Content Provider* yaitu penyedia layanan informasi data yang dapat di minta oleh pengguna. Komponen LBS dapat ditunjukkan pada gambar berikut:



Gambar 2. 1 Komponen *Location Based Service* (LBS)

(Sumber : Susanty, 2019)

a. Unsur Utama pada *Location Based Service* (LBS)

Location Based Service (LBS) memiliki unsur utama yaitu :

1. *Location* (*API Maps*) menyediakan perangkat bagi sumber atau *source* untuk *location based service* (LBS), *Application Programming Interface* (*API maps*) menyediakan fasilitas untuk menampilkan dan memanipulasi peta.
2. *Location Provider* (*API Location*) menyediakan teknologi pencarian lokasi yang digunakan oleh perangkat. *API Location* berhubungan dengan data GPS (*Global Positioning System*) dan data lokasi *real-time*. *API Location* berada pada data android yaitu data paket internet yang digunakan oleh perangkat.

b. Cara akses layanan *Location Based Service* (LBS)

Pada *platform* ada dua cara yang berbeda untuk mengakses layanan LBS:

1. Inisiatif dari *platform*:

Pengguna mengirimkan permintaan (teks) untuk informasi tentang layanan di daerah dekat sekitarnya.

2. Inisiatif dari pengguna

Pengguna register terlebih dahulu untuk menerima tertentu informasi setiap kali dekat dengan tempat pengguna. Pengguna menerima diminta informasi pada item baru apabila di dekat tempat sekitar tersebut.

2.5. Global Positioning System (GPS)

Global Positioning System (GPS) merupakan suatu kumpulan satelit dan sistem kontrol yang memungkinkan sebuah penerima GPS untuk mendapatkan lokasinya di permukaan bumi 24 jam sehari. Sistem ini menggunakan sejumlah satelit yang berada di orbit bumi, yang memancarkan sinyal ke bumi dan di tangkap oleh sebuah alat penerima. *Global Positioning System* (GPS) adalah sistem untuk menentukan posisi di permukaan bumi dengan bantuan sinkronisasi sinyal satelit. Sistem ini menggunakan minimal 4 satelit yang mengirimkan gelombang mikro ke bumi. Sinyal ini di terima oleh alat penerima di permukaan dan di gunakan untuk menentukan posisi, kecepatan, arah dan waktu (Susanty, Astari, & Thamrin, 2019).

2.6. Teori *Haversine*

Rumus *Haversine* digunakan untuk menghitung jarak antara dua titik di permukaan bumi dengan menggunakan lintang dan bujur sebagai variabel masukan. Rumus *Haversine* adalah persamaan penting dalam navigasi, memberikan jarak melingkar besar antara dua titik pada permukaan bola (bumi) dengan bujur dan lintang. Penggunaan rumus ini mengasumsikan mengabaikan efek *elips*, cukup akurat untuk sebagian besar perhitungan, juga mengabaikan ketinggian perbukitan rendah dan lembah di permukaan bumi (Chopde, 2013).

Perumusan :

$$\begin{aligned}\Delta\text{lat} &= \text{Lat2} - \text{lat1} \\ \Delta\text{long} &= \text{Long2} - \text{long1} \\ a &= \sin^2(\Delta\text{lat} / 2) + \cos(\text{lat1}) \cos(\text{lat2}) \sin^2(\Delta\text{long} / 2) \\ c &= 2 \operatorname{atan2}(\sqrt{a}, \sqrt{1-a}) \\ D &= R c\end{aligned}$$

Informasi :

$$\begin{aligned}R &= \text{Bumi radius yaitu. } 6371000 \text{ (m)} \\ \Delta\text{lat} &= \text{Jumlah perubahan lintang} \\ \Delta\text{long} &= \text{Jumlah perubahan bujur} \\ a &= \text{Sumbu berpotongan perhitungan} \\ c &= \text{Perhitungan salib sumbu} \\ D &= \text{Jarak (meteran)} \\ 1 \text{ derajat} &= 0,0174533 \text{ radian}\end{aligned}$$

2.7. Perangkat Lunak Pendukung

Berikut adalah perangkat lunak pendukung dalam penunjang pembangunan aplikasi yang akan di bangun.

2.7.1 *Android Studio*

Android studio adalah IDE (*Integrated Development Environment*) resmi untuk pengembangan aplikasi *Android* dan bersifat *open source* atau gratis. Peluncuran *Android Studio* ini diumumkan oleh *Google* pada 16 mei 2013 pada *event Google I/O Conference* untuk tahun 2013. Sejak saat itu, *Android Studio* menggantikan

Eclipse sebagai IDE resmi untuk mengembangkan aplikasi *Android* (Juansyah, 2015).

