

BAB II

TINJAUAN PUSTAKA

II.1. Penelitian Terkait

Untuk mendukung keberhasilan penelitian ini, penulis menggunakan referensi dari jurnal-jurnal dan penelitian terkait untuk membantu penelitian yang dilakukan agar sesuai dengan yang diharapkan.

Penelitian yang dilakukan oleh Dede Abdurahman, Muhamad Ade Kurnia, 2019, “Rancang Bangun Aplikasi Pelayanan Donor Darah Pada PMI Kabupaten Majalengka Berbasis Androis dan Web Menggunakan Teknologi Firebase”. Penelitian ini telah menghasilkan aplikasi dan fitur-fitur dari Firebase kedalam aplikasi pelayanan donor darah hingga tahap installasi pada perangkat web dan perangkat smartphone berbasis.

Penelitian yang dilakukan oleh Rizal Rachmad dari STMIK Nusa Mandiri Jakarta pada tahun 2019 yang berjudul “Pengembangan Sisitem Informasi Donor

Darah Berbasis Web Pada Unit Transfusi Darah Kabupaten sumedang”. Penelitian ini telah menghasilkan suatu website donor darah dan membantu masyarakat. Adapun desain *interface* yang dihasilkan meliputi desain *form* pendaftaran, *form login*, *form* permintaan darah, *form* berita acara pengiriman, *form* kelola donor, hingga *form* riwayat donor. Dan untuk desain sistem meliputi pembuatan *Entity Relational Diagram* (ERD), *Logical Relational Structure* (LRS) dan, *Undifined Model Language* (UML).

Penelitian yang dilakukan oleh Tommy Satrio Adriyanto, Purwatiningtyas dan Askar Yunianto dari Universitas Stikubank pada tahun 2019 yang berjudul “Sistem Informasi Pendaftaran Donor Darah Online Untuk Mengetahui Ketersediaan Darah Donor di UUD Palang Merah Indonesia Kota Semarang Berbasis Web Mobile”. Penelitian ini telah menghasilkan suatu sistem informasi pendaftaran donor darah *online* untuk mengetahui ketersediaan darah donor di UDD Palang Merah Indonesia kota Semarang berbasis web mobile terdiri dari 2 halaman yaitu halaman pengguna untuk melakukan melakukan donor darah, melihat stok darah donor, melihat jadwal mobile unit di UDD PMI kota Semarang yang dibuat dalam bentuk web *mobile* sedangkan halaman admin digunakan untuk mengelola data darah donor, data pasien, data darah masuk, data darah keluar, data jadwal mobile unit UDD PMI kota Semarang, mencetak laporan yang dibuat dalam bentuk web.

Penelitian yang dilakukan oleh Begawan Raka Sakti, Wina Witanti, dan Asep Id Hadiana dari Universitas Jenderal Achmad Yani Indonesia pada tahun 2021 yang berjudul “Sistem Informasi Bank Darah dengan Location Based Service untuk Meningkatkan Efisiensi Pencarian Golongan Darah (Studi Kasus: UTD PMI Cimahi)”. Penelitian ini telah menghasilkan Sistem dapat membagikan informasi mengenai pencarian letak bank darah terdekat dari lokasi pengguna saat ini, sistem dapat mengelola pendaftaran donor darah yang diajukan resipien, sistem dapat

mengelola pengambilan stok darah, dan sistem dapat menyebarkan informasi seputar palang merah indonesia, bank darah, unit transfusi darah, *event* donor darah, dan syarat untuk menjadi pendonor darah.

Pada penelitian ini penulis menerapkan metode Location Based Service (LBS) untuk mengatasi masalah yang dihadapi oleh pendonor, karna mengimplementasikan metode Location Based Service ini dapat membantu pendonor agar tau lokasi PMI yang ada disekitarnya,

II.2. Landasan Teori

Adapun landasan teori yang digunakan sebagai landasan dalam penulisan proposal skripsi ini adalah sebagai berikut :

II.2.1. Perancangan

Perancangan adalah suatu proses yang bertujuan untuk menganalisis, menilai, memperbaiki dan menyusun suatu sistem, baik sistem fisik maupun non fisik untuk waktu yang akan datang dengan memanfaatkan informasi yang ada. Dalam perancangan ada beberapa tahapan yang harus dilakukan sebelum hasil dari rancangan dijadikan sebuah produk berjenis aplikasi.

