**SKRIPSI**

**APLIKASI BLOOD BANK BERBASIS WEBSITE**

**DI KABUPATEN KAMPAR**

**(STUDI KASUS : PALANG MERAH INDONESIA (PMI))**

****

**NAMA : YUDI FARADILLA**

**NIM : 1855201041**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS PAHLAWAN TUANKU TAMBUSAI**

**RIAU**

**2022**

**LEMBARAN PENGESAHAN**

**Skripsi yang Berjudul:**

**APLIKASI BLOOD BANK BERBASIS WEBSITE**

Disusun Oleh :

**Nama : Yudi Faradilla**

**NIM : 1855201041**

**Program Studi : S1 Teknik Informatika**

Bangkinang, September 2022

Disetujui Oleh :

**Pembimbing I**

**Prof. Dr. H Amir Luthfi**

**Pembimbing II**

**Novi Yona Sidratul Munti, M.Kom.**

**NIP-TT 096 542 170**

Mengetahui :

**Program Studi S1 Teknik Informatika**

**Ketua Prodi,**

**Deddy Gusman, S.Kom., M.T.I.**

**NIP-TT 096 542 160**

**Fakultas Teknik**

**Dekan,**

**Emon Azriadi, S.T., M.Sc.E.**

**NIP-TT. 096 542 194**

**KATA PENGANTAR**

Dengan mengucapkan puji dan syukur kehadirat Allah SWT atas rahmat yang telah dilimpahkan-Nya, sehingga peneliti dapat menyusun dan menyelesaikan Proposal Penelitian ini, yang diajukan untuk melengkapi dan memenuhi salah satu syarat menyelesaikan pendidikan pada program S1 Teknik Informatika Fakultas Teknik Universitas Pahlawan Tuanku Tambusai dengan judul “**APLIKASI BLOOD BANK BERBASIS WEBSITE DI KABUPATEN KAMPAR**”. Dalam menyelesaikan skripsi ini, peneliti merasakan betapa besarnya manfaat bimbinga yang telah diberikan oleh semua pihak terutama yang memberikan masukan-masukan dan data-data sehingga dapat dijadikan suatu pedoman dan landasan bagi penelitian dalam menggali semua permasalahan yang erat kaitnya dengan skripsi ini.

Pada kesempatan ini peneliti ucapkan terima kasih kepada :

1. Prof. Dr. H Amir Luthfi, selaku Rektor Universitas Pahlawan Tuanku Tambusai sekaligus Pembimbing I yang banyak memberikan bimbingan dan pengarahan serta dorongan kepada Peneliti dalam menyelesaikan Penelitian ini,
2. Emon Azriadi, S.T., M.Sc.E., selaku Dekan Fakultas Teknik Universitas Pahlawan Tuanku Tambusai,
3. Deddy Gusman, S.Kom., M.T.I., selaku Ketua Program Studi S1 Teknik Informatika Universitas Pahlawan Tuanku Tambusai,
4. Novi Yona Sidratul Munti, S.Kom., M.Kom., selaku Sekretaris Program Studi S1 Teknik Informatika Universitas Pahlawan Tuanku Tambusai sekaligus Pembimbing II yang banyak memberikan bimbingan dan pengarahan serta dorongan kepada Peneliti dalam menyelesaikan Penelitian ini,
5. Dosen serta para Staf Universitas Pahlawan Tuanku Tambusai yang telah memberikan dorongan dan motivasi kepeda Peneliti sehingga dapat menyelesaikan Penelitian ini dengan baik,
6. Dengan rasa hormat dan terima kasih yang tak terhingga kepada Ayahanda, Ibunda, dan seluruh Rekan Rekan Seperjuangan Program Studi S1 Teknik Informatika yang telah bermurah hati dalam membantu menyelesaikan Penelitian ini.

Bangkinang, September 2022

Peneliti

Yudi Faradilla

**DAFTAR ISI**

**LEMBARAN PENGESAHAN ii**

**KATA PENGANTAR iii**

**DAFTAR ISI v**

**DAFTAR TABEL viii**

**DAFTAR GAMBAR ix**

**BAB I PENDAHULUAN 1**

1. Latar Belakang 1
2. Rumusan Masalah 3
3. Tujuan Penelitian 3
4. Manfaat Perancangan 3
5. Batasan Masalah 4

**BAB II LANDASAN TEORI 5**

1. Tinjauan Pustaka 5
2. Kerangka Teori 6
3. Golongan Darah 6
4. Donor Darah 7
5. Perancangan Sistem 11
6. Sistem Informasi 12
7. Rekayasa Perangkat Lunak (RPL) 13
8. *Website* 17
9. Internet 19
10. XAMPP 20
11. *Database* 20
12. MySQL 22
13. PHP (*Hypertext Preprocessor*) 22
14. *Hypertext Markup Language* (HTML) 24
15. Laravel 25
16. CSS (*Cascading Style Sheets*) 27
17. *Bootstrap* 28
18. *Javascript* 29
19. Teknik Pengumpulan Data 30
20. UML (*Unified Modeling Language*) 31
21. Figma 37
22. Metode *Waterfall* 38
23. Metode *Blackbox* 38
24. Kerangka Konsep 38

**BAB III METODE PENELITIAN 41**

1. Desain Penelitian 41
2. Lokasi Penelitian 42
3. Sampel 42
4. Alat Penelitian 43
5. Metode Pengumpulan Data 44
6. Perancangan Proses 44

**BAB IV ANALISIS DAN IMPLEMENTASI 46**

1. Analisis Perancangan Sistem 46
2. Analisa Kebutuhan Fungsional Sistem 46
3. Identifikasi Kasus Pengguna Bisnis 47
4. *Usecase Diagram* 48
5. *Class Diagram* 63
6. *Activity Diagram* 64
7. Struktur Aplikasi yang Digunakan 69
8. Rancangan 70
9. *Entity Relationship Diagram* (ERD) 71
10. Perancangan *Database* 71
11. Implementasi 75
12. Implentasi *Database* 76
13. Implementasi Sistem 78
14. Pengujian Sistem 84
15. *Black Box Testing* 84

**BAB V KESIMPULAN DAN SARAN 88**

1. Kesimpulan 88
2. Saran 88

**DAFTAR PUSTAKA 90**

**DAFTAR TABEL**

**Tabel 2.1** *Use Case Diagram* 32

**Tabel 2.2** *Activity Diagram* 33

**Tabel 2.3** *Class Diagram* 35

**Tabel 2.4** *Entity Relationship Diagram* 37

**Tabel 3.1** *Process Business* 45

**Tabel 4.1** Deskripsi *Usecase Diagram* 50

**Tabel 4.2** Skenario *Usecase* Registrasi 52

**Tabel 4.3** Skenario *Usecase Login* 53

**Tabel 4.4** Skenario *Usecase* Lihat dan Pengambilan *Stock* Darah 54

**Tabel 4.5** Skenario *Usecase* Data Pendonor 55

**Tabel 4.6** Skenario *Usecase* Pengeluaran 58

**Tabel 4.7** Skenario *Usecase* Formulir Pendonor 59

**Tabel 4.8** Skenario *Usecase History* Pendonor 60

**Tabel 4.9** Skenario *Usecase Profile* Pendonor 61

**Tabel 4.10** Perancangan tabel *users* 72

**Tabel 4.11** Perancangan tabel *profiles* 72

**Tabel 4.12** Perancangan tabel *formulirs* 73

**Tabel 4.13** *Black Box Testing* 85

**DAFTAR GAMBAR**

**Gambar 2.1** Tujuan RPL 14

**Gambar 2.2** Ruang Lingkup RPL 15

**Gambar 2.3** Kerangka Konsep 39

**Gambar 3.1** Model *Waterfall* 41

**Gambar 4.1** *Usecase Diagram* Aplikasi *Bloodbank* 50

**Gambar 4.2** *Class Diagram* 62

**Gambar 4.3** *Activity Diagram* Registrasi 65

**Gambar 4.4** *Activity Diagram Login* 65

**Gambar 4.5** *Activity Diagram Dashboard Admin* 66

**Gambar 4.6** *Activity Diagram* Data Pendonor 66

**Gambar 4.7** *Activity Diagram* Pengeluaran 67

**Gambar 4.8** *Activity Diagram* Formulir Donor 67

**Gambar 4.9** *Activity Diagram History* 68

**Gambar 4.10** *Activity Diagram Profile* 68

**Gambar 4.11** *Entity Relationship Diagram* 71

**Gambar 4.12** *Database project* 76

**Gambar 4.13** Struktur Tabel *Users* 77

**Gambar 4.14** Struktur Tabel *Profiles* 77

**Gambar 4.15** Struktur Tabel Formulirs 78

**Gambar 4.16** Tampilan *Homepage* 79

**Gambar 4.17** Tampilan *Aboutpage* 79

**Gambar 4.18** Tampilan Syarat Donor 80

**Gambar 4.19** Tampilan *Registrationpage* 80

**Gambar 4.20** Tampilan *Loginpage* 81

**Gambar 4.21** Tampilan *Dashboard* 81

**Gambar 4.22** Tampilan *Dashboard Admin* 82

**Gambar 4.23** Tampilan Formulir Donor 82

**Gambar 4.24** Tampilan *Historypage* 83

**Gambar 4.25** Tampilan *Profilepage* 83

**Gambar 4.26** Tampilan Data Pendonor 84

**Gambar 4.27** Tampilan Pengeluaran 84

**BAB 1**

**PENDAHULUAN**

1. **Latar Belakang**

Berdasarkan fakta di lapangan, bahwa Palang Merah Indonesia (PMI) Bangkinang Kota memiliki permasalahan yaitu bagian Unit Donor Darah masih menggunakan pengolahan data Pendonor secara manual sehingga membuat berkurangnya kualitas pelayanan kepada Masyarakat. Sekarang Sistem Informasi telah berkembang seiring dengan perkembangan teknologi informasi yang sangat cepat dan terbukti berperan dalam berbagai kegiatan. Salah satu kegiatan yang dapat menggunakan sistem infomasi adalah donor darah. Donor darah merupakan suatu kegiatan yang dilakukan oleh seseorang secara sukarela untuk diambil darahnya. Donor darah di Indonesia diatur oleh Peraturan Pemerintah No. 2/2011 tentang pelayanan donor darah yang diatur oleh Palang Merah Indonesia (PMI) sebagai tujuan sosial dan kemanusiaan. Donor darah di bawah pengawasan PMI juga dijamin UU No. 36/2009 tentang Kesehatan, bahwa pemerintah bertanggung jawab atas pelaksanaan pelayanan donor darah yang aman, mudah diakses, dan sesuai dengan kebutuhan masyarakat.

