

Aplikasi Pendaftaran dan Transaksi di Klinik Hewan di Bandung Berbasis Android

Tiara Sabila¹, Ely Rosely², Heru Nugroho³

^{1,2,3}Program Studi D3 Manajemen Informatika, Fakultas Ilmu Terapan, Universitas Telkom

¹arasabila@student.telkomuniversity.ac.id, ²ely.rosely@tass.telkomuniversity.ac.id, ³heru@tass.telkomuniversity.ac.id

Aplikasi Pendaftaran dan Transaksi di Klinik Hewan di Bandung Berbasis Android merupakan aplikasi yang digunakan oleh pemilik pasien untuk melihat data pemilik pasien secara lengkap melalui *smartphone*. Aplikasi ini digunakan oleh pemilik pasien dengan fitur-fitur tertentu. Pemilik pasien dapat mengakses beberapa fitur seperti melihat data pemilik pasien, melihat data pasien, melihat data rekam medis pasien, melihat riwayat transaksi, melakukan reservasi jadwal pada klinik tertentu, melakukan pendaftaran pada lebih dari satu klinik, dan melihat lokasi klinik hewan terdekat. Aplikasi dibangun dengan berbasis android yang dibangun dengan metode waterfall, menggunakan bahasa pemrograman Java dan PHP, dan menggunakan tools Android Studio. Dengan dibangunnya aplikasi ini dapat membantu pemilik pasien mengetahui data pasien secara lengkap, mereservasi jadwal, melihat lokasi klinik terdekat, dan membantu proses pendaftaran pada lebih dari satu klinik melalui *smartphone*.

Kata Kunci: Waterfall, Android, Klinik Hewan, Java, PHP, Smartphone

Android Based Application of Registration and Patient Transaction is an application used by the patient owner to view the patient's data completely via smartphone. This app is used by patient owners and animal clinics with certain features. The patient owner can access several features such as viewing patient's owner data, view patient data, view patient medical records, view transaction history, schedule reservations at a particular clinic, register at more than one clinic, and look for location of the nearest animal clinics. The app is built with android based using waterfall method, using Java and PHP programming language, and using Android Studio tools. With the construction of this application can help the patient owner to know the patient's data completely, to reserve the schedule, to look for location of the nearest clinic, and to assist the registration process on more than one clinic via smartphone.

Keywords: Waterfall, Android, Animal Clinic, Java, PHP, Smartphone

I. PENDAHULUAN

Penggunaan teknologi saat ini telah dimanfaatkan untuk berbagai hal. Menurut [1], data pengguna android di Indonesia sejak tahun 2012 terus mengalami peningkatan. Mulai dari

Januari 2017 sampai dengan Juli 2017, pengguna android di Indonesia meningkat sebanyak 6,72% yaitu dari 77,27% hingga 83,99%. Salah satu pemanfaatan teknologi informasi saat ini yaitu untuk mencari informasi mengenai pemeliharaan hewan. Seiring berkembangnya zaman, minat masyarakat untuk memelihara hewan semakin tinggi. Ketertarikan seseorang untuk memelihara hewan biasanya karena hewan peliharaan bisa menjadi teman bahkan keluarga, sebagai penyaluran hobi, sebagai cerminan diri, sebagai properti atau benda, dan sebagai sumber pendapatan dari segi bisnis. Banyak aspek yang harus diperhatikan dalam memelihara hewan, salah satunya adalah kesehatan dari hewan peliharaan itu sendiri. Salah satu cara untuk memantau kesehatan dari hewan peliharaan tersebut adalah dengan melakukan pemeriksaan dan vaksinasi di klinik hewan terdekat.

Banyaknya peminat dalam memelihara hewan, hal ini membuka kesempatan bagi dokter hewan untuk membuka praktik di rumah bahkan membuka klinik hewan. Sampai saat ini sudah ada cukup banyak klinik hewan yang tersebar di Bandung namun lokasinya masih sulit ditemukan. Dalam perkembangannya, tidak sedikit masalah yang dialami oleh pihak klinik hewan maupun pemilik pasien peliharaan dalam berbagai aspek mulai dari pendaftaran pasien hingga transaksi yang dilakukan. Berdasarkan hasil kuesioner untuk pemilik pasien, terdapat beberapa masalah pada proses pendaftaran hingga proses transaksi di klinik. Sekitar 80% Masalah tersebut antara lain pendaftaran secara berulang pada klinik yang sama, pemilik pasien tidak memiliki salinan rekam medis dari hewan peliharaan, pemilik tidak bisa melihat riwayat transaksi yang sudah dilakukan secara lengkap, kesulitan mencari lokasi klinik hewan terdekat, dan kesulitan pemilik pasien mengetahui jadwal praktik dokter.

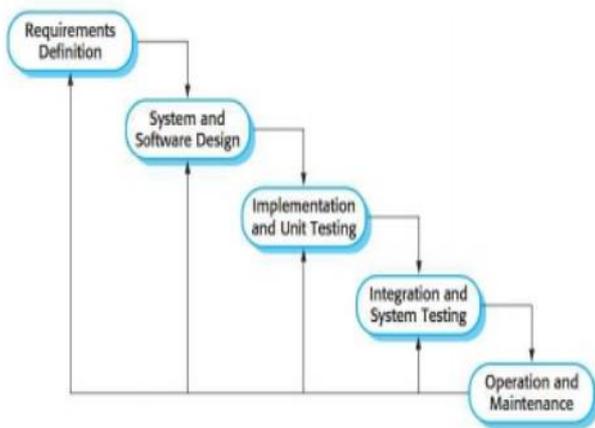
Berdasarkan masalah diatas maka dibuatlah Aplikasi Pendaftaran dan Transaksi Pasien Berbasis Android untuk pemilik pasien yang dapat digunakan untuk melakukan pendaftaran, melihat data pribadi dan hewan peliharaan secara lengkap, melihat jadwal praktik dokter dan reservasi jadwal,

dan melacak lokasi klinik hewan terdekat dari pemilik pasien berada.

Aplikasi ini diharapkan dapat memudahkan pemilik pasien dalam melakukan pendaftaran, melihat data pribadi dan hewan peliharaan secara lengkap, melihat jadwal praktik dokter dan reservasi jadwal, dan melihat lokasi klinik hewan terdekat.

II. METODE PENGEMBANGAN PERANGKAT LUNAK

Dalam pengembangan aplikasi ini sendiri digunakan sebuah metode pengerjaan dengan model pengembangan Linier Sequential Model atau biasa disebut Model Waterfall. Alasan penggunaan model Waterfall karena pihak *client* sendiri sangat sibuk dan sulit ditemui dan juga jam operasional setiap klinik yang berbeda serta jarak yang berjauhan. Selain itu keterbatasan sumber daya pengembang juga menjadi salah satu alasan dalam penggunaan model Waterfall.