Android studio sendiri dikembangkan berdasarkan *IntelliJ IDEA* yang mirip dengan *Eclipse* disertai dengan *ADT plugin (Android Development Tools)*.

Android studio memiliki fitur :

- a. Projek berbasis pada *Gradle Build*
- b. *Refactory* dan pembenahan bug yang cepat
- c. *Tools* baru yang bernama “*Lint*” dikalim dapat memonitor kecepatan, kegunaan, serta kompetibelitas aplikasi dengan cepat.
- d. Mendukung *Proguard And App-signing* untuk keamanan.
- e. Memiliki GUI aplikasi android lebih mudah
- f. Didukung oleh *Google Cloud Platfrom* setiap aplikasi yang dikembangkan.

2.7.2 Java Development Kit (JDK)

Java Development Kit (JDK) adalah sekumpulan perangkat lunak yang dapat kamu gunakan untuk mengembangkan perangkat lunak yang berbasis Java, sedangkan *JRE* adalah sebuah implementasi dari *Java Virtual Machine* yang benar-benar digunakan untuk menjalankan program java. Baisanya, setiap *JDK* berisi satu atau lebih *JRE* dan berbagai alat pengembangan lain seperti sumber *compiler java, bundling, debuggers, development libraries* dan lain sebagainya (Juansyah, 2015).

2.7.3 Google Maps API

Google Maps API adalah layanan berbasis web yang menyediakan informasi rinci suatu wilayah geografis dan situs di seluruh dunia. Selain peta jalan konvensional, *Google Maps* memperlihatkan di udara dengan satelit dari banyak tempat. Di beberapa kota, *Google Maps* menawarkan pemandangan jalan yang terdiri foto yang diambil dari kendaraan (Ariyanti, Khairil, & Kanedi, 2015).

2.8. Metode Pengumpulan Data

Teknik pengumpulan data merupakan faktor penting demi keberhasilan penelitian. Hal ini berkaitan dengan bagaimana cara mengumpulkan data, siapa sumbernya, dan apa alat yang digunakan (Hendryadi, 2014).

1. Observasi

Observasi merupakan salah satu teknik pengumpulan data yang tidak hanya mengukur sikap dari responden (wawancara) namun juga dapat digunakan untuk merekam berbagai fenomena yang terjadi (situasi, kondisi). Teknik ini digunakan bila penelitian ditujukan untuk mempelajari perilaku manusia, proses kerja, gejala-gejala alam dan dilakukan pada responden yang tidak terlalu besar.

2. Wawancara

Wawancara merupakan teknik pengumpulan data yang dilakukan melalui tatap muka dan tanya jawab langsung antara pengumpul data maupun peneliti terhadap nara sumber atau sumber data.

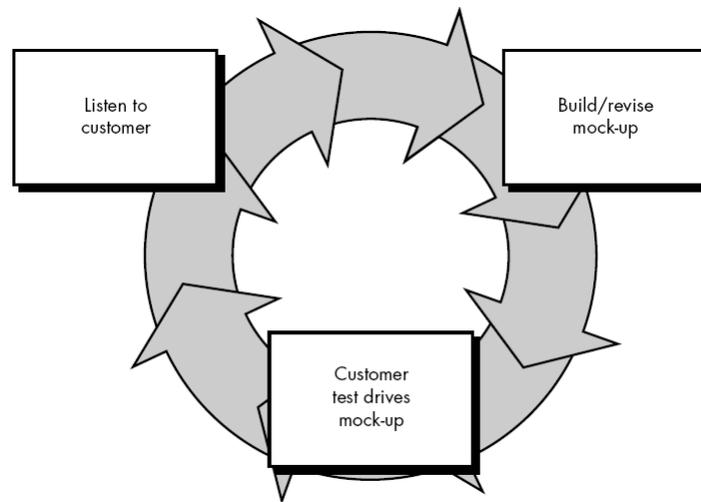
3. Kuesioner

Kuesioner adalah daftar pertanyaan tertulis yang telah disusun sebelumnya. Pertanyaan-pertanyaan yang terdapat dalam kuesioner, atau daftar pertanyaan tersebut cukup terperinci lengkap dan biasanya sudah menyediakan pilihan jawaban (kuesioner tertutup) atau memberikan kesempatan responden menjawab secara bebas (kuesioner terbuka).