Darah merupakan bagian yang sangat penting dalam tubuh manusia begitu juga dalam hal penggolongan darah manusia yaitu terdapat empat golongan darah manusia yang umum dikenal dan merupakan penggolongan darah yang penting yaitu golongan darah A, B, AB dan O. dalam proses transfusi darah dari satu orang ke orang lain, pengenalan golongan darah harus dilakukan untuk menghindari hal-hal yang tidak diinginkan. Pendonoran darah dari pendonor ke penerima harus diselesaikan jenis golongan darahnya. Kesalahan dalam pengenalan golongan darah akan dapat membahayakan nyawa penerima karena terjadi pembekuan darah akibat bertemunya antigen yang berbeda (Dhimas Bayususetyo, Rukun Santoso, 2017).

*World Health Organization* (WHO) memperkirakan bahwa setidaknya perlu 1% dari total penduduk untuk menyumbangkan darahnya dalam memenuhi kebutuhan minimum darah di suatu negara. Secara global, 70 negara memiliki tingkat pendonor darah kurang dari tingkat optimum, yaitu 10/1000 penduduk. Benua Afrika hanya berhasil mengumpulkan darah untuk memenuhi 41% dari permintaan pada tahun 2006. Negara membutuhkan 36.000 unit darah setiap tahunnya. Arsip menunjukkan bahwa 23.275 unit darah dikumpulkan pada tahun 2009 diikuti dengan penurunan untuk 20.401 unit yang dikumpulkan dalam 2010 dan 16.562 unit yang dikumpulkan pada tahun 2011 (Kemenkes RI, 2014).

Ketersediaan darah untuk donor secara ideal adalah 2,5% dari jumlah penduduk. Sehingga jika jumlah penduduk di Indonesia sebesar 247.837.073 jiwa, maka idealnya dibutuhkan darah sebanyak 4.956.741 kantong darah. Akan tetapi pada tahun 2013 lalu jumlah darah yang terkumpul dari donor sebanyak 2.480.352 kantong darah. Akibatnya rumah sakit masih sering mengalami kesulitan dalam memenuhi kebutuhan transfusi darah melebihi donor darah di masa sekarang. Tingkat kesediaan yang tinggi untuk mendonorkan darah harus dianggap sebagai kesempatan bagi penggerak inisiatif masyarakat di masa mendatang (Kemenkes RI, 2014).

Berdasarkan gambaran diatas maka peneliti merancang sistem informasi *website* dalam penelitian yang berjudul “**APLIKASI BLOOD BANK BERBASIS WEBSITE DI KABUPATEN KAMPAR**”.

1. **Rumusan Masalah**

Berdasarkan uraian latar belakang, maka rumusan masalah yang dapat disimpulkan adalah:

1. Membuatkan Aplikasi Sistem Informasi berbasis *website* untuk mengatasi permasalahan yang terjadi pada Palang Merah Indonesia (PMI) Kabupaten Kampar dalam pengolahan dan pencarian data Pendonor,
2. Aplikasi *Bloodbank* berbasis *website* dapat mempermudah masyarakat dalam melakukan pencarian informasi di Palang Merah Indonesia (PMI) Kabupaten Kampar.
3. **Tujuan Penelitian**

Secara umum tujuan dari penelitian ini sebagai berikut:

1. Mempermudah Unit Donor Darah di Palang Merah Indonesia (PMI) Bangkinang Kota dalam melakukan pengolahan data Pendonor,
2. Merancang dan membangun *website* Pencarian Golongan Darah menggunakan *framework* Laravel.
3. **Manfaat Perancangan**

Adapun manfaat dari penelitian ini yaitu:

1. Sistem Informasi akan memudahkan Unit Donor Darah Palang Merah Indonesia (PMI) Bangkinang Kota mencari data Pendonor saat dibutuhkan,
2. Penulis dapat menerapkan ilmu yang diperoleh di bangku kuliah dan menambah pengetuhan tentang pembuatan sistem informasi menggukan *framework* Laravel,
3. Dapat menambah referensi mengenai sistem informasi mengunakan *framework* Laravel bagi mahasiswa yang akan melakukan penelitian di masa yang akan datang serta sebagai tolak ukur sejauh mana pemahaman mahasiswa tentang materi perkuliahan yang di pelajari.
4. **Batasan Masalah**

Sesuai dengan latar belakang dan perumusan masalah yang telah diuraikan diatas maka batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Bahasa pemograman yang digunakan adalah *framework* Laravel,
2. *Web Server* menggunakan Apache yang didukung PHP dilengkapi dengan *PhpMyAdmin* sebagai *Control Center* MySQL,
3. *Website* ini bisa di akses secara *Localhost*.

**BAB II**

**LANDASAN TEORI**

1. **Tinjauan Pustaka**

Hasil penelitian terdahulu digunakan untuk mengetahui persaman dan perbedaan dari beberapa penelitian yang ada, Berikut adalah beberapa penelitian terdahulu yang digunakan dalam penelitian ini:

1. Penelitian tentang Sistem Informasi Bank Darah dengan Location Based Service untuk Meningkatkan Efisiensi Pencarian Golongan Darah dibuat Oleh (Begawan Raka Sakti, Wina Witanti, 2021), yang membahas tentang salah satu teknologi yang memberikan kemudahan bagi penggunanya, dengan adanya fungsi untuk menentukan suatu lokasi dan dapat memberi informasi lokasi terdekat dari penggunanya. Simulasi Pencarian Darah ini diharapkan dapat membantu orang yang membutuhkan darah tanpa harus mendatangi langsung ke Rumah Sakit yang belum tentu terdapat jenis darah yang di inginkan khususnya untuk wilayah Sidoarjo. Simulasi Pencarian Darah Berbasis Web Menggunakan Framework Bootstrap dan Codeigniter dengan menggunakan PHP, CSS dan MySQL.Aplikasi yang dikembangkan untuk penelitian ini menggunakan *Hypertext Prepocessor* (PHP) yang merupakan bahasa pemrograman umum dan juga untuk pengembangan *web*. Selain itu juga digunakan *CodeIgniter* (*Framework* PHP), *Bootstrap* (*Framework* CSS) untuk membuat *web* dinamis dan responsif dengan cepat tanpa harus membuat dari awal.
2. Penelitian tentang Sistem Informasi Pelayanan Donor Darah Berbasis Website (Studi Kasus : PMI Tasik Malaya) dibuat Oleh (R. A. Gustaman et al., 2016) yang membahas bahwa Palang Merah Indonesia (PMI) Tasikmalaya memiliki permasalahan yang sama dengan cabang PMI lainnya, yaitu pada bagian Unit Donor Darah masih menggunakan pengolahan data secara manual dan integrasi data antara PMI terhadap Bank Darah Rumah Sakit (BDRS) masih menjadi faktor berkurangnya kualitas pelayanan kepada pasien. Teknologi Informasi menjadi salah satu solusi dalam melakukan manajemen data antara PMI dan BDRS. Metode dalam penelitian ini adalah *what*, *who*, *where*, *when*, *why*, *how* (5W1H) dengan metode pengembangan aplikasi adalah *Extreme Programming* (XP) hanya pada tahapan *Planning* dan tahapan design. SMS *gateway* dilibatkan pada sistem pelayanan donor darah, sehingga memudahkan pasien mendapatkan informasi data darah. Hasil dari penelitian ini adalah aplikasi *Transaction Processing System* (TPS) dan SMS *gateway* berjalan sesuai dengan rancangan, pada tahapan planning dan tahapan design pada metode XP (*Extreme Programming*) melibatkan satu admin dan banyak *user* (pasien). Teknologi SMS *gateway* memudahkan pasien dalam mendapatkan informasi stok darah yang diinginkan.
3. **Kerangka Teori**
4. **Golongan Darah**

Golongan darah merupakan ciri khusus darah dari suatu individu karena adanya perbedaan jenis karbohidrat dan protein pada permukaan membran sel darah merah. Didunia ini sebenarnya dikenal sekitar 46 jenis antigen selain antigen ABO dan Rh (Andriyani *et al*,2015). Sistem ABO yang ditemukan oleh Karl Landsteiner merupakan sistem yang paling penting dalam bank darah dan ilmu kedokteran transfusi, antigen-antigen utamanya disebut A dan B, antibodi utamanya adalah anti-A dan anti-B. Gen-gen yang menentukan ada tidaknya aktivitas A atau B terletak di kromosom 9 (Ronald, 2004).

Dalam sistem golongan darah ABO ini, berlaku asas yang mengatakan bahwa serum seseorang tidak akan mengendapkan sel darah merah orang itu sendiri serta sel darah merah orang lain yang bergolongan sama. Jadi, serum darah dari orang yang bergolongan darah A tidak akan mengaglutinasikan sel darah merah dari orang yang bergolongan darah A. Hal yang sebaliknya juga berlaku untuk serum yang bergolongan darah B. Serum dari orang yang bergolongan darah AB tidak dapat mengendapkan sel darah merah golongan AB, juga tidak dapat mengaglutinasikan sel darah merah golongan A maupun golongan B. Sel darah merah golongan O tidak dapat diaglutinasikan oleh serum dari orang yang bergolongan darah A, B, maupun AB (Sadikin, 2001).