Gambar 1 Classic Life Waterfall Model [2]

Berikut ini tahapan-tahapan pada model Waterfall:

1. Requirements Definition

Pada tahap ini, kebutuhan perangkat lunak harus didapatkan termasuk kegunaan perangkat lunak yang diharapkan pengguna. Informasi tersebut diperoleh dengan melakukan wawancara dan survey. Wawancara dilakukan dengan pihak klinik untuk mengetahui bagaimana pengelolaan data pasien pada klinik tersebut. Selain itu, survey dilakukan kepada pemilik pasien dengan mengisi kuesioner yang telah disediakan untuk mengetahui kebutuhan pengguna. Hasil dari wawancara dan survey dapat digunakan sebagai acuan untuk membangun aplikasi.

2. System and Software Design

Perancangan system yang akan dibuat dilakukan pada tahap ini sebelum dilakukan *coding*. Perancangan sistem menghasilkan gambaran dari aplikasi sesuai kebutuhan pengguna. Selain itu, tahap ini juga menspesifikasikan kebutuhan perangkat keras dan sistem serta pengguna dari aplikasi ini. Untuk menggambarkan rancangan aplikasi yang akan dibuat dapat menggunakan diagram seperti :

- a. *Entity Diagram Relationship*
- b. *Usecase Diagram*
- c. *Class Diagram*
- d. *Sequence Diagram*

3. Implementation and Unit Testing

Pada tahap ini dilakukan pembuatan aplikasi sesuai dengan rancangan yang telah dibuat pada tahap sebelumnya. Beberapa bahasa pemrograman yang digunakan dalam pembuatan aplikasi antara lain :

- a. Java
- b. PHP
- c. MySQL

4. Integration and System Testing

Setelah aplikasi dibuat, maka pada tahap ini dilakukan *testing* atau pengujian pada perangkat lunak yang dibuat. Pengujian dilakukan dengan metode *black box testing* untuk menguji apakah masih ada *bug* atau tidak dan menggunakan metode *User Acceptance Test (UAT)* yaitu pengujian oleh pengguna untuk mendapat konfirmasi bahwa aplikasi yang sudah dikembangkan dapat diterima oleh pengguna. Hasil dari pengujian ini harus sesuai dengan kebutuhan pengguna aplikasi.

5. Operation and Maintenance

Tahap ini merupakan pengoperasian pada lingkungan sebenarnya dan dilakukan pemeliharaan. Tahap ini tidak dilakukan karena keterbatasan waktu penulis dalam menyelesaikan proyek akhir.

III. TINJAUAN PUSTAKA

Berikut tinjauan pustaka yang menjadi referensi dalam penelitian ini:

A. Teori Pokok Pembahasan

1. Klinik Hewan

Klinik Hewan adalah tempat usaha pelayanan jasa medik veteriner yang dijalankan oleh suatu manajemen dengan dipimpin oleh seorang dokter hewan penanggungjawab dan memiliki fasilitas untuk pengamatan hewan yang mendapat gangguan kesehatan tertentu.

2. Hewan Peliharaan

Tren memelihara hewan kesayangan semakin meningkat. Banyak orang memelihara hewan kesayangan seperti anjing, kucing, kura-kura, bahkan ular untuk menghilangkan kejenuhan dan stres. Ramainya penjualan hewan di berbagai *petshop* akhir-akhir ini bisa dijadikan indikasi bahwa minat masyarakat dalam memelihara hewan semakin berkembang. Hewan peliharaan seperti anjing, kucing, kura-kura, dan beberapa hewan eksotis lainnya dapat dijadikan sebuah hiburan bagi pemeliharanya. Memelihara hewan dapat dilakukan oleh siapa saja, baik orang dewasa, orang tua, remaja, maupun anak-anak [3].

3. Android

Android merupakan sebuah sistem operasi telepon seluler dan computer tablet layar sentuh (*touch screen*) yang berbasis Linux. Namun seiring perkembangannya android telah berubah menjadi platform yang begitu cepat dalam melakukan inovasi. Hal ini tidak lepas dari pengembang utama, yaitu Google. Google-lah yang mengakuisisi android, kemudian membuat sebuah platform. Platform android terdiri dari Sistem Operasi Berbasis Linux, sebuah GUI (Graphic User Interface), sebuah Web Browser dan Aplikasi End-User yang dapat di-*download* dan juga para pengembang bisa dengan leluasa berkarya serta menciptakan aplikasi yang terbaik dan terbuka untuk digunakan oleh berbagai macam perangkat [4].

4. Perkembangan Sistem Operasi Android

a. Android versi 1.0 (Beta)

Versi beta dirilis pada 5 November 2007, kemudian dirilis versi komersialnya pada 23 September 2008 dengan memasukkan berbagai fitur seperti Android Market, Gmail, Google Maps, Web browser, dan sebagainya.

b. Android versi 1.1

Android versi 1.1 dirilis pada 9 Maret 2009 dengan pembaharuan seperti penambahan beberapa fitur mulai dari Google Mail Service (GMS), Alarm Clock, hingga File Attachment pada pesan.

c. Android versi 1.5 (Cupcake)

Versi 1.5 Cupcake dirilis pada 30 April 2009, terdapat fitur-fitur baru pada versi Cupcake seperti kemampuan mengunggah video ke Youtube, integrasi Home Screen dan Widget, copy paste pada browser, dan lain-lain.

d. Android versi 1.6 (Donut)

Android versi 1.6 Donut dirilis pada 15 September 2009 dan dibekali dengan fitur utama yaitu integrasi kamera,

video dan galeri, kemudian mendukung layar resolusi WVGA, serta Perbaikan Google Play (Android Market).

e. Android versi 2.0 – 2.1 (Éclair)

Android versi Éclair dirilis pada 9 Desember 2009 dengan beragam fitur yang menarik seperti dukungan Google Maps Beta, pencarian SMS, hingga Bluetooth 2.1 yang memungkinkan transfer data lebih cepat.

f. Android versi 2.2 – 2.3 (Froyo)

Android versi Froyo dirilis pada 20 Mei 2010 dengan penambahan fitur utama seperti Adobe Flash, Hotspot Portable, dan perekaman video dengan kualitas HD.

g. Android versi 2.3 – 2.3.7 (Gingerbread)

Android versi Gingerbread (versi 2.3, 2.3.2, 2.3.3, 2.3.7) dirilis pada 6 Desember 2010, pada versi ini telah dilakukan perbaikan Mockup hemat energi, dukungan terhadap NFC, papan ketik virtual, dan peningkatan fasilitas Copy Paste.