Menurut Soetam Rizky (2011), “ Perancangan adalah sebuah proses untuk mendefinisikan sesuatu yang akan dikerjakan dengan menggunakan teknik yang bervariasi serta didalamnya melibatkan deskripsi mengenai arsitektur serta detail komponen dan juga keterbatasan yang akan dialami dalam proses pengerjaannya”

Menurut Deddy Ackbar Rianto (2016), “ Perancangan dapat diartikan perencanaan dari pembuatan suatu sistem yang menyangkut berbagai komponen sehingga akan menghasilkan sistem yang sesuai dengan hasil dari tahap analisa sistem”

Berdasarkan definisi tersebut dapat disimpulkan bahwa perancangan adalah perencanaan dan pembuatan dengan menggunakan teknik yang bervariasi serta didalamnya melibatkan deskripsi mengenai arsitektur serta detail komponen dan juga keterbatasan yang akan dialami dalam proses pengerjaannya serta memungkinkan terjadinya kegagalan yang harus diselesaikan menyangkut pengaturan dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem yang berjalan.

II.2.2. Donor Darah

Donor darah adalah suatu proses pengambilan darah dari diri seseorang secara sukarela untuk disimpan di bank darah sebagai stok darah untuk kemudian digunakan para tranfusi darah. Dan setiap beberapa waktu, adapula penggalangan pemyumbangan darah yang diadakan di tempat-tempat umum seperti di pusat perbelanjaan, perusahaan tempat ibadah, dan sebagainya.

II.2.3. *Location Based Service (LBS)*

Menurut Sfaat (2015:26) Location Based Service (LBS) atau layanan berbasis lokasi adalah layanan informasi yang dapat diakses melalui mobile device dengan menggunakan mobile network, yang dilengkapi kemampuan untuk memanfaatkan lokasi dan mobile device tersebut. LBS memberikan kemungkinan komunikasi dan interaksi dua arah. Selain itu LBS digunakan untuk menggambarkan teknologi yang digunakan untuk menemukan lokasi perangkat atau suatu objek tertentu (Sfaat, 2012:218), seperti menemukan lokasi mesin ATM terdekat atau mengetahui keberadaan teman. Layanan berbasis lokasi dapat digambarkan sebagai suatu layanan yang berada pada pertemuan tiga teknologi yaitu :Geographic Information System, Internet Service, Mobile Devices. Secara garis besar Location Based Service dibagi menjadi dua (Sucinta, 2012), yaitu Pull

Service yaitu layanan yang diberikan jika ada permintaan dari pelanggan akan kebutuhan suatu informasi. Jenis layanan ini dapat dianalogikan seperti mengakses suatu web pada jaringan internet. Untuk Pull Service bias dibagi lagi menjadi dua yaitu berdasarkan layanan fungsional seperti memesan ambulan dengan menekan tombol pada device atau layanan service seperti mencari lokasi bidan/dokter terdekat dari posisi pengguna. Dan yang adalah Push Service yaitu layanan ini diberikan langsung oleh service provider tanpa menunggu permintaan dari pelanggan berupa informasi yang berkaitan dengan kebutuhan pelanggan.

II.2.3. Android

Android adalah sistem operasi untuk telepon seluler yang berbasis Linux. Android menyediakan platform terbuka bagi para pengembang buat menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. Awalnya, Google Inc. Membeli Android Inc., Pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan Android, dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia. (master.com; 2014:5)

Pada saat perilisan perdana Android, 5 November 2007, Android bersama *Open Handset Alliance* menyatakan mendukung pengembangan *Open Source* pada perangkat *Mobile*. Di lain pihak, *Google* merilis kode-kode Android di bawah lisensi *Apache*, sebuah lisensi perangkat lunak dan *open platform* perangkat seluler.