1. **Donor Darah**
2. **Pengertian Donor Darah**

Donor darah adalah proses pengambilan darah dari seseorang secara sukarela untuk disimpan di bank darah untuk kemudian dipakai pada transfusi darah. Transfusi darah adalah proses pemindahan darah dari seseorang yang sehat (donor) ke orang sakit (resipien). Darah yang dipindahkan dapat berupa darah lengkap dan komponen darah. Donor darah biasa dilakukan rutin di pusat donor darah lokal, dan setiap beberapa waktu dilakukan kegiatan donor darah di tempat keramaian, misalanya saja di pusat perbelanjaan, di sekolah, Universitas, di kantor perusahaan besar, ataupun di tempat ibadah. Hal ini dimaksudkan, agar mempemudah dan menarik simpati masyarakat untuk melakukan donor darah, hal ini juga mempermudah para pendonor agar melakukan donor darah, tanpa harus ke pusat donor darah. Adapula mobil darah yang juga dapat digunakan untuk dijadikan tempat menyumbang. Biasanya bank darah memiliki banyak mobil darah (Depkes RI, 2009).

1. **Syarat Donor Darah**

Beberapa syarat yang bertujuan untuk menjamin keselamatan pendonor dan penerima darah menurut (Permenkes RI, 2015) adalah sebagai berikut :

1. Umur 17-60 tahun (usia 17 tahun diperbolehkan menjadi donor bila mendapat izin tertulis dari orang tua),
2. Berat badan minimal 45 kg,
3. Temperatur tubuh berkisar antara 36,6-370C,
4. Tekanan darah baik, yang ditunjukkan dengan systole 110- 160 mmHg dan diastole 70-100 mmHg,
5. Denyut nadi teratur yaitu sekitar 50- 100 kali/menit,
6. Hemoglobin baik pria maupun perempuanminimal 12,5 gram,
7. Bagi penyumbang darah wanita tidak sedang hadi, hamil atau menyusui,
8. Tidak menderita penyakit jantung, hati, ginjal, paru, kencing manis, pendarahan, kejang atau penyakit kulit kronis,
9. Tidak pernah menderita penyakit hepatitis B,
10. Tidak pernah menderita penyakit tuberculosis, sifilis, epilepsy dan sering kejang,
11. Tidak pernah mengalami ketergantungan obat, alkoholisme akut dan kronik,
12. Tidak pernah menderita penyakit kulit pada vena (pembuluh darah balik) yang akan ditusuk,
13. Tidak mempunyai kecenderungan perdarahan atau penyakit darah, misalnya defisiensi G6PD, thalasemia dan polibetemiavera,
14. Tidak mengidap penyakiy HIV/AIDS (homoseks, morfinis, berganti-ganti pasangan seks, pemakai jarum suntik tidak steril).
15. **Manfaat Donor Darah**

Donor darah akan membantu menurunkan resiko terkena serangan jantung dan masalah jantung lainnya. Penelitian menunjukkan, mendonorkan darah akan mengurangi kelebihan zat besi dalam tubuh. Walaupun masih perlu penelitian lagi untuk memastikannya, kelebihan zat besi diduga berperan menimbulkan kelainan pada jantung. Kelebihan itu akan membuat kolesterol jahat (LDL) membentuk antikolesterol (plak lemak yang akan mneyumbat pembuluh darah). Menurunnya angka masalah penyakit jantung terutama terlihat pada para pendonor yang tidak merokok (A. B. Gustaman & Suji., 2013).

Manfaat mendonorkan darah secara rutin setiap tiga bulan sekali maka menyebabkan tubuh akan terpacu untuk memproduksi sel-sel darah merah baru, sedangkan fungsi sel-sel darah merah adalah untuk oksigenisasi dan mengangkut sari-sari makanan. Dengan demikian fungsi darah menjadi lebih baik sehingga donor menjadi sehat. Selain itu, kesehatan pendonor akan selalu terpantau karena setiap kali donor dilakukan pemeriksaan kesehatan dan pemeriksaan uji saring darah terhadap infeksi yang dapat ditularkan lewat darah. Manfaat lainnya dari mendonorkan darah adalah mendapatkan kesehatan psikologis karena menyumbangkan hal yang tidak ternilai harganya kepada yang membutuhkan akan membuat kita merasakan kepuasan psikologis.

Sebuah penelitian menemukan, orang usia lanjut yang rutin menjadi pendonor darah akan merasakan tetap berenergi dan bugar (A. B. Gustaman & Suji., 2013).

1. **Efek samping Donor Darah**

Donor darah juga memiliki efek samping bagi tubuh menurut (Prayitno, 2005), yaitu :

1. Terlalu sering melakukan donor darah akan menyebabkan tulang keropos. Sebagian orang menganggap bahwa melakukan donor darah akan menyebabkan tulang menjadi lebih cepat keropos. Alasan ini didasari dengan terlalu sering donor darah, akan menyebabkan tulang belakang bekerja lebih ekstra dan menyebabkan osteoporosis. Efek samping ini tentu saja tidak benar. Jika donor darah saat kondisi tubuh sehat, produksi darah tidak akan terganggu.
2. Donor darah menyebabkan pusing dan muntah. Jika pusing berlebih setelah melakukan donor darah, mungkin saat melakukan donor darah saat tekanan darah di bawah normal atau hipotensi. Perasaan pusing ringan dan juga mual merupakan hal normal pada sebagian orang.
3. Setelah donor darah harus istirahat penuh selama sehari. Setelah selesai melakukan donor darah, tidak harus beristirahat selama sehari penuh untuk mengembalikan tenaga agar normal. Dengan memenuhi kebutuhan nutrisi dan asupan cairan yang cukup, keadaan akan pulih seperti semula.
4. Donor darah akan memberikan efek pada tubuh menjadi mudah gemuk. Hal ini juga tidak dibenarkan, walaupun ada beberapa orang yang makan dengan porsi sedikit, maka selanjutnya akan lebih banyak makan setika selesai melakukan donor darah.
5. **Perancangan Sistem**
6. **Perancangan**

Perancangan adalah penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari berbagai elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi (Trianto & Yulianeu 2018).

Menurut (Robert J. Verzello/John Reuter III, 1982), perancangan adalah tahap setelah analisis dari siklus pengembang system Pendefinisian dari kebutuhan kebutuhan fungsional dan persiapan untuk rancang bangun implementasi : “ Mengembarkan bagaimana suatu sistem dibentuk”.

1. **Sistem**

Secara garis besar sistem merupakan suatu kumpulan komponen dan elemen yang saling terintegrasi, komponen yang terorganisir dan bekerja sama dalam mewujudkan suatu tujuan tertentu.

Menurut (Sri Mulyani, 2016) sistem bisa diartikan sebagai sekumpulan sub sistem, komponen yang saling bekerja sama dengan tujuan yang sama untuk menghasilkan output yang sudah ditentukan sebelumnya.

1. **Sistem informasi**

Menurut (Bonnie Soeherman, 2008), Sistem Informasi merupakan serangkaian komponen berupa manusia, prosedur, data, dan teknologi (seperti komputer) yang digunakan untuk melakukan sebuah proses untuk pengambilan keputusan guna penunjang keberhasilan bagi setiap organisasi (dalam pencapaian tujuan).

(Mulyanto, 2008) Menyatakan “Sistem Informasi merupakan suatu komponen yang terdiri dari manusia, teknologi informasi, dan prosedur kerja yang memproses, menyimpan, menganalisis, dan menyebarkan informasi untuk mencapai suatu tujuan.”

1. **Rekayasa Perangkat Lunak (RPL)**

Rekayasa perangkat lunak adalah suatu perintah yang digunakan untuk memproses sebuah produksi perangkat lunak dimulai dari tahap awal pembuatan system hingga selesainya sebuah system yang akan digunakan. Menurut (Sidratul munti, 2019) karakteristik perangkat lunak lebih dikenal sebagai elemen logic daripada fisik, oleh karena itu perangkat lunak memiliki karakteristik yang berbeda dari perangkat keras:

1. Perangkat lunak dikembangkan atau di rekayasa, jadi tiddak diproduksi dalam pengertian klasik,
2. Merupakan produk yang unik (tidak ada seri produksi),
3. Perangkat lunak tidak pernah akan rusak karena selalu diperbarui,
4. Tidak terlihat,
5. Perangkat lunak pada umumnya dibangun sesuai keinginan, jadi tidak dibentuk,
6. *Fleksibel,* sehingga mudah dimodifikasi,
7. Dihubungkan dengan sistem komputer.

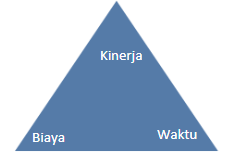
Rekayasa perangkat lunak mempunyai beberapa tujuan yaitu sebagai berikut:

1. Memperoleh biaya produksi perangkat lunak yang rendah,
2. Menghasilkan perangkat lunak yang kinerjanya tinggi, handal dan tepat waktu,
3. Menghasilkan perangkat lunak yang dapat bekerja pada berbagai jenis *platfrom*,
4. Menghasilkan perangkat lunak yang biaya perawatannya rendah,
5. Mampu melaksanakan analisis kebutuhan, perangcangan, implementasi dan pengujian perangkat lunak dengan pendekatan terstruktur,
6. Mampu Menyusun dokumentasi pengembangan perangkat lunak.

Istilah Rekayasa Perangkat Lunak secara umum disepakati sebagai terjemahan dari istilah *Software Engineering*. Istilah *Software Engineering* dipopulerkan tahun 1968 pada *Software Engineering Coference*. Yang diselenggarakan oleh NATO. Sebagian orang mengartikan Rekayasa Perangkat Lunak hanya sebatas pada bagaimana membuat program komputer. Padahal ada perbedaan yang mendasar antara perangkat lunak (*software*) dan program komputer.

* + - 1. **Tujuan Rekayasa Perangkat Lunak (RPL)**

Bidang rekayasa akan selalu berusaha menghasilkan output yang kinerjanya tinggi, biaya rendah dan waktu penyelesaian yang tepat. Tujuan RPL dapat ditunjukkan pada Gambar 2.1.



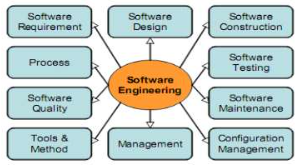
**Gambar 2.1** Tujuan RPL

Sumber : (Findawati, 2018).

Secara lebih khusus kita dapat menyatakan tujuan RPL adalah :

* 1. Memperoleh biaya produksi perangkat lunak yang rendah,
  2. Menghasilkan perangkat lunak yang kinerjanya tinggi, andal dan tepat waktu,
  3. Menghasilkan perangkat lunak yang dapat bekerja pada berbagai jenis *platform*,
  4. Menghasilkan perangkat lunak yang biaya perawatanya rendah.
     + 1. **Ruang Lingkup Rekayasa Perangkat Lunak (RPL)**

Berdasarkan definisi dari para ahli yang telah dibahas sebelumnya, maka ruang lingkup RPL dapat ditunjukkan pada gambar 2.2.



**Gambar 2.2** Ruang Lingkup RPL

Sumber : (Findawati, 2018).

1. *Software* *Requirment* Merupakan kegiatan yang dilakukan untuk mengidetifikasi dan menganalisis kebutuhan perangkat lunak. Hasil akhir tahapan ini spesifikasi dan model perangkat lunak.
2. *Software Design* adalah tahapan perancangan arsitektur, komponen, antar muka, dan karakteristik lain dari perangkat lunak.
3. *Construction* berhubungan dengna detail pengembangan perangkat lunak, termasuk algoritma, pengkodean, pengujian dan pencarian kesalahan.
4. *Software Testing* meliputi pengujian pada keseluruhan perilaku perangkat lunak.
5. *Software Maintenance* mencakup upaya-upaya perawatan Ketika perangkat lunak telah dioperasikan.
6. *Software Configuration* managemen behubungan dengan usaha perubahan configuration perangkat lunak untuk memenuhi kebutuhan tertentu.
7. *Software Engineering* managemen berkaitan dengan pengelolahan dan pengukuran Rekayasa Perangkat Lunak, termasuk perancangan proyek perangkat lunak.
8. *Software Engineering Tools and Methods* mencakup kajian teoritis tentang alat bantu dan metode Rekayasa Perangkat Lunak.
9. *Software Engineering Process* Berhubungan dengan defenisi, impelementasi pengukuran, pengelolahan, perubahan, dan perbaikan proses Rekayasa Perangkat Lunak.
10. *Software Quality* menitikberatkan pada kualitas dan daur hidup Perangkat Lunak.
11. ***Website***

*Web* atau lengkapya WWW (*World Wide Web*) adalah sebuah koleksi keterhubungan dokumen-dokumen yang disimpan di internet dan diakses menggunakan *protocol* (*hyper text transfer protocol*).

*Web* adalah halaman informasi yang disediakan melalui jalur internet dan memiliki fasilitas hiperteks untuk menampilkan data berupa teks, gambar, suara, animasi, dan data multimedia lain nya sehingga bisa diakses diseluruh dunia selama terkoneksi dengan jaringan internet.

*Web* dapat diartkan sekumpulan halaman situs yang saling berhubungan menampilkan berbagai macam informasi maupun data berupa teks, gambar diam ataupun bergerak, data animasi, suara, video maupun gabungan dari semuanya yang bisa diakses dimana saja kapan saja diseluruh dunia. (Rachman Andi et al., 2017). Dimana *Web* dikategorikan menjadi dua, yaitu;

1. ***Web* Statis**

*Web* yang menampilkan informasi-informasi bersifat statis (tetap). Disebut statis karena pengguna tidak dapat berinteraksi dengan *web* tersebut. Pengguan hanya dapat melihat isi dokumen pada halaman web dan apabila diklik akan berpindah kehalaman web yang lain. Interaksi pengguna terbatas hanya melihat informasi yang ditampilkan, tetapi tidak bisa mengolah informasi yang dihasilkan. Biasanya merupakan HTML yang ditulis pada editor teks dan disimpan dalam bentuk, HTML, atau HTM.

1. ***Web* Dinamis**

*Web* yang menampilkan informasi serta dapat berinteraksi dengan pengguna *Web* yang dinamis memungkinkan pengguna untuk berinteraksi dengan form sehingga dapat mengolah informasi yang ditampilkan. Bersifat teraktif, tidak kaku dan terlihat lebih indah.

Sebelum mulai membuat desain tampilan *Web*, ada baiknya memahami lebih dahulu fungsi sebuah *Web,* sehingga desain yang dibuat disesuaikan dengan fungsi *Web* tersebut, secara umum *Web* memiliki fungsi, yaitu:

1. Fungsi Komunikasi

Sebagian besar *web* mempunyai fungsi komunikasi. Beberapa fasilitas yang memberikan fungsi komunikasi ini, seperti : *web base email,* halaman *form contact¸chatting* dan lain-lain.

1. Fungsi Informasi

*Web* mempunyai fungsi informasi seperti, *news, profile company, library,* referensi, dan lain-lain.

1. Fungsi *Entertainment*

*Web* Mempunyai fungsi hiburan. Beberapa contoh *web* dengan fungsi ini, misalnya *web-web* yang menyediakan *online game, online music, online movie,* dan sebagainya.

1. Fungsi Transaksi

Sebuah *Web* dapat dijadikan saran untuk melakukan transaksi bisnis, seperti: *online order,* pembayaran menggunakan kartu kredit, dan lain-lain.

1. **Internet**

Internet digunakan sebagai media penghubung pertukaran informasi melalui jaringan online yang menghubungkan website-website yang ada. internet juga dapat dikategorikan sebagai penghubung antar media komputer di seluruh dunia.

Menurut (Priyanto Hidayatullah, Jauhari Khairul Kawistara, 2017) “internet adalah jaringan global yang menghubungkan komputer-komputer seluruh dunia, dengan internet sebuah komputer bisa mengakses data yang terdapat pada komputer lain di benua yang berbeda”.

Menurut (Sofia, Budhi Prianto, 2010) mengemukakan bahwa “internet merupakan kumpulan dari jutaan komputer di seluruh dunia yang terkoneksi satu sama lain”. Sedangkan menurut (Zaki, SmitDev Community, 2008) menyatakan bahwa “internet adalah jaringan komputer skala dunia yang memungkinkan orang-orang untuk saling berhubungan menggunakan berbagai layanan, seperti email, chat online, transfer file, dan halaman web (WWW)”. Internet adalah kelompok atau kumpulan jutaan komputer. Penggunaan internet memungkinkan kita untuk mendapatkan informasi dari komputer yang ada di dalam kelompok tersebut dengan asumsi bahwa pemilik komputer memberikan izin akses. Untuk mendapatkan sejumlah informasi, sekumpulan protokol harus digunakan, yaitu sekumpulan aturan yang menetapkan bagaimana suatu informasi dapat dikirim dan diterima (Simarmata, 2010). Internet dapat diartikan sebagai alat penghubung sistem sistem atau media penghubung para pengguna untuk mengakses informasi yang terdapat dalam jaringan *World Wide Web* (WWW) diseluruh dunia.

1. **XAMPP**

Menurut (Wahana Komputer, 2014), XAMPP merupakan singkatan dari X (empat sistem operasi apa pun), *Apache*, PHP, dan *Perl*. XAMPP adalah *tools* yang menyediakan peket perangkat lunak dalam satu buah paket. Dalam paket XAMPP sudah terdapat *Apache* (*web server*), MySQL (*database*), PHP (*serve side scripting*), Perl, FTP server, PhpMyAdmin dan berbagai pustaka bantu lainnya.

Program ini tersedia dalam GNU *General Public License* dan bebas, merupakan *web server* yang mudah untuk digunakan yang dapat menampilkan halaman web yang dinamis. Untuk mendapatkanya XAMPP anda dapat mendownload langsung dari web resminya.

1. ***Database***

Menurut (Fathansyah, 2015) “Basis data terdiri dari 2 kata yaitu Basis dan Data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat berserang/berkumpul. Sedangkan data adalah representasi fakta dunia nyata mewakili suatu objek seperti manusia (pegawai, siswa, pembelian pelanggan), barang hewan, peristiwa, konsep, keadaan, dan sebagian yang berwujudkan dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya”.

1. **Definisi *Database***

Menurut (M. Shalahuddin, Rosa A.S, 2014) “basis data merupakan salah satu bagian dalam rekayasa perangkat lunak yang terkomputerisasi dan bertujuan utama memelihara data yang sudah diolah atau media penyimpanan informasi agar dapat diakses dengan mudah dan cepat”.Dapat disimpulkan bahwa basis data bagian dari rekayasa perangkat lunak yang terkomputerisasi sebagai media penyimpanan informasi yang saling berhubungan atau punya relasi untuk penyimpanan data informasi agar dapat diakses dengan mudah dan cepat.

1. ***Structured Quety Laguage* (SQL)**

Menurut (M. Shalahuddin, Rosa A.S, 2014) SQL (*Structured Quety Laguage*) adalah “bahasa yang digunakan untuk mengolola data pada RDBMS (*Relational DBMS*) yang dikembangkan berdasarkan teori aljabar relasional dan kalkulus”.

1. ***PHPmyAdmin***

Menurut (MADCOMS, 2016) “*PhpMyAdmin* adalah sebuah aplikasi *Open Source* yang berfungsi untuk memudahkan manajemen MySQL. Dengan menggunakan *PhpMyAdmin*, dapat membuat *database*, membuat tabel, meng-*insert*, menhapus dan meng-*update* data dengan GUI dan terasa lebih mudah, tanpa perlu mengetikkan perintah SQL secara manual”.

Fungsi dari halaman ini adalah sebagai pengendali *database* MySQL. Karena dengan adanya halaman ini semua hal tersebut dapat dilakukan hanya dengan meng-klik menu fungsi yang ada pada halaman *PhpmyAdmin*.

1. **MySQL**

(Parulian, 2017) menyebutkan bahwa MySQL Merupakan sebuah perangkat lunak dengan sistem manajemen database Sql (*database management system*) atau DBMS yang *multithread*, *multi-user*, penggunaan yang cukup besar yakni sekitar 6 juta di seluruh dunia. MySQL AB dibawah lisensi GNU *General Public License* (GPL) membuat MySQL tersedia sebagai perangkat lunak gratis.

(Parulian, 2017) menyatakan bahwa MySQL adalah sistem manajemen *database* digunakan untuk penyimpanan data dalam tabel terpisah dan berfungsi menempatkan semua data dalam satu ruang yang besar.

Berdasarkan pengertian tersebut maka disimpulkan bahwa MySQL merupakan bahasa komputer ataupun bahasa pemrograman yang difokuskan untuk *database* atau penyimpanan data. Kegunanaan dari MySQL adalah untuk menyimpan data-data dalam kapasitas ruang yang besar. MySQL memiliki banyak keunggulan contohnya seperti *database* yang aman dan tidak memerlukan pembelian dalam menggunakannya.

1. **PHP (*Hypertext Preprocessor*)**

(Tim EMS, 2016) menyatakan PHP merupakan suatu singkatan dari *Hypertext Preprocessor* atau Merupakan bahasa *script* di mana ketika menggunakan PHP maka dapat dibuat *web* dinamis dengan kode PHP yang kemudian ditautkan di antara *script* kode-kode HTML. Hal ini merupakan bahasa markup standar untuk dunia *web*.

Dapat disimpulkan dari beberapa pengertian diatas bahwa PHP merupakan suatu bagian terpenting dalam pembuatan *website* dinamis. Hal ini karena dalam PHP terhadap script yang berisi kode-kode untuk membuat *web*.

PHP merupakan Bahasa pemograman yang banyak digunakan oleh seorang *programmer* kerena Bahasa pemrograman ini sangat mudah dipahami dan mudah dipahami dan mudah dikoneksikan didalam *database.*

PHP juga merupakan Bahasa pemograman yang bersifat *open source* yang bisa didownload gratis. Hingga saat ini PHP versi terakhir adalah versi 8.0.0 dimana bisa didownload pada situs resmi PHP [*http://php.net/downloads.php*](http://php.net/downloads.php).

PHP merupakan perangkat lunak *open source*, yang mana penulisan kode program PHP menyatu dengan HTML yang berjalan pada *server-side*. Hal ini berarti semua *syntax* yang telah dieksekusi saja yang dikirimkan pada sisi *client/browser* tanpa bisa mengetahui kode yang digunakan. PHP merupakan Bahasa pemograman yang sesuai untuk membuat aplikasi *website* dinamis seperti CMS karena memiliki perfoma yang tinggi, mudah dipelajari, *multiplatform,* aman *open source,* serta mudah dikoneksikan dengan berbagai macam *database*.

Dan dengan pengembangan yang dilakukan PHP kini memiliki kinerja yang lebih tinggi dari pada versi-versi sebelumnya, berkemampuan OPP (*Object Oriented Programming), syntax highlighting, array* multidimensi, dan dapat diperluas melalui mekanisme extension, PHP merupakan salah satu Bahasa *server-side* yang didesain khusus untuk perancangan aplikasi web dan tergolong aman digunakan (Sidratul munti, 2019).

Adapun kelebihan bahasa pemrograman PHP dari bahasa pemrograman lain adalah sebagai berikut:

1. Bahasa pemrograman PHP adalah sebuah bahasa *script* yang tidak melakukan sebuah kompilasi dalam penggunaanya.
2. *Web Server* yang mendukung PHP dapat ditemukan dimana-mana dari mulai *apache*, *IIS, Lighttpd, hingga Xitami* dengan konfigurasi yang relatif mudah.
3. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan *developer* yang siap membantu dalam pengembangan.
4. Dalam sisi pemahamanan, PHP adalah bahasa *scripting* yang paling mudah karena memiliki referensi yang banyak.
5. PHP adalah bahasa *open source* yang dapat digunakan di berbagai mesin (*Linux, Unix, Macintosh, Windows*) dan dapat dijalankan secara runtime melalui console serta juga dapat menjalankan perintah-perintah sistem.
6. ***Hypertext Markup Language* (HTML)**

Menurut (Priyanto Hidayatullah, Jauhari Khairul Kawistara, 2017) *Hypertext Markup Language* (HTML) adalah Bahasa *standard* yang digunakan unutk menampilkan halaman *web*. Yang bisa dilakukan dengan HTML yaitu:

1. Mengatur tampilan dari halaman *web* dan isinya,
2. Membuat tabel dalam halaman *web,*
3. Mempublikasikan halaman *web* secara *online*,
4. Menambahkan objek-objek seperti citra, audio, video, animasi, java applet dalam halaman *web*.
5. **Laravel**

Menurut (Aminudin, 2015) Laravel adalah sebuah *Framework* PHP dirilis dibawah lisensi MIT dengan kode sumber yang sudah disediakan oleh Github, sama seperti *framework*-*framework* yang lain, Laravel dibangun dengan konsep MVC (*Model-Controller-View*), kemudian Laravel dilengkapi juga *Command Line Tool* yang bernama “*Artisan*” yang bisa digunakan untuk *packaging bundle* dan *installation bundle* melalui *command prompt*.

Berikut ini beberapa fitur yang dimiliki oleh *framework* Laravel menurut (Aminudin, 2015):

1. *Bundles*

*Bundles* yaitu sebuah fitur dengan *system* pengemasan modular dan berbagai *bundle* telah tersedia untuk digunakan dalam aplikasi Anda.

1. *Eloquent* ORM

*Eloquent* ORM merupakan penerapan PHP lanjutan dari pola “*active record”* menyediakan metode internal untuk mengatasi kendala hubungan antara objek *database*. Pembangun *query* Laravel *Fluent* didukung *Eloquent*.

1. *Application Logic*

*Application Logic* merupakan bagian dari aplikasi yang dikembangkan, baik menggunakan *Controllers* maupun sebagai bagian dari deklarasi *Route*. *Syntax* yang digunakan untuk mendefinisikannya mirip dengan yang digunakan oleh *framework* Sinatra.

1. *Reverse Routing*

*Reverse Routing* mendefinisikan hubungan antara *link* dan *route*, sehingga jika suatu saat ada perubahan pada *route* secara otomatis akan tersambung dengan *link* yang relevan. Ketika *link* yang dibuat dengan menggunakan nama-nama dari *route* yang ada, secara otomatis laravel akan membuat URL yang sesuai.

1. *Restful Controllers*

*Restful Controllers* memberikan sebuah *option* (pilihan) untuk memisahkan logika dalam melayani HTTP *GET* dan permintaan *POST*.

1. *Class Auto Loading*

*Class Auto Loading* menyediakan otomatis *loading* untuk *class-class* PHP, tanpa membutuhkan pemeriksaan manual terhadap jalur masuknya. Fitur ini mencegah loading yang tidak perlu.

1. *View Composers*

*View Composers* adalah kode *unit logical* yang dapat dijalankan ketika sebuah *view* di *load*.

1. *IoC Container*

*IoC Container* memungkinkan untuk objek baru yang dihasilkan dengan mengikuti prinsip control pembalik, dengan pilihan contoh dan referensi dari objek baru sebagai Singletons.

1. *Migrations*

*Migrations* menyediakan versi sistem control untuk skema *database*, sehingga memungkinkan untuk menghubungkan perubahan adalah basis kode aplikasi dan keperluan yang dibutuhkan dalam merubah tata letak *database*. Mempermudah dalam penempatan dan memperbarui aplikasi.

1. *Unit Testing*

*Unit Testing* mempunyai peran penting dalam *framework* Laravel, dimana *Unit Testing* ini mempunyai banyak tes untuk mendeteksi dan mencegah regresi. *Unit Testing* dapat dijalankan melalui fitur “*artisan command-line*”.

1. *Automatic Pagination*

*Automatic Pagination* menyederhanakan tugas dari penerapan halaman, menggantikan penerapan yang manual dengan metode otomatis yang terintegrasi ke Laravel

1. **CSS (*Cascading Style Sheets*)**

Menurut (Priyanto Hidayatullah, Jauhari Khairul Kawistara, 2017) “CSS berfungsi untuk membuat tampilan lebih rapih dan menarik”.

Ada 4 cara untuk memasang kode CSS ke dalam HTML / Halaman *Web* (Priyanto Hidayatullah, Jauhari Khairul Kawistara, 2017) yaitu :

1. *Inline Style Sheet*

Cara ini adalah menuliskan langsung *script* CSS ke dalam *tag* HTMLyang diinginkan. Cara ini sebaiknya hanya digunakan jika ingin memformat suatu elemen satu kali saja.

1. *Internal Style Sheet*

Ini dilakukan dengan cara memasang kode CSS berada pada *file* HTML bersangkutan namun diletakan khusus pada suatu tempat yaitu diantara *tag* <*head*> dan </*head*>.

1. Me-*link* ke *External*

CSS Kode CSS *external* ditulis dalam suatu *file* terpisah yang disimpan dengan ekstensi .css. Kemudian harus memanggil *file* CSS tersebut pada halaman *web* yang memerlukannya.

1. *Import* CSS *file*

Meng-*import* CSS kedalam suatu *file* CSS yang lain mernggunakan *tag import*.

1. ***Bootstrap***

*Bootstrap* merupakan sebuah *Framework* CSS untuk membangun *website* yang menarik agar memudahkan pengembang disebut *Bootstrap*. Sulit untuk mengembangkan dan pemeliharaannya jika tidak ada konsistensinya terhadap aplikasi individual. *Bootstrap* memberikan solusi rapi dan seragam terhadap solusi yang umum, tugas *interface* yang setiap pengembang hadapi.

Menurut (Nahado, 2015) “*Bootstrap* adalah platform untuk membuat *interface website* dan aplikasi berbasis *web*”. *Bootstrap* berisi kode HTML dan CSS yang telah dilengkapi desain untuk tipografi, bentuk, tombol, navigasi dan sebagainya. *Bootstrap* bertujuan untuk meringankan pembuatan dan pengembangan *web*.

Dapat disimpulkan bahwa *bootstrap* merupakan sebuah aplikasi yang dijadikan sebagai alat bantuk untuk mendesain halaman *web* yang menarik.

1. ***Javascript***

*Javascript* menurut (Sunyoto, 2007) adalah bahasa *scripting* yang popular di internet dan dapat bekerja di sebagian besar *browser* popoler seperti *Internet Explorer* (IE), *Mozilla Firefox*, *Netscape* dan *Opera*. Kode *Javascript* dapat disisipkan dalam halaman *web* menggunakan *tag SCRIPT*. Beberapa hal tentang *Javascript*:

1. *Javascript* didesain untuk menambah interaktif suatu *web,*
2. *Javascript* merupakan sebuah bahasa *scripting,*
3. Bahasa *scripting* merupakan bahasa pemograman yang ringan,
4. *Javascript* berisi baris kode yang dijalankan di komputer (*web browser*),
5. *Javascript* biasanya disisipkan (*embedded*) dalam halaman HTML,
6. *Javascript* adalah bahasa interpreter (dieksekusi tanpa proses kompilasi),
7. Setiap orang dapat menggunakan *Javascript* tanpa membayar lisensi.
8. **Teknik Pengumpulan Data**

Dalam penelitian ini, peneliti menggunakan beberapa teknik pengumpulan data yakni, pengamatan (observasi), wawancara (*interview*), dokumentasi dan Studi Pustaka.

* + - 1. **Wawancara**

Wawancara adalah metode pengambilan data dengan cara menanyakan sesuatu kepada seseorang yang menjadi informan atau responden. Teknik wawancara yang digunakan dalam penelitian ini adalah wawancara bebas terpimpin, dimana pertanyaan yang di berikan tidak terpaku pada pedoman wawancara dan dapat diperdalam maupun dikembangkan sesuai dengan situasi dan kondisi di lapangan (Gulo & Hardiwati 2002).

* + - 1. **Observasi**

Observasi merupakan metode pengumpulan data di mana peneliti mencatat informasi selama penelitian (Gulo & Hardiwati, 2002), Data observasi berupa deskripsi yang faktual, cermat, dan terperinci mengenai lapangan, kegiatan kemanusiaan, dan situasi sosial serta di mana kegiatan–kegiatan itu terjadi.

* + - 1. **Dokumentasi**

Dokumentasi adalah teknik pengumpulan data yang ditujukan kepada subyek penelitian. Dokumentasi dalam penelitian ini lebih pada pengumpulan dokumentasi pendukung data-data penelitian yang dibutuhkan.

* + - 1. **Studi Pustaka**

Studi Pustaka merupakan langkah awal dalam metode pengumpulan data. Studi Pustaka merupakan metode pengumpulan data yang diarahkan kepada pencarian data dan informasi melalui dokumen-dokumen, baik dokumen tertulis, foto-foto, gambar, maupun dokumen elektronik yang dapat mendukung dalam proses penulisan.”Hasil penelitian juga akan semakin kredibel apabila didukung foto-foto atau karya tulis akademik dan seni yang telah ada.” (Sugiyono, 2005).

1. **UML (*Unified Modeling Language*)**

Menurut (Booch et al., 2005), UML adalah Bahasa standar untuk membuat rancangan *software*. UML biasanya digunakan untuk menggambarkan dan membangun, dokumen artifak dari *software intensive system*. Model UML yang dipakai dalam pembuatan Sistem Informasi Pencarian Golongan Darah antara lain *Use Case Diagram, Activity Diagram, Class Diagram, ER Diagram.*

***Use Case Diagram***

Menurut (Whitten et al., 2004), *Use case modeling* adalah sebuah pendekatan yang memfasilitasi pengembangan yang berpusat pada penggunaan. *Use case* *diagram* digunakan untuk memodelkan proses berdasarkan perspektif pengguna sistem. *Use case diagram* terdiri atas diagram untuk *use case* dan *actor*. *Actor* merepresentasikan orang yang akan mengoperasikan atau orang yang berinteraksi dengan sistem aplikasi. Simbol-simbol yang digunakan dalam *use case diagram* dapat dilihat pada Tabel 2.1.

**Tabel 2.1** *Use Case Digram*

| **No** | **Gambar** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| 1 |  | *Actor* | Menggambarkan sesorang yang beriteraksi dengan sistem, di mana hanya bisa menginputkan informasi dan menerima informasi dari sistem dan tidak memegang kendali pada *use case*. |
| 2 |  | *Association* | Menghubungkan *link* antar *element*. |
| 3 |  | *Generalization* | Sebuah elemen yang menjadi spesialisasi dari *element* yang lain |
| 4 |  | *Include* | Metode yang harus terpenuhi agar sebuah *event* dapat terjadi. |
| 5 |  | *Extend* | Metode yang hanya berjalan di bawah kondisi tertentu |
| 6 |  | *Dependency* | Sebuah *element* yang bergantung beberapa cara kepada elemen lainnya |

***Activity Diagram***

Menurut (Whitten et al., 2004), activity diagram digunakan untuk menggambarkan alur dari proses bisnis atau indaka – indaka *usecase* secara berurutan. Diagram ini juga digunakan untuk menggambar *action* (indakan) yang akan dieksekusikan ketika suatu proses sedang berjalan dan berserta hasil dari proses eksekusi tersebut. Simbol-simbol yang digunakan dalam *activity* diagram dapat dilihat pada Tabel 2.2.

**Tabel 2.2** *Activity Diagram*

| **No** | **Gambar** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| 1 |  | *Start Point* | Merupakan awal dalam aktifitas |
| 2 |  | *End Point* | Akhir dalam aktivitas |
| 3 |  | *Activites* | Menggambarkan proses kegiatan/aktifitas. |
| 4 |  | *Decision Point* | Menggambarkan pilihan pengambilan keputusan aktifitas. |
| 5 |  | *Swimlane* | Berfungsi sebagai pembagi *activity diagram* yang menunjuksiapa yang melakukan aktifitas. |

***Class Diagram***

Menurut (Whitten et al., 2004), *class diagram* menggambarkan struktur objek yang terdapat pada sebuah sistem. *Diagram* ini menunjukkan objek – objek yang terdapat pada suatu sistem dan relasi antar objek – objek tersebut. *Class* memiliki tiga area pokok yaitu: Nama, Atribut dan Metode / *Operation*. Atribut dan Metode dapat mempunyai sifat *Private* (tidak dapat dipanggil dari luar *class*), *Protected* (Hanya dapat dipanggil oleh class yang bersangkutan dan anak yang mewarisinya) dan *Public* (dapat dipanggil siapa saja). Berikut adalah contoh dari sebuah kelas. *Class Diagram* secara khas meliputi: Nama Kelas (*Class Name*), Atribut (*Attributes*), Operasi (*Operations*), dan Relasi (*Relationships*).

**Tabel 2.3** *Class Diagram*

| **No** | **Gambar** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| 1 |  | *Assosiation* | Hubungan statis antar kelas. Asosiasi menggambarkan kelas yang memiliki atribut berupa kelas lain. |
| 2 |  | *Directed Assosiation* | Asosiasi dengan makna kelas yang satu digunakan oleh kelas yang lain. |
| 3 |  | *Generalization* | Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus) atau untuk menyatakan hubungan *intheritance*. |
| 4 |  | *Dependency* | Relasi antar kelas dengan makna kebergantungan antar kelas. |
| 5 |  | *Agregation* | Hubungan yang menyatakan bahwa suatu kelas menjadi atribut bagi kelas lain. |
| 6 |  | *Composition* | Bentuk khusus dari agregasi dimana kelas yang menjadi bagian diciptakan setelah kelas menjadi *whole*. |
| 7 |  | *Realization* | Hubungan antar kelas dimana sebuah kelas memiliki keharusan untuk mengikuti aturan yang ditetapkan oleh kelas lainnya. |

***Entity Relationship Diagram***

Menurut (Brady & Loonam, 2010), *Entity Relationship Diagram* (ERD) merupakan teknik yang digunakan untuk memodelkan kebutuhan data dari suatu organisasi, biasanya oleh sistem analisis dalam tahap analisis persyaratan proyek pengembangan sistem Terdapat beberapa alat atau *tools* yang digunakan untuk membuat pemodelan data. Salah satunya adalah *Entity Reliationship Diagram* (ERD) yang merupakan pemodelan data yang menggunakan beberapa notasi untuk menggambarkan data dalam konteks entitas dan hubungan yang digambarkan oleh data tersebut. Untuk menggambarkan digunakan beberapa notasi dan simbol seperti pada tabel 2.4.

**Tabel 2.4** *Entity Relationship Diagram*

| **No** | **Gambar** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
| 1 |  | Entitas | Suatu objek yang dapat di identifikasi dalam lingkungan pemakai |
| 2 |  | Relasi | Menunjukkan adanya hubungan di antara sejumlah entitas yang berbeda |
| 3 |  | Atribut | Berfungsi mendeskripsikan karakter entitas(atribut yang berfungsi sebagai key diberi garis bawah). |
| 4 |  | *Association* | Sebagai penghubung antara relasi dengan entitas, relasi, dan entitas dengan atribut. |

1. **Figma**

Figma merupakan aplikasi berbasis website untuk mendesain sebuah user interface dan user experience yang dapat memudahkan desainer dalam membuat aplikasi, website, serta berbagai komponen user interface yang dapat digabungkan dalam project lainnya. Figma dapat digunakan diberbagai platform, seperti Mac, Windows, Linux bagi pengguna sistem operasi lain juga dapat mengakses melalui Website Figma.