h. Android versi 3.0 – 3.2 (Honeycomb)

Android versi Hneycomb lebih diperuntukkan bagi pengguna perangkat tablet dengan Mockup yang lebih *user friendly*, fitur multitasking, dan fitur lainnya yang sesuai dengan perangkat tablet.

i. Android versi 4.0 (Ice Cream Sandwich)

Android versi 4.0 resmi dirilis pada 19 Oktober 2011 dengan penambahan fitur seperti membuka kunci dengan pengenalan wajah, perbaikan masukan teks dan suara, serta tombol virtual untuk menggantikan tombol fisik.

j. Android versi 4.1 – 4.3 (Jelly Bean)

Android versi Jelly Bean dirilis pada 9 Juli 2012, pada versi Jelly Bean terdapat perbaikan untuk fitur pada perangkat tablet, fitur baru seperti Bluetooth Smart Support, autocomplete pada dialpad, dan penambahan beberapa bahasa masukan teks.

k. Android versi 4.4 (KitKat)

Android versi Kitkat dirilis pada tanggal 31 Oktober 2013, terdapat pembaharuan pada versi ini seperti pembaharuan Mockup, mendukung Wireless Printing, hingga peningkatan performa dan keamanan.

l. Android versi 5.0 (Lollipop)

Pada android versi 5.0 Lollipop mengalami peningkatan yang sangat signifikan mulai dari Mockup baik pada smartphone maupun pada perangkat tablet, peningkatan performa dan keamanan, battery saver, dan dapat digunakan pada perangkat TV dan *wearable* lainnya yang didukung OS Android.

m. Android versi 6.0 (Marshmallow)

Android versi 6.0 Marshmallow dirilis pada tanggal 5 Oktober 2015. Pada versi ini terdapat beberapa penambahan jalan pintas atau *shortcut* dan permission,

durasi baterai yang lebih lama, serta membuka kunci dengan sidik jari (*fingerprint*).

n. Android versi 7.0 (Nougat)

Android versi 7.0 Nougat dirilis pada tanggal 22 Agustus 2016. Pada versi ini terdapat pembaharuan masukan emoji, peningkatan Mockup dan performa, mendukung *Double Window* yaitu membuka dua aplikasi tanpa menutup salah satunya. Pada versi ini juga ditambahkan fitur penting yaitu "Doze On the Go", fitur ini dapat memperkecil daya baterai yang digunakan oleh Smartphone ketika sedang disimpan di dalam saku.

o. Android versi 8.0 (Oreo)

Google kembali meresmikan nama Android versi terbaru yaitu Versi 8.0 dengan nama Oreo, namun belum dirilis secara resmi. Pada versi ini terdapat beberapa fitur baru seperti *picture in picture*, peningkatan penghematan baterai, kecepatan *booting* dua kali lebih cepat, *autofill with permission*, background limits, *notification dots* dan *Android Instant Apps*, serta peningkatan performa.

B. Alat Bantu Pemodelan Aplikasi

1. Business Process Model and Notation (BPMN)

BPMN (Business Process Model and Notation) adalah metode flowchart yang memodelkan langkah-langkah proses bisnis yang direncanakan dari ujung ke ujung. Tujuan utama dari BPMN adalah untuk menyediakan suatu notasi yang mudah dipahami oleh semua pengguna bisnis, mulai dari bisnis analis yang membuat draft dari awal proses, para pengembang teknis yang bertanggung jawab untuk menerapkan teknologi yang akan melakukan proses-proses tersebut, hingga kepada orang-orang bisnis yang akan mengelola dan memantau proses mereka [5].

2. Unified Model Language (UML)

Unified Modelling Language (UML) adalah sebuah "bahasa" yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem perangkat lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. UML dapat membuat model untuk semua jenis aplikasi perangkat lunak, di mana aplikasi tersebut dapat berjalan pada perangkat keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML juga menggunakan Class dan Operation dalam konsep dasarnya, maka ia lebih cocok untuk penulisan perangkat lunak dalam bahasa - bahasa berorientasi objek seperti C++, Java, C# atau VB.NET. Walaupun demikian, UML tetap dapat digunakan untuk membuat model aplikasi prosedural dalam VB atau C.

3. Use Case Diagram

UML menyediakan serangkaian gambar dan diagram yang sangat baik. Salah satu diagram yang disediakan adalah Use Case Diagram. Salah satu kontributor terhadap diagram use case dalam UML adalah Ivar Jacobsen. Use case menggambarkan external view dari sistem yang akan dibuat modelnya. Pooley (2003:15) mengatakan bahwa model use case dapat dijabarkan dalam diagram use case, tetapi perlu diingat bahwa diagram tidak identik dengan model karena model lebih luas dari diagram. Komponen pembentuk diagram use case adalah aktor, use case, dan hubungan (*link*).

4. Class Diagram

Menurut John Satzinger, 2010, dalam buku *System Analysis and Design in a Changing World* menyatakan bahwa dalam UML ada dua jenis diagram yaitu domain class diagram dan design class diagram. Fokus domain class diagram adalah pada sesuatu dalam lingkungan kerja pengguna, bukan pada class perangkat lunak yang nantinya akan dirancang. Sedangkan design class diagram menggambarkan kelas berorientasi objek yang dibutuhkan dalam pemrograman, navigasi dia antara kelas, *attribute name*, serta *method names* dan propertinya. Design class diagram inilah yang akan digunakan dalam merancang aplikasi yang akan dibuat [6].

5. Entity Relationship Diagram (ER-D)

ER-D adalah merupakan salah satu model yang digunakan untuk mendesain database dengan tujuan menggambarkan data yang berelasi pada sebuah database. Umumnya setelah perancangan ERD selesai berikutnya adalah mendesain database secara fisik yaitu pembuatan tabel, index dengan tetap mempertimbangkan performance. Kemudian setelah database selesai dilanjutkan dengan merancang aplikasi yang melibatkan database.

6. Sequence Diagram

Sequence diagram adalah diagram yang menggambarkan interaksi antar objek. Sequence diagram secara khusus menjabarkan behavior sebuah skenario tunggal. Diagram tersebut menunjukkan sejumlah objek contoh dan pesan-pesan yang melewati objek ini dalam sebuah use case [7].

7. Mockup

MockUp adalah sebuah media visual atau preview dari sebuah konsep desain "datar" yang diberikan efek visual sehingga hasilnya sangat tempak atau menyerupai wujud nyata, mockup dapat memberikan gambaran nyata dari

sebuah konsep desain bagaimana konsep itu akan terlihat nantinya jika sudah diaplikasikan menjadi atau kedalam benda nyata, apakah terlihat bagus atau kurang sesuai. Mockup juga dapat menambahkan nilai visual dari sebuah konsep desain, misalnya sebuah desain dibuat dalam versi datar maka desain itu akan terlihat bisa-bisa saja, namun hanya dengan menggunakan mockup untuk mentweak sebuah konsep desain maka desain tersebut akan terlihat lebih menarik dan mengundang apresiasi [8].