Di dunia ini terdapat dua jenis distributor sistem operasi Android. Pertama yang dapat mendukung penuh dari *Google* atau *Google Mail Services (GMS)* dan kedua adalah yang benar-benar bebas distribusinya tanpa dukungan langsung *Google* atau dikenal sebagai *Open Handset Distribution (OHD)*.

Telepon yang pertama memakai sistem operasi Android adalah *HTC Dream*, yang dirilis pada 22 Oktober 2008. Pada penghujung tahun 2010 diperkirakan hampir semua perusahaan seluler di dunia menggunakan Android sebagai *operating system*. Adapun versi-versi Android yang pernah dirilis adalah sebagai berikut :

1. Android versi 1.0 & 1.1 (Astro-Alpha & Bender-Beta)

Platform pertama Android dirilis pada september 2008 disini lah awal urutan QS Android bermula lewat *smartphone* pertama yaitu *HTC Dream*.

2. Android versi 1.5 (*Cupcake*)

Dirilis pada tanggal 30 April 2009 dengan berbagai fitur di sebuah perangkat *smartphone* untuk menggantikan *featured phone* pada saat itu.

3. Android versi 1.6 (*Donut*)

Dirilis pada 15 September 2009 beberapa bulan setelah dirilisnya versi sebelumnya. Android ini pun memberikan dukungan layar yang lebih besar.

4. Android versi 2.0 / 2.1 (*Eclair*)

Dirilis pada 3 Desember 2009 kembali diluncurkan Android dengan dukungan fitur Bluetooth hingga fitur kamera yang mulai menjadi nilai jual *smartphone* kala itu. Android ini digunakan pada perangkat seperti ***HTC Nexus One***.

5. Android versi 2.2 (*Froyo: Frozen Yoghurt*)

Dirilis pertama kali pada tanggal 20 Mei 2010, memberikan peningkatan pada kecepatan kerja, fitur USB tethering, WiFi hotspot, serta fitur keamanan.

6. Android versi 2.3 (*Gingerbread*)

Diluncurkan pada Desember 2010, memberikan kelebihan terutama pada tampilan tatap muka alias *user interface* yang digunakan pada tipe Android yang satu ini.

7. Android versi 3.0 (*Honeycomb*)

Diluncurkan pada 10 Mei 2010 sebagai revisi dari versi Android sebelumnya, untuk mendukung Samsung yang merilis perangkat *table Samsung Galaxy Tab Series*.

8. Android versi 4.0 (*Ice Cream Sandwich*)

Dirilis pada 19 Oktober 2011 silam dan masih tetap menggunakan nama *dessert*. Android ini mampu memberikan banyak pembaruan. Mulai dari animasi yang semakin halus, sederhana, dan mudah digunakan.

9. Android versi 4.1 dan 4.3 (*Jelly Beans*)

Sistem operasi Android ini dirilis pada Juni 2012 dengan membawa sejumlah peningkatan terutama di sektor pengolahan grafis dimana bisa memberikan peningkatan fungsi pada *userinterface* dan **teknologi Vsync** Yang digunakannya.

10. Android versi 4.4 (*KitKat*)

Dirilis pada Oktober 2013 jenis Android terfavorit karena mendukung hampir seluruh perangkat *smartphone* di dunia. memberikan optimalisasi yang baik, termasuk pada perangkat *smartphone* yang memiliki spesifikasi kurang mumpuni alias cukup rendah saat itu.

11. Android versi 5.0 dan 5.1 (*Lollipop*)

Dirilis pada Juni 2014 versi Android ini sudah mendukung arsitektur 64-bit yang sudah memungkinkan penggunaan RAM di atas 3GB. Salah satunya **Asus Zenfone 2** yang sudah mengusung RAM 4GB saat itu.

12. Android versi 6.0 (*Marshmallow*)

Sistem operasi ini pertama kali diperkenalkan pada Mei 2015 dan mulai dirilis pada Oktober 2015 silam. Nama OS Android ini secara jelas memberikan peningkatan pada sistem keamanan dengan sistem *fingerprint sensor* sebagai sistem keamanan biometrik yang digunakan.