2.9. Metode Pengembangan Sistem

Prototype merupakan metodologi pengembangan *software* yang menitik pada pendekatan aspek desain, fungsi dan *user-interface*. *Developer* dan *user* fokus pada *user-interface* dan bersama-sama mendefinisikan spesifikasi, fungsi, desain dan bagaimana *software* bekerja. *Developer* dan *user* bertemu melakukan komunikasi dan menentukan tujuan umum, kebutuhan yang diketahui dan gambaran bagian-bagian yang akan dibutuhkan. *Developer* mengumpulkan detail dari kebutuhan dan memberikan suatu gambaran dengan cetak biru (*prototype*). Dari proses tersebut akan diketahui detail-detail yang harus dikembangkan atau ditambahkan oleh *developer* terhadap cetak biru, atau menghapus detail-detail

yang tidak diperlukan oleh *user*. Proses akan terjadi terus menerus sehingga produk sesuai dengan keinginan dari *user*.



Gambar 2. 2 Metode Pengembangan Sistem *Prototype*

(Sumber : Rosa, 2015)

Pengembangan dari perancangan sistem ini dalam pelaksanaannya menggunakan tiga tahap siklus pengembangan model *prototype* yaitu:

1. Mendengarkan Pelanggan (*Listen to Customer*) merupakan tahap pertama dalam merancang sebuah sistem. Pada tahap ini akan menentukan informasi-informasi yang dibutuhkan oleh pelanggan agar tercipta sebuah aplikasi sehingga mengarah pada tujuan dibuatnya aplikasi tersebut.
2. Membangun dan Memperbaiki prototipe (*Build/revise mockup*) dalam tahap ini dilakukan perancangan dan pengkodean untuk sistem yang diusulkan yang mana tahapannya meliputi perancangan proses-proses yang akan terjadi dalam sistem, perancangan diagram UML yang akan digunakan, perancangan antarmuka keluaran serta dilakukan tahap pengkodean terhadap rancangan-rancangan yang telah didefinisikan, kelengkapan *software* dan *hardware*.
3. Pengujian prototipe pada tahapan ini dilakukan pengujian terhadap sistem yang telah disusun dan melakukan pengenalan terhadap sistem yang telah diujikan serta evaluasi apakah sistem yang sudah jadi sudah sesuai dengan yang diharapkan.

2.10 Teori *Blackbox Testing*

Black box testing adalah *tipe testing* yang memerlukan perangkat lunak yang tidak diketahui kinerja internalnya. Sehingga para tester memandang perangkat lunak seperti layaknya sebuah “kotak hitam” yang tidak penting dilihat isinya, tapi dikenal proses testing dibagian luar. Metode ujicoba *blackbox* memfokuskan pada keperluan fungsional dari software. Karena itu ujicoba *blackbox* memungkinkan pengembang software untuk membuat himpunan kondisi input yang akan melatih seluruh syarat-syarat fungsional suatu program (Sari, 2016).

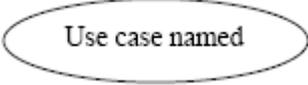
2.11 UML (*Unified Modeling Language*)

Unified Modeling Language (UML) adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. UML merupakan metodologi dalam mengembangkan sistem berorientasi objek dan juga merupakan alat untuk mendukung pengembangan sistem. *Unified Modeling Language* (UML) adalah sebuah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan software berbasis OO (*Object-Oriented*). UML sendiri juga memberikan standar penulisan sebuah sistem *blue print*, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen - komponen yang diperlukan dalam sistem *software*. Diagram *Unified Modelling Language* (UML) (Suendri, 2018) antara lain sebagai berikut:

1. *Use Case Diagram*, *Use case* menggambarkan *external view* dari sistem yang akan kita buat modelnya (Rosa, 2015) Model *use case* dapat dijabarkan dalam diagram *use case*, tetapi perlu diingat, diagram tidak indetik dengan model karena model lebih luas dari diagram. *Use case* harus mampu menggambarkan urutan aktor yang menghasilkan nilai terukur (Rosa, 2015).

Tabel 2. 1 Simbol-Simbol *UseCase*.

(Sumber : Rosa, 2015)

SIMBOL	NAMA	KETERANGAN
	Actor	Actor adalah pengguna sistem. Actor tidak terbatas hanya manusia saja, jika sebuah sistem berkomunikasi dengan aplikasi lain dan membutuhkan input atau memberikan <i>output</i> , maka aplikasi tersebut juga bisa dianggap sebagai actor.
	Use Case	<i>Use case</i> digambarkan sebagai lingkaran elips dengan nama use case dituliskan didalam elips tersebut.
	Association	Asosiasi digunakan untuk menghubungkan actor dengan <i>use case</i> . Asosiasi digambarkan dengan sebuah garis yang menghubungkan antara Actor dengan <i>Use Case</i> .