C. Alat Bantu Pembangunan Aplikasi

1. Android Studio

Android Studio adalah Lingkungan Pengembangan Terpadu - Integrated Development Environment (IDE) untuk pengembangan aplikasi Android, berdasarkan IntelliJ IDEA. Selain merupakan editor kode IntelliJ dan alat pengembang yang berdaya guna, Android Studio menawarkan fitur lebih banyak untuk meningkatkan produktivitas Anda saat membuat aplikasi Android, misalnya:

- Sistem versi berbasis Gradle yang fleksibel
- Emulator yang cepat dan kaya fitur
- Lingkungan yang menyatu untuk pengembangan bagi semua perangkat Android
- Instant Run untuk mendorong perubahan ke aplikasi yang berjalan tanpa membuat APK baru
- Template kode dan integrasi GitHub untuk membuat fitur aplikasi yang sama dan mengimpor kode contoh
- Alat pengujian dan kerangka kerja yang ekstensif
- Alat Lint untuk meningkatkan kinerja, kegunaan, kompatibilitas versi, dan masalah-masalah lain
- Dukungan C++ dan NDK
- Dukungan bawaan untuk , mempermudah pengintegrasian Google Cloud Messaging dan App Engine [9]

2. Java

Java adalah bahasa pemrograman berorientasi objek yang dikembangkan oleh Sun Microsystems sejak tahun 1991. Rilis resmi level beta pada November 1995. Java telah berjalan pada segala perangkat dari laptop sampai pusat data, konsol *game* sampai supercomputer ilmiah, Java merupakan bahasa orientasi objek untuk pengembangan aplikasi mandiri, aplikasi berbasis Internet, aplikasi untuk perangkat cerdas yang dapat berkomunikasi lewat Internet atau jaringan komunikasi. Java telah menjadi bahasa untuk pengembangan aplikasi skala enterprise berbasis jaringan besar [10]. Seperti pada perangkat Android dari Google ,dan

Beberapa Aplikasi Desktop ,seperti program AntiVirus , program media player, Aplikasi Enterprise dll.

3. MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang multithread dan multi-user. MySQL adalah Relational Database Management System (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (General Public License). Setiap orang bebas menggunakan MySQL.

MySQL sebenarnya merupakan turunan dari SQL (Structured Query Language). SQL adalah sebuah konsep pengoperasian database, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Sebagai database server, MySQL dapat dikatakan lebih unggul dibandingkan database server lainnya dalam query data. Hal ini terbukti untuk query yang dilakukan oleh single user, kecepatan query MySQL bisa sepuluh kali lebih cepat dari PostgreSQL dan lima kali lebih cepat dibandingkan Interbase [11].

MySQL dikembangkan oleh sebuah perusahaan Swedia bernama MySQL AB, yang kala itu bernama TcX DataKonsult AB, sejak sekitar 1994- 1995 meski cikal bakal kodenya bisa disebut sudah ada sejak 1979. MySQL versi 1.0 dirilis Mei 1996 secara terbatas kepada 4 orang. Baru dibulan Oktober versi 3.11. dilepas ke publik. Namun, mula-mula kode ini tidak dibeikan di bawah lisensi General Public License, melainkan lisensi khusus yang intinya kurang lebih begini, Source Code MySQL dapat dilihat dan gratis, serta server MySQL dapat dipakai tanpa biaya, tapi hanya untuk kebutuhan nonkomersial [12].

4. PHP

PHP merupakan singkatan dari *Hypertext Preprocessor*. Pada awalnya PHP merupakan kependekan dari Personal Home Page (situs personal) dan PHP itu sendiri pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995, dan pada saat PHP masih bernama FI (Form Interpreter), yang wujudnya berupa sekumpulan script yang digunakan untuk mengolah data form dari web. Selanjutnya Rasmus merilis kode sumber tersebut untuk umum. PHP adalah sebuah bahasa scripting yang terpasang pada HTML. Sebagian besar sintaksnya mirip dengan bahasa pemrograman C, Java, ASP dan Perl, ditambah beberapa fungsi PHP yang spesifik dan mudah dimengerti.

PHP digunakan untuk membuat tampilan web menjadi lebih dinamis, PHP bisa menampilkan atau menjalankan beberapa file dalam satu file dengan cara di-*include* atau *require*. PHP itu sendiri sudah dapat berinteraksi dengan beberapa

database walaupun dengan kelengkapan yang berbeda, yaitu seperti: DBM, FilePro, (Personic, Inc), Informix, Ingres, Interbase, Microsoft Access, MSSQL, MySQL, Oracle, PostgrSQL, dan Sybase [13].

5. Framework Slim

Slime merupakan *framework* PHP yang dapat memudahkan penulisan kode agar lebih singkat dalam membangun *website* dan API. Slim memiliki beberapa fitur antara lain adalah menyediakan jalur lebih cepat dengan memanggil metode khusus HTTP dan URI dengan parameter. Slim mendukung implementasi pesan PSR-7 HTTP sehingga dapat menginspeksi dan memanipulasi pesan metode HTTP, *status*, *URI*, *headers*, *cookies* dan *body*. Selain itu, Slim mendukung injeksi dependensi sehingga pengguna dapat mengendalikannya dengan perangkat lain [14].

6. JSON

JSON (*JavaScript Object Notation*) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (*generate*) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemrograman JavaScript}, Standar ECMA-262 Edisi ke-3 - Desember 1999. JSON merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran-data.

JSON terbuat dari dua struktur:

- Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (*object*), rekaman (*record*), struktur (*struct*), kamus (*dictionary*), tabel hash (*hash table*), daftar berkunci (*keyed list*), atau *associative array*.
- Daftar nilai terurutkan (*an ordered list of values*). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (*array*), vektor (*vector*), daftar (*list*), atau urutan (*sequence*).

Struktur-struktur data ini disebut sebagai struktur data universal. Pada dasarnya, semua bahasa pemrograman moderen mendukung struktur data ini dalam bentuk yang sama maupun berlainan. Hal ini pantas disebut demikian karena format data mudah dipertukarkan dengan bahasa-bahasa pemrograman yang juga berdasarkan pada struktur data ini.

JSON menggunakan bentuk sebagai berikut:

- a. Objek adalah sepasang nama/nilai yang tidak terurutkan. Objek dimulai dengan { (kurung kurawal buka) dan

diakhiri dengan } (kurung kurawal tutup). Setiap nama diikuti dengan : (titik dua) dan setiap pasangan nama/nilai dipisahkan oleh , (koma).