13. Android versi 7.0 dan 7.1 (*Nougat*)

Pertama kali diperkenalkan pada Juni 2016 dengan menampilkan ikon robot Android dengan batangan Nougat. Versi Android ini mengalami perubahan dari segi tampilan antarmuka. Selain itu, ada juga fitur *splitscreen* untuk membagi tampilan layar untuk dua aplikasi sekaligus.

14. Android versi 8.0 dan 8.1 (*Oreo*)

Dirilis mulai Agustus 2017 sistem operasi ini menawarkan *project treble* yang mampu mempercepat dalam proses pembaruan pada *smartphone*.

15. Android versi 9.0 (*Pie*)

Diperkenalkan pada Agustus 2018 memberikan navigasi berupa *gesture* yang menggantikan tombol fisik *Home*, *Back*, dan *Recent Apps*. Fitur lainnya yang cukup berguna adalah sistem notifikasi, pengatur kecerahan, hingga sistem *screenshoot* terbaru yang lebih mudah.

16. Android versi 10

Dirilis pada 13 Maret 2019 salah satu fitur dalam versi Android terbaru 2019 ini adalah ***Dark Mode*** alias mode gelap yang diklaim mampu meningkatkan performa baterai.

17. Android versi 11

Dirilis pada tahun 2020 dimana terdapat tambahan fitur seperti notifikasi, *screen recorder* tanpa aplikasi tambahan, *picture-in-picture*, hingga mengambil *screenshot* panjang.

18. Android versi 12

Dirilis pada 18 Mei 2021 Android 12 ini menawarkan beragam fitur menarik yang belum ada pada pendahulunya, seperti perubahan desain *material you* yang memudahkan pengguna handphone, peningkatan fitur privasi, dan mode satu tangan.

II.2.4. Visual Studio Code

Visual Studio Code adalah editor source code yang dikembangkan oleh Microsoft untuk Windows, Linux dan MacOS. Ini termasuk dukungan untuk debugging, GIT Control yang disematkan, penyorotan sintaks, penyelesaian kode cerdas, cuplikan, dan kode refactoring. Hal ini juga dapat disesuaikan, sehingga pengguna dapat mengubah tema editor, shortcut keyboard, dan preferensi. Visual Studio Code gratis dan open-source, meskipun unduhan resmi berada di bawah lisensi proprietary.

Kode Visual Studio didasarkan pada Elektron, kerangka kerja yang digunakan untuk menyebarkan aplikasi Node.js untuk desktop yang berjalan pada Blink layout. Meskipun menggunakan kerangka Elektron, Visual Studio Code tidak menggunakan Atom dan menggunakan komponen editor yang sama (diberi kode nama "Monaco") yang digunakan dalam Visual Studio Team Services yang sebelumnya disebut Visual Studio Online (Lardinois, 2015).

II.2.5. Flutter

Flutter merupakan sebuah SDK untuk pengembangan aplikasi mobile yang dikembangkan oleh Google untuk membangun aplikasi yang memiliki kinerja tinggi serta dapat dipublikasi ke platform Android dan iOS dari codebase tunggal. Flutter dapat dengan mudah dipelajari karena menggunakan bahasa pemrograman Dart yang pastinya terasa familiar jika sudah terbiasa menggunakan bahasa pemrograman Java atau Javascript. Selain itu Flutter juga menyertakan kerangka reactive-functional, mesin render 2D, widget siap pakai, dan tools untuk pengembangan. (CrossTechno, 2019).

II.2.6. Unified Modelling Language (UML)

Pemodelan perangkat lunak bekerja dengan cara yang cukup serupa layaknya seorang arsitek atau insinyur teknik sipil yang akan membuat sebuah bangunan, ia biasanya membuat denah – denah yang menggambarkan bentuk jadi dari bangunan.