2. *Class Diagram*, Kelas sebagai suatu set objek yang memiliki atribut dan perilaku yang sama, kelas kadang disebut kelas objek (Rosa, 2015). *Class* memiliki tiga area pokok yaitu :
1. Nama, kelas harus mempunyai sebuah nama.
 2. Atribut, adalah kelengkapan yang melekat pada kelas. Nilai dari suatu kelas hanya bisa diproses sebatas atribut yang dimiliki.
 3. Operasi, adalah proses yang dapat dilakukan oleh sebuah kelas, baik pada kelas itu sendiri ataupun kepada kelas lainnya.

Tabel 2. 2 Simbol *Class Diagram*

(Sumber : Rosa, 2015)

GAMBAR	NAMA	KETERANGAN
	Generalization	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	N-Ary Association	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
	Class	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
	Collaboration	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor.
	Realization	Operasi yang benar-benar dilakukan oleh suatu objek.
	Dependency	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
	Association	Apa yang menghubungkan antara objek satu dengan objek lainnya.

3. *Activity Diagram*, Diagram aktifitas menunjukkan aktifitas sistem dalam bentuk kumpulan aksi-aksi, bagaimana masing-masing aksi tersebut dimulai, keputusan yang mungkin terjadi hingga berakhirnya aksi. *Activity diagram* juga dapat menggambarkan proses lebih dari satu aksi salam waktu bersamaan. “Diagram *activity* adalah aktifitas-aktifitas, objek, *state*, transisi *state* dan *event*. Dengan kata lain kegiatan diagram alur kerja menggambarkan perilaku sistem untuk aktifitas” (Rosa, 2015).

Tabel 2. 3 Simbol Activity Diagram

(Sumber : Rosa, 2015)

SIMBOL	KETERANGAN
	Titik Awal
	Titik Akhir
	Activity
	Pilihan Untuk mengambil Keputusan
	<i>Fork</i> ; Digunakan untuk menunjukkan kegiatan yang dilakukan secara parallel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	<i>Rake</i> ; Menunjukkan adanya dekomposisi
	Tanda Waktu
	Tanda pengiriman
	Tanda penerimaan
	Aliran akhir (<i>Flow Final</i>)

2.12. Penelitian Terkait

Pada (Irawati, Aswin, & Salim, 2018) menghasilkan penelitian tentang menampilkan posisi secara geografis dalam pencarian lokasi panti asuhan di Kota Ambon. Dengan sistem ini dapat menyelesaikan permasalahan masyarakat yang kurang mengetahui panti asuhan tersebut. Tetapi pada sistem ini pengguna hanya dapat menggunakan pencarian lokasi saja. Sedangkan pengguna LBS (*location based service*) pada pencarian rute menampilkan informasi panti asuhan terkait untuk memudahkan donatur dalam menyimpulkan panti asuhan yang lebih membutuhkan donasi. Selanjutnya, pada Setiani, & Rahmalia, (2019) memanfaatkan sistem pemesanan salon online untuk menentukan informasi lokasi tempat pelanggan ke pemesan salon saja. Selanjutnya, pada Ika, dan Dwi, (2018) menghasilkan penelitian rancang bangun aplikasi Location Based Service untuk pencarian lokasi amal usaha Muhammadiyah. Dimana proses perhitungan jarak terpendek antar lokasinya menggunakan metode *Haversine*, hal ini dikarenakan titik-titik koordinat lokasi berada pada permukaan bumi yang merupakan bidang lengkung. Kemudian penentuan rute terpendek menggunakan algoritma. Selanjutnya, pada Essayad, (2011) menghasilkan LBS (*location based service*)

memungkinkan pengguna di perjalanan dengan akses untuk geografis data dari mikro-laptop, tablet PC, personal *digital assistant* (PDA). Layanan ini mengacu pada teknologi GPRS2 UMTS3 dan dapat menggunakan data lokasi spasial perangkat GPS di peta. Pada penelitian kali ini pengembangan menggunakan LBS (*location based service*) yang diharapkan dapat menentukan lokasi panti asuhan di Kota Bandar Lampung, dapat menghitung jumlah data panti asuhan, anak asuh, pengurus panti, jumlah laki-laki dan perempuan serta menampilkan informasi panti asuhan secara detail dengan menggunakan aplikasi android.