- b. Larik adalah kumpulan nilai yang terurutkan. Larik dimulai dengan [(kurung kotak buka) dan diakhiri dengan] (kurung kotak tutup). Setiap nilai dipisahkan oleh , (koma).
- c. Nilai (*value*) dapat berupa sebuah string dalam tanda kutip ganda, atau *angka*, atau *true* atau *false* atau *null*, atau sebuah *objek* atau sebuah *larik*. Struktur-struktur tersebut dapat disusun bertingkat.
- d. String adalah kumpulan dari nol atau lebih karakter Unicode, yang dibungkus dengan tanda kutip ganda. Di dalam string dapat digunakan *backslash escapes* "\" untuk membentuk karakter khusus. Sebuah karakter mewakili karakter tunggal pada string. String sangat mirip dengan string C atau Java.
- e. Angka adalah sangat mirip dengan angka di C atau Java, kecuali format oktal dan heksadesimal tidak digunakan.

Spasi kosong (*whitespace*) dapat disisipkan di antara pasangan tanda-tanda tersebut, kecuali beberapa detil *encoding* yang secara lengkap dipaparkan oleh bahasa pemrograman yang bersangkutan [15].

7. Apache

Apache HTTP Server Project adalah upaya pengembangan perangkat lunak kolaboratif yang ditujukan untuk menciptakan implementasi kode sumber HTTP, Web-grade, featureful, dan server HTTP (Web) yang tersedia dengan source terbuka. Proyek ini dikelola bersama oleh sekelompok sukarelawan yang berada di seluruh dunia, menggunakan Internet dan Web untuk mengkomunikasikan, merencanakan, dan mengembangkan server dan dokumentasi terkaitnya. Proyek ini merupakan bagian dari Apache Software Foundation. Selain itu, ratusan pengguna telah menyumbangkan ide, kode, dan dokumentasi untuk proyek ini. File ini dimaksudkan untuk secara singkat menggambarkan sejarah Apache HTTP Server dan mengenali banyak kontributor [16].

8. Extensible Markup Language (XML)

XML terletak pada inti web service yang digunakan untuk mendeskripsikan data. Fungsi utama dari XML adalah komunikasi antar aplikasi, integrasi data, dan komunikasi aplikasi eksternal dengan partner luaran. Dengan standardisasi XML, aplikasi-aplikasi yang berbeda dapat dengan mudah berkomunikasi antar satu dengan yang lain. SML adalah singkatan dari *Xtensible Markup Language*. Bahasa *markup* adalah sekumpulan aturan-aturan yang mendefinisikan suatu sintaks yang digunakan untuk

menjelaskan dan mendeskripsikan teks atau data dalam sebuah dokumen melalui penggunaan tag. XML juga menggunakan elemen yang ditandai dengan tag pembukan, tag penutup, dan atribut elemen. Pada XML, pembuat bisa menggunakan tag dan atribut sesuai kehendak pembuatnya [4].

9. Black Box Testing

Pengujian Kotak Hitam (*Blackbox Testing*) khusus di didesain untuk mencari kesalahan dengan melakukan ujicoba pada interface software. Pegujian Kotak Hitam (*Blackbox Testing*) mendemonstrasikan fungsi dari perangkat lunak yang beroperasi, dengan mengecek apakah input sudah bisa diterima dengan baik, dan hasil outputnya sesuai dengan apa yang diharapkan. Uji coba Kotak Hitam (*Blackbox Testing*) melakukan pengecekan pada integritas informasi eksternal. Pada dasarnya pengujian Kotak Hitam (*Blackbox Testing*) hanya memeriksa hasil output yang dihasilkan apakah sudah sesuai dengan apa yang diharapkan dan dinyatakan benar. Namun pengujian Kotak Hitam (*Blackbox Testing*) tidak mengecek logika dari perangkat lunak.

Pengujian unit merupakan pengujian secara individual terhadap semua program untuk memastikan bahwa program bebas dari kesalahan. Namun jika ditemukan *error* atau kesalahan pada program, *user* akan langsung mencari kesalahannya dan proses untuk melakukan pencarian kesalahan ini dikenal dengan *debugging*.

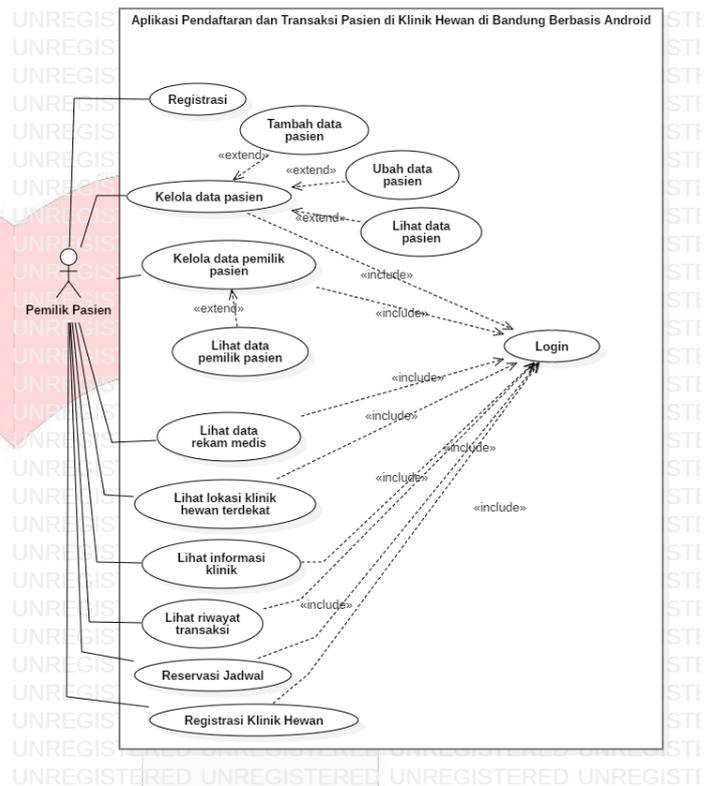
10. User Acceptance Test

User Acceptance Test (UAT) yaitu pengujian oleh pengguna untuk mendapat konfirmasi bahwa aplikasi yang sudah dikembangkan dapat diterima oleh pengguna. Pada jenis pengujian ini, perangkat lunak akan diserahkan pada pengguna untuk mengetahui apakah perangkat lunak memenuhi harapan pengguna dan bekerja seperti yang diharapkan. Pada pengembangan perangkat lunak, *User Acceptance Test* (UAT), juga disebut pengujian beta (*beta testing*), pengujian aplikasi (*application testing*), dan pengujian pengguna akhir (*end user testing*) adalah tahapan pengembangan perangkat lunak ketika perangkat lunak diuji pada “dunia nyata” yang dimaksudkan oleh pengguna. UAT dapat dilakukan dengan *in-house testing* dengan membayar relawan atau subjek pengujian menggunakan perangkat lunak atau biasanya mendistribusikan perangkat lunak secara luas dengan melakukan pengujian versi yang tersedia secara gratis untuk diunduh melalui web. Pengalaman awal pengguna akan diteruskan kembali kepada para pengembang yang membuat perubahan sebelum akhirnya melepaskan perangkat lunak komersial [17].