Sebagai seorang perancang sistem perangkat lunak juga bertindak dengan cara yang serupa, hanya saja dirancang bukan bangunan, melainkan sistem perangkat lunak. Mengambarkan komponen sistem perangkat lunak dalam bentuk geometri tertentu dalam aplikasi. (Adi Nugroho;2014:6)

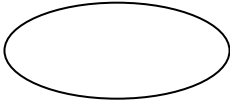
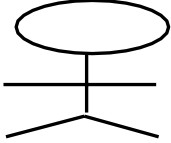
II.2.6.1. Diagram UML

1. Use Case Diagram

Yaitu menggambarkan fungsional yang diharapkan dari sebuah sistem, yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng-*create* sebuah daftar belanja, dan sebagainya. Seorang / sebuah

aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. (Sri Dharwiyanti dan Roma Satria Wahono; 2003: 4). Adapun simbol *Use Case Diagram* dapat dilihat pada tabel berikut : Tabel II.1. *Use Case Diagram*

Tabel II.1. *Use Case Diagram*

No	Simbol	Deskripsi
1.	<i>Use Case</i> 	Fungsi yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.
2.	Aktor /actor 	Adalah orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri.
3.	Asosiasi / association 	Komunikasi antara aktor dan use case yang berpartisipasi pada use case.
4.	<<Include>> 	Merupakan di dalam use case lain (required) atau pemanggilan use case lain, contohnya adalah pemanggilan sebuah program.
5.	<<Extends>> 	Merupakan perluasan dari use case lain jika kondisi atau syarat.

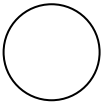
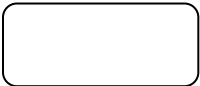
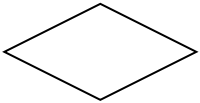
Sumber : Ade Hendini (2016), “pemodelan UML Sistem Informasi Monitoring Penjualan Dan Stok Barang”.

2. Activity Diagram

Merupakan *sate diagram* khusus, dimana sebagian besar *sate action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *activity diagram* tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem), tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Adapun simbol *activity diagram* yang ditunjukkan pada gambar di bawah ini :

Tabel II.2. Activity Diagram

Tabel II.2. Activity Diagram

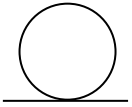
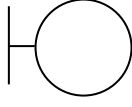
No	Simbol	Deskripsi
	Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
2.	Aktivitas 	Aktivitas dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
3.	Percabangan/ <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.	Penggabungan 	Asosiasi, penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.

Suber : Ade Hendini (2016), “pemodelan UML Sistem Informasi Monitoring Penjualan Dan Stok Barang”.

3. *Sequence* Diagram

Menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence* diagram terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait). Adapun simbol *sequence* diagram yang ditunjukkan pada gambar di bawah ini : Tabel II.3. *Sequence* Diagram

Tabel II.3. *Sequence Diagram*

No	Simbol	Deskripsi
1.	Entity Class 	Merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang menggambarkan hubungan yang akan dilakukan
2.	Boundary Class 	Berisi kumpulan kelas yang menjadi <i>interfaces</i> atau interaksi antara satu atau lebih aktor dengan sistem.
3.	Control Class 	Suatu objek digunakan untuk menghubungkan <i>Boundary</i> dengan tabel.
4.	Message 	Simbol mengirim pesan antar <i>class</i> .

Sumber : Ade Hendini (2016), “pemodelan UML Sistem Informasi Monitoring Penjualan Dan Stok Barang”.

4. Class Diagram

Adalah sebuah spesifikasi yang jika diinisiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi). Adapun simbol *class diagram* yang ditunjukkan pada gambar dibawah ini : Tabel II.4. *Multiplicity Class Diagram*

Tabel II.4. Multiplicity Class Diagram

No	Multiplicity	Penjelasan
1	1	<i>Satu dan hanya satu</i>
2	0..*	<i>Boleh tidak ada atau 1 atau lebih</i>
3	1..*	<i>1 atau lebih</i>
4	0..1	<i>Boleh tidak ada, maksimal 1</i>
5	n..n	<i>Batasan antar, contoh 2..4 mempunyai arti minimal 2 maksimal 4</i>

Sumber : Ade Hendini (2016), “pemodelan UML Sistem Informasi Monitoring Penjualan Dan Stok Barang”.