1. **Metode *Waterfall***

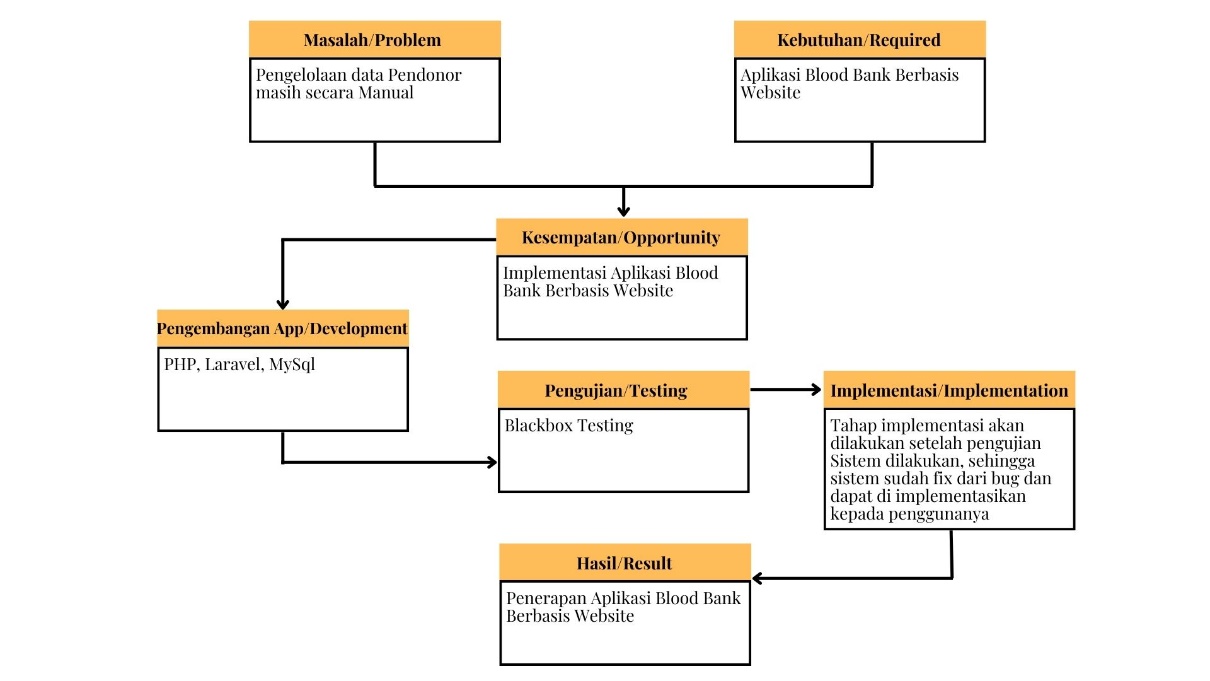
Model *waterfall* adalah model yang paling banyak digunakan untuk tahap pengembangan. Model *waterfall* ini juga dikenal dengan nama model tradisional atau model klasik. Model air terjun (*waterfall*) sering juga disebut model sekuensial linier (*squential linear*) atau alur hidup klasik (*Classic cycle*)”. Model air terjun ini menyediakan pendekatan alur hidup perangkat lunak secara sekuensial terurut dimulai dari analisis, desain, pengkodean, pengujian dan tahap pendukung (*support*) (Susilo, 2018).

1. **Metode *Blackbox***

Metode *Blackbox Testing* merupakan salah satu metode yang mudah digunakan karena hanya memerlukan batas bawah dan batas atas dari data yang di harapkan, Estimasi banyaknya data uji dapat dihitung melalui banyaknya *field* data entri yang akan diuji, aturan entri yang harus dipenuhi serta kasus batas atas dan batas bawah yang memenuhi. Dan dengan metode ini dapat diketahui jika fungsionalitas masih dapat menerima masukan data yang tidak diharapkan maka menyebabkan data yang disimpan kurang valid (Cholifah et al., 2018).

1. **Kerangka Konsep**

Kerangka konsep didalam Aplikasi Blood Bank Berbasis *Website* di Kabupaten Kampar (Studi Kasus: Palang Merah Indonesia (PMI)) ini mengacu pada perumusan masalah yang sudah dirumuskan dan mengacu pada materi Sistem Informasi.



**Gambar 2.3** Kerangka Konsep

Kerangka Konsep diatas sebagai berikut:

1. Masalah / *Problem*

Pada penelitian ini yang menjadi masalah adalah Unit Donor Darah di Palang Merah Indonesia (PMI) Bangkinang Kota masih mengolah data secara manual,

1. Kebutuhan / *Required*

Kebutuhan Sistem untuk mempermudah Unit Donor Darah di Palang Merah Indonesia (PMI) Bangkinang Kota mengolah, meng-*update,* maupun mencari data Pendonor,

1. Kesempatan / *Opportunity*

Implementasi Sistem ini adalah salah satu solusi yang tepat untuk memudahkan Unit Donor Darah di Palang Merah Indonesia (PMI) Bangkinang Kota,

1. Pengembangan Aplikasi / *Application Development*

Aplikasi berbasis *Website* ini dibangun dengan menggunakan *PHP* (*Hypertext Preprocessor*) Sebagai Bahasa pemograman, *Laravel* sebagai *Framework, MySql* sebagai *Web server,*

1. Pengujian / *Testing*

Tahap pengujian sistem dalam penelitian ini menggunakan pengujian yaitu pengujian *black box* *testing*, dimana pengujian *blackbox* berguna untuk menguji semua fitur sudah berjalan dengan seharusnya sebelum sistem di gunakan *user*,

1. Implementasi / *Implementation*

Tahap implementasi akan dilakukan setelah pengujian Sistem dilakukan, sehingga sistem sudah *fix* dari *bug* dan dapat di implementasikan kepada penggunanya,

1. Hasil / *Result*

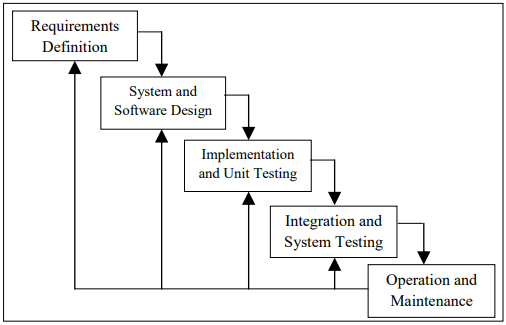
Hasil yang diharapkan dari sistem yang telah dibuat yaitu Implementasi Aplikasi Blood Bank Berbasis Website di Kabupaten Kampar ini dapat memudahkan Unit Donor Darah di Palang Merah Indonesia (PMI) Bangkinang Kota.

**BAB III**

**METODE PENELITIAN**

* 1. **Desain Penelitian**

Model *waterfall* adalah model yang paling banyak digunakan untuk tahap pengembangan. Model *Waterfall* ini juga dikenal dengan nama model tradisional atau model klasik. Model air terjun (*Waterfall*) sering juga disebut model sekuensial linier (*Squential Linear*) atau alur hidup klasik (*Classic Cycle*)”. Model air terjun ini menyediakan pendekatan alur hidup perangkat lunak secara sekuensial terurut dimulai dari analisis, desain, pengkodean, pengujian dan tahap pendukung (*Support*) (Susilo, 2018).

****

**Gambar 3.1** Metode *Waterfall*

**Sumber:** *E-book* Sommerville 2011

Menurut (Vilma et al., 2022) 5 tahap yang dapat untuk melakukan metode ini diantaranya :

1. (Analisis) Proses pengumpulan kebutuhan dilakukan secara intensif untuk mespesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa dibutuhkan oleh *user*,
2. (Desain) perangkat lunak adalah proses *multi* langka yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean,
3. (Pengodean) Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain,
4. (Pengujian) fokus pada perangkat lunak secara dari segi logis dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan,
5. Pendukung (*support*) atau Pemeliharaan (*maintenance*) Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru.
   1. **Lokasi Penelitian, Waktu Penelitian dan Sampel**
   2. Lokasi Penelitian

Palang Merah Indonesia (PMI) Kab. Kampar,

* 1. Waktu penelitian dilakukan pada bulan Juli – September 2022.
  2. **Sampel**

Sampel wilayah dalam penelitian ini adalah beberapa Pendonor di Palang Merah Indonesia (PMI) Kab. Kampar.

* 1. **Alat Penelitian**

Dalam pelaksanaan penelitian ini, ada beberapa alat bantu yang dibutuhkan agar terlaksananya penelitian ini dengan lebih efektif dan efisien, antara lain:

* + 1. Perangkat Keras (*Hardware*)

Perangkat keras (*Hardware*) yang digunakan dalam penelitian ini adalah sebagai berikut:

* + - 1. *Computer*, spesifikasi sebagai berikut:

1. *Processor* AMD Ryzen™ 5 2600 CPU @ 3.9Ghz,
2. *Memory* (RAM) 16000 MB.
   * + 1. *Mouse*, *Keyboard*, dan *Speakers*,
       2. *Printer*.
     1. Perangkat Lunak (*Software*)

Perangkat lunak (*Software*) yang digunakan dalam penelitian ini adalah sebagai berikut:

1. *Operating System Windows* 10 *Home,*
2. Aplikasi XAMPP,
3. *Laravel* sebagai bahasa pemograman *web*,
4. Google Chrome sebagai *web browser* yang digunakan,
5. StarUML untuk merancang diagram UML,
6. Snip & Sketch untuk membuat *capture* gambar,
7. Visual Code Studio sebagai aplikasi *code editor*,
8. Figma sebagai Desain *User Interface* (UI) *Website*,
9. Microsoft Office Professional 2016 untuk mengolah data penelitian.
   1. **Metode Pengumpulan Data**

Dalam penelitian ini, peneliti menggunakan beberapa teknik pengumpulan data kualitatif yakni, pengamatan (observasi), wawancara (*interview*), dokumentasi dan Studi Pustaka.

* + 1. **Observasi**

Observasi dilakukan dengan peninjauan langsung di Palang Merah Indonesia (PMI). Waktu pelaksanaan Observasi pada tanggal 15 April 2022 sampai 17 September 2022.

* + 1. **Wawancara**

Wawancara adalah proses memperoleh keterangan atau informasi dari narasumber. Peneliti melakukan wawancara dengan Staf di Palang Merah Indonesia (PMI).