IV. ANALISIS DAN PERANCANGAN

1. Use Case Diagram

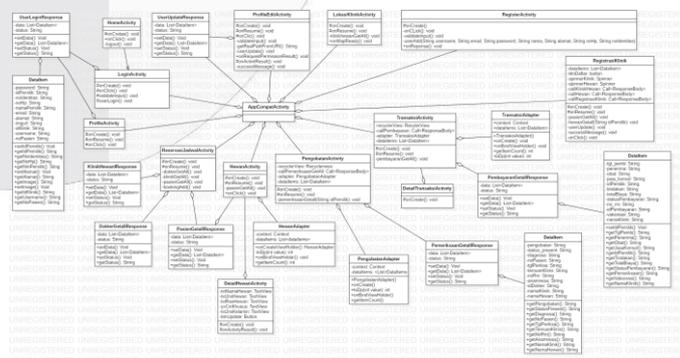
Berikut merupakan Use Case Diagram dari Aplikasi Pendaftaran dan Transaksi di Klinik Hewan di Bandung Berbasis Android:



Gambar 2 Use Case Diagram Aplikasi dan Transaksi Pasien (Pengelolaan Data Klinik)

2. Class Diagram

Berikut merupakan class diagram dari aplikasi yang dibangun pada penelitian ini:



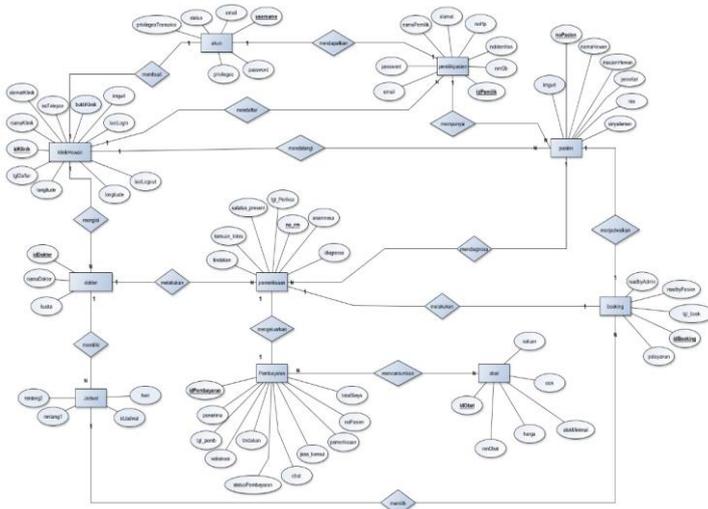
Gambar 3 Class Diagram Aplikasi

3. ER-Diagram

Berikut rancangan ER-Diagram dari aplikasi yang dibangun dalam penelitian ini:

V. IMPLEMENTASI

Dalam pengimplementasian Aplikasi Pendaftaran dan Transaksi di Klinik Hewan di Bandung dapat digunakan dengan cara menginstal aplikasi tersebut pada smartphone dengan sistem operasi Android. Aplikasi ini dapat digunakan oleh pemilik pasien dengan beberapa fitur antara lain menambah data hewan, daftar pada lebih dari satu klinik hewan, reservasi jadwal, melihat catatan pengobatan, melihat riwayat transaksi, dan melihat lokasi klinik hewan terdekat. Berikut tampilan antarmuka dari Aplikasi Pendaftaran dan Transaksi di Klinik Hewan di Bandung (Gambar 4.1 s.d. Gambar 4.22):



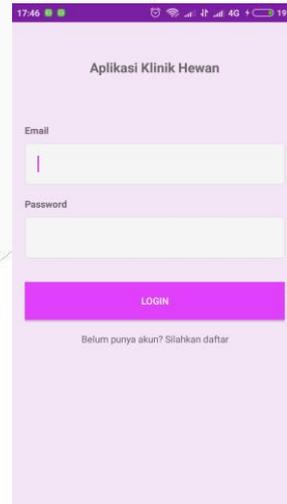
Gambar 4 ER-Diagram Aplikasi Pendaftaran dan Transaksi Pasien

1. Halaman Registrasi



Gambar 6 Halaman Registrasi

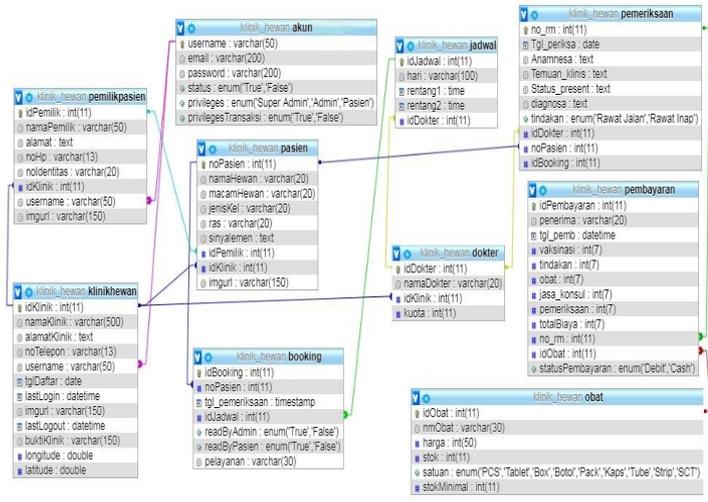
2. Halaman Login



Gambar 7 Halaman Login

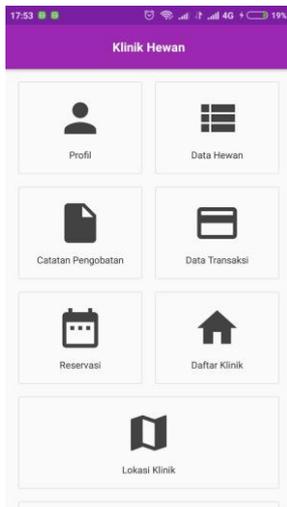
4. Relasi Antartabel

Berikut skema relasi antartabel dari aplikasi yang dibangun dalam penelitian ini:



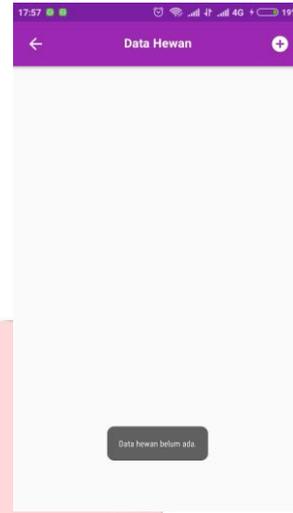
Gambar 5 Tabel Relasi Basis Data Klinik Hewan

3. Halaman Utama



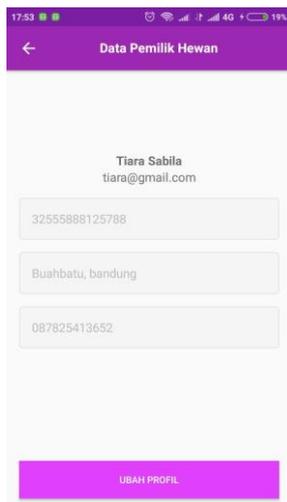
Gambar 8 Halaman Utama Aplikasi

5. Halaman Menu Data Hewan

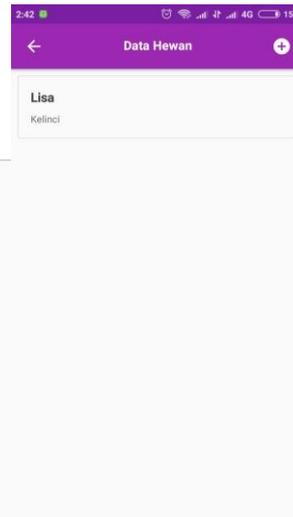


Gambar 11 Halaman Data Hewan (Jika tidak ada data)

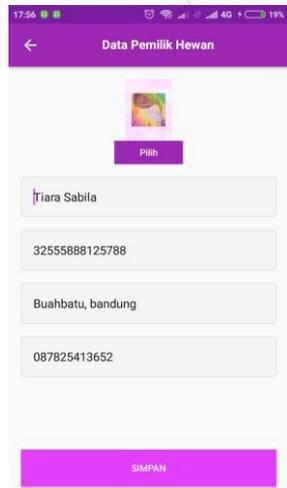
4. Halaman Profil



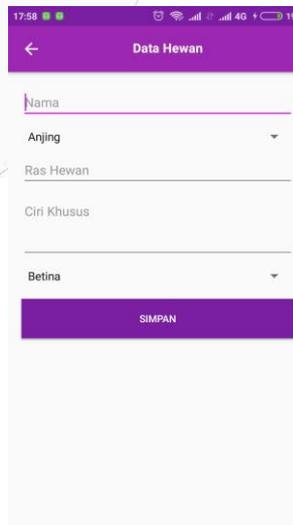
Gambar 9 Halaman Profil Pemilik Pasien



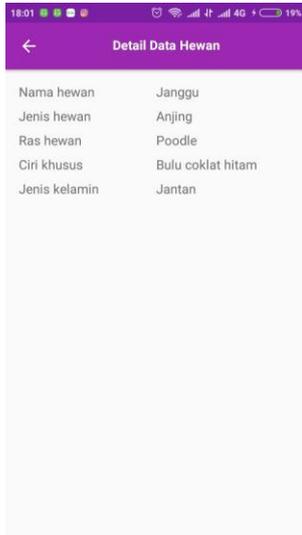
Gambar 12 Halaman List Data Hewan



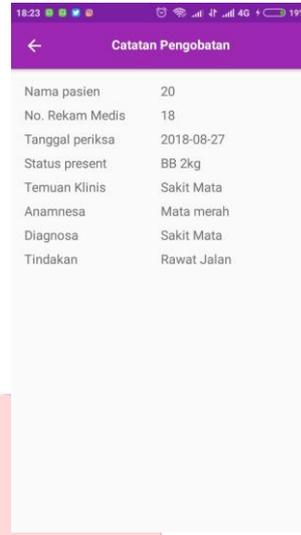
Gambar 10 Halaman Ubah Profil Pemilik Pasien



Gambar 13 Halaman Tambah Data Hewan

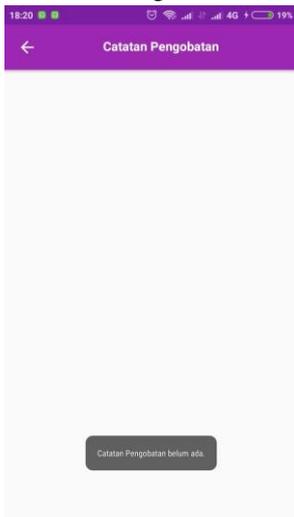


Gambar 14 Halaman Rincian Data Hewan



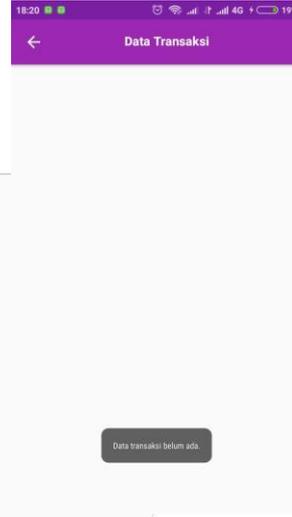
Gambar 17 Halaman Rincian Catatan Pengobatan

6. Halaman Menu Catatan Pengobatan



Gambar 15 Halaman Catatan Pengobatan (Jika tidak ada data)

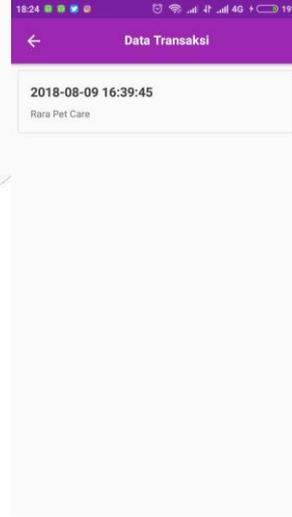
7. Halaman Menu Data Transaksi



Gambar 18 Halaman Data Transaksi (Jika tidak ada data)