* + 1. **Dokumentasi**

Dokumentasi sangat diperlukan bagi peneliti karena bertujuan untuk mempertajam analisis penelitian sebagai bukti yang sah dan aktual.

* + 1. **Studi Pustaka**

Peneliti melakukan Studi Pustaka untuk menggali teori-teori dasar dan konsep yang telah ditemukan oleh para peneliti terdahulu. Mengikuti perkembangan penelitian dalam bidang yang ingin diteliti.

* 1. ***Process Business***

Proses bisnis adalah suatu kumpulan aktivitas atau pekerjaan terstruktur yang saling terkait untuk menyelesaikan suatu masalah tertentu atau yang menghasilkan produk atau layanan. Berikut adalah table Proses Bisnis Palang Merah Indonesia (PMI) Kabupaten Kampar:

**Tabel 3.1** *Process Business*

| **Input** | **Process** | **Output** |
| --- | --- | --- |
| * Pendonor datang ke kantor Palang Merah Indonesia (PMI). * Staf Palang Merah Indonesia (PMI) terkadang mendirikan stand agar masyarakat dapat melakukan donor darah secara sukarela. | * Penyimpanan Darah menggunakan mesin Blood Bank Refrigerator di suhu -3°C sd -3,5°C. * Darah ini memiliki *expired* dalam jangka waktu 30 - 35 Hari. * Perdonor hanya bisa melakukan donor darah dalam jangka waktu 2 bulan satu kali. | * Pasien yang membutukan kantung darah harus mengisi biodata dan membawa surat rekomendasi dari Rumah Sakit ke Palang Merah Indonesia (PMI). * Pasien membawa atau mencari Pendonor ke Palang Merah Indonesia (PMI). |

**BAB IV**

**ANALISIS DAN IMPLEMENTASI**

* 1. **Analisis Perancangan Sistem**

Berdasarkan hasil penelitian dan pengamatan dari Perancangan Aplikasi *Bloodbank* berbasis *website* di Palang Merah Indonesia (PMI) Bangkinang Kota, memiliki permasalahan yaitu bagian Unit Donor Darah masih menggunakan pengolahan data dan pencarian data Pendonor secara manual sehingga membutuhkan waktu yang sangat banyak dan membuat berkurangnya kualitas pelayanan kepada Masyarakat. Dalam mengatasi masalah tersebut dibutuhkan suatu perangkat lunak untuk menangani sistem pengolahan dan pencarian data pendonor di-Palang Merah Indonesia (PMI) Bangkinang Kota untuk membantu para Staf Palang Merah Indonesia (PMI) mempercepat dan akurat dalam memberikan informasi yang lebih baik sehingga bisa membantu kualitas pelayan kepada masyarakat.

Berdasarkan masalah yang ditemukan maka dapat diusulkan untuk menggunakan sebuah sistem yang bisa memberikan informasi yang lebih cepat dan akurat serta memberikan kemudahan penggunanya untuk proses pengimputan, pencarian, dan penyimpanan data pendonor secara sistematis.

* 1. **Analisa Kebutuhan Fungsional Sistem**

Pada tahap ini kebutuhan fungsional sangat dibutuhkan dalam mengidentifikasi apa saja yang dibutuhkan dan diinginkan oleh pengguna dari sistem. Berikut adalah kebutuhan fungsional yang dijelaskan dalam bentuk *use case* diagram*, class* diagram, dan *activity* diagram.

* + - * 1. **Identifikasi Kasus Pengguna Bisnis**

Pada identifikasi ini, kasus penggunaan dapat melibatkan komunikasi antara sejumlah aktor. Orang akan melihat sistem kasusu penggunaan lebih tersruktur karena pada umumnya orang akan memberi tahu tentang apa saja yang bisa dilakukan.

Daftar *use case* bisnis:

* + 1. B1 : Ketersediaan *stock* darah : Pasien langsung datang ke Palang Merah Indonesia (PMI) untuk menanyakan *stock* darah yang tersedia,
    2. B2 : Syarat donor : Pendonor datang ke Palang Merah Indonesia (PMI) untuk melihat apa saja syarat donor darah, kelompok orang yang tidak boleh melakukan donor, dan jenis jenis golongan darah,
    3. B3 : Donor darah : Pendonor datang ke Palang Merah Indonesia (PMI) untuk mengisi formulir donor darah,
    4. B4 : *History* pendonor : Pendonor menanyakan langsung kepada staf Palang Merah Indonesia (PMI) atau melihat kartu donor untuk mengetahui sudah berapa kali melakukan donor darah,
    5. B5 : Pelayanan : staf Palang Merah Indonesia (PMI) masih menggunakan cara manual untuk mengolah dan mencari data dari pasien maupun pendonor,
    6. B6 : *History* Pengeluaran : staf Palang Merah Indonesia (PMI) masih menggunakan cara manual untuk mencatat pengeluaran darah.
       - 1. ***Usecase Diagram***

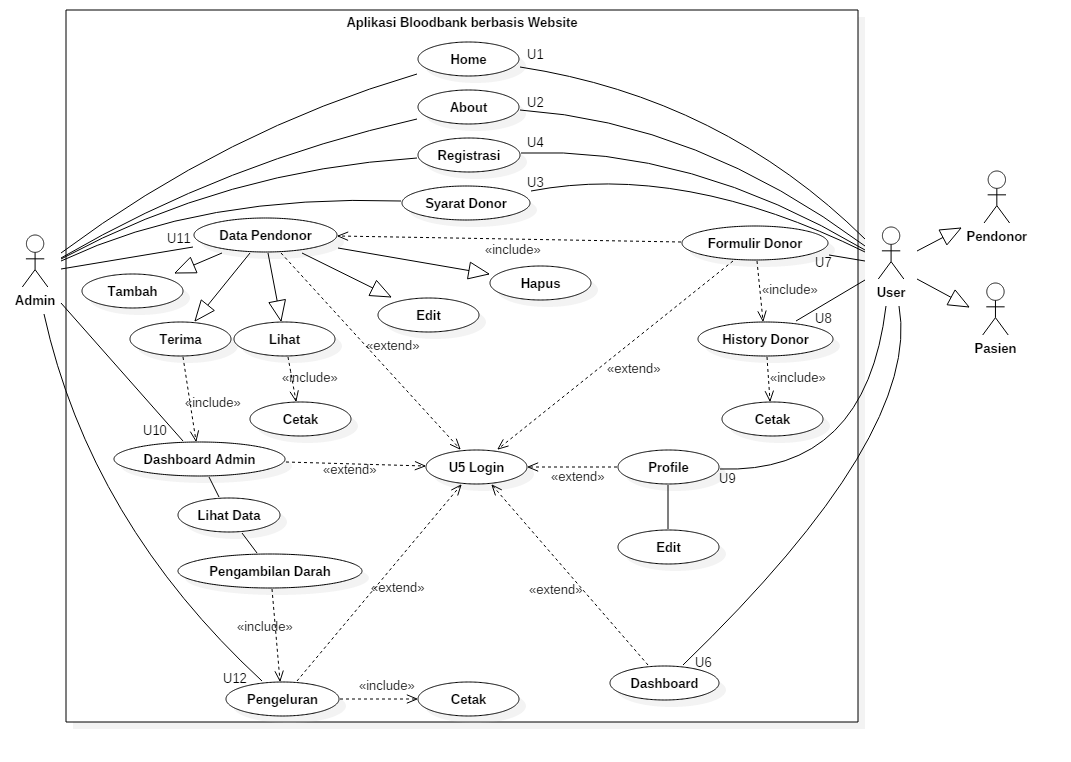
*Actor:*

1. Admin : pengolah semua data pada Aplikasi *Blood Bank* Berbasis *Website* di Palang Merah Indonesia (PMI),
2. Pendonor : pengguna yang sudah melakukan registrasi pada Aplikasi *Blood Bank* Berbasis *Website* di Palang Merah Indonesia (PMI),
3. Pengunjung : pengguna yang belum melakukan registrasi pada Aplikasi *Blood Bank* Berbasis *Website* di Palang Merah Indonesia (PMI).

Daftar *Usecase Diagram* Sistem:

1. U1 : *Homepage* : Halaman yang dapat di akses oleh seluruh *user* untuk melihat *stock* darah yang tersedia di Palang Merah Indonesia (PMI),
2. U2 : *Aboutpage* : Halaman yang dapat di akses oleh seluruh *user* untuk melihat informasi Palang Merah Indonesia (PMI),
3. U3 : Syarat donor : Halaman yang dapat di akses oleh seluruh *user*  untuk melihat Syarat donor, Kelompok yang tidak boleh melakukan donor, dan Jenis jenis golongan darah,
4. U4 : *Registration page* : Halaman untuk melakukan registrasi Pendonor pada Aplikasi *Blood Bank* Berbasis *Website* di Palang Merah Indonesia (PMI),
5. U5 : *Login poge* : Halaman untuk melakukan login Admin atau Pendonor pada Aplikasi *Blood Bank* Berbasis *Website* di Palang Merah Indonesia (PMI),
6. U6 : *Dashboard* Pendonor: Halaman yang bisa di akses oleh Pendonor untuk melihat jumlah *stock* darah yang tersedia di Palang Merah Indonesia (PMI),
7. U7 : Formulir Pendonor : Halaman yang di akses oleh pendonor untuk mengisi formulir donor,
8. U8 : *History* Pendonor : Halaman yang bisa di akses oleh Pendonor untuk melihat Riwayat donor, status donor, dan meng-*print* formulir donor yang sudah di *submit*,
9. U9 : *Profilepage* : Halaman yang bisa di akses oleh Pendonor untuk melihat dan meng-*edit profile*,
10. U10 : *Dashboard Admin* : Halaman yang bisa di akses oleh *Admin* untuk melihat jumlah *stock* darah, melihat data darah Pendonor, dan pengambilan *stock* darah yang tersedia,
11. U11 : Data Pendonor : Halaman yang bisa di akses oleh *Admin* untuk menambah