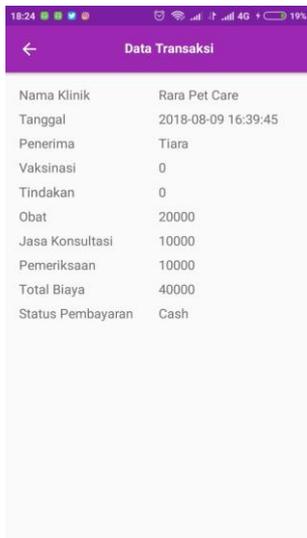


Gambar 16 Halaman List Catatan Pengobatan

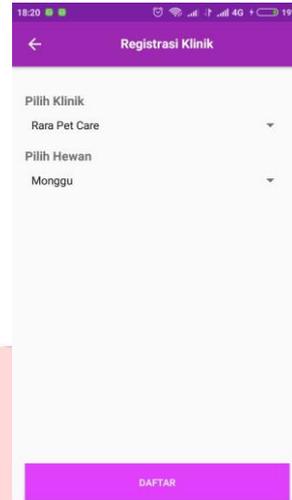


Gambar 19 List Data Transaksi

9. Halaman Menu Daftar Klinik

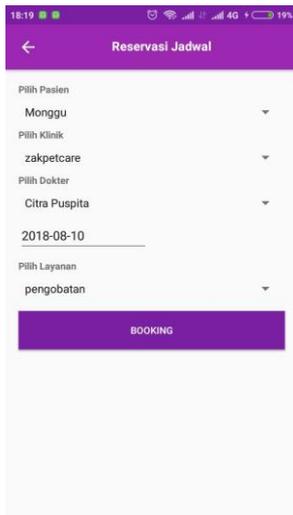


Gambar 20 Halaman Rincian Data Transaksi

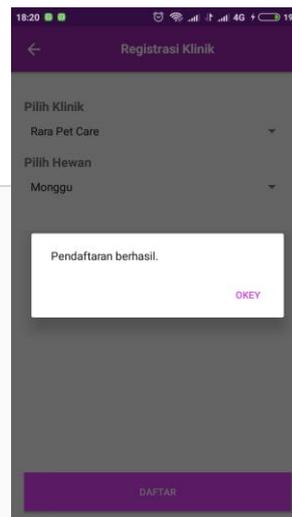


Gambar 23 Halaman Daftar Klinik

8. Halaman Menu Reservasi Jadwal

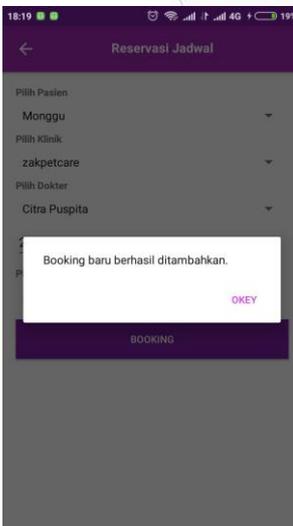


Gambar 21 Halaman Form Reservasi Jadwal



Gambar 24 Dialog Daftar Klinik

10. Halaman Menu Lokasi

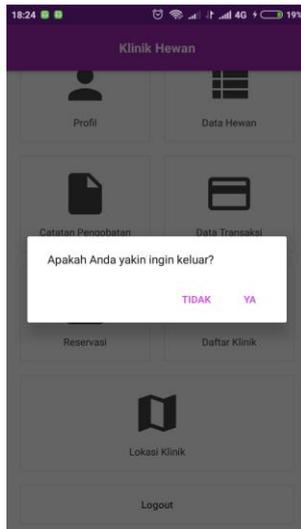


Gambar 22 Dialog Reservasi Jadwal



Gambar 25 Halaman Lihat Lokasi Klinik Hewan

11. Halaman Menu Logout



Gambar 26 Konfirmasi Logout

VI. KESIMPULAN

Dari aplikasi yang telah dibuat dapat disimpulkan bahwa, Aplikasi Pendaftaran dan Transaksi di Klinik Hewan di Bandung dapat digunakan oleh pemilik pasien. Pemilik pasien dapat melihat informasi mengenai data pendaftaran dan transaksi secara lengkap. Pada aplikasi ini, pemilik pasien dapat mendaftar pada lebih dari satu klinik. Selain itu, pemilik pasien dapat melihat dan mengubah profil pemilik pasien, menambahkan dan melihat data hewan, melihat catatan pengobatan dari berbagai klinik, melihat riwayat transaksi di berbagai klinik yang terdaftar, reservasi jadwal pada klinik yang terdaftar, serta dapat melihat lokasi klinik hewan terdekat yang terdaftar pada aplikasi.

REFERENSI

- [1] Statista, "Statista," *The Statistics Journal*. [Online]. Available: <https://www.statista.com/statistics/263453/global-market-share-held-by-smartphone-operating-systems/>.
- [2] "Waterfall Model of SDLC," *I Answer 4 U*. [Online]. Available: <https://www.ianswer4u.com/2011/11/waterfall-model.html>.
- [3] R. AgroMedia, *Buku Pintar Merawat Hewan Kesayangan*. Jakarta: PT Agromedia Pustaka, 2008.
- [4] A. Dharma K, *Trik Kolaborasi Android dengan PHP+MySQL*. Yogyakarta: Penerbit Lokomedia, 2015.
- [5] R. Satria, "Brainmatics." [Online]. Available: <http://brainmatics.com/businesss-process-model-and-notation-bpmn/>.

- [6] E. Triandini, *Step by Step Design Proyek Menggunakan UML*. Yogyakarta: CV. Andi Offset, 2012.
- [7] S. Mulyani, *Metode Analisis dan Perancangan Sistem*. 2016.
- [8] R. Adi S, "Primecyber," 2015. [Online]. Available: <http://www.primecyber.com/2015/11/apa-pengertian-mockup.html>.
- [9] Android, "Android," *Android*. [Online]. Available: <https://developer.android.com/studio/intro/index.html?hl=id>.
- [10] B. Hariyanto, *Esensi-Esensi Bahasa Pemrograman Java*. Bandung: Penerbit Informatika Bandung, 2014.
- [11] A. Kristanto, *Kupas Tuntas PHP & MySQL*. Penerbit Cable Book, 2010.
- [12] E. Mardiani, N. Rahmansyah, H. Kurniawan, A. Muliawati, and D. Sidik, *Membuat Aplikasi penjualan menggunakan Java Netbeans, MySQL, dan iReport*. Jakarta: ElexMedia Komputindo, 2017.
- [13] *Kupas Tuntas Adbe Dreamwaver dengan Pemrograman PHP&MySQL*. Yogyakarta: Penerbit CV. Andi Offset & Madooms, 2013.
- [14] "Slim," *Slim a micro framework PHP*. [Online]. Available: <https://www.slimframework.com/>. [Accessed: 04-Jul-2018].
- [15] "JSON," *JSON*. [Online]. Available: <http://www.json.org/json-id.html>.
- [16] APACHE, "The Apache HTTP Server Project." [Online]. Available: https://httpd.apache.org/ABOUT_APACHE.html.
- [17] "Iansyah Belajar," 2016. [Online]. Available: <http://www.iansyahbelajar.com/2016/06/apa-itu-pengujian-kotak-hitam.html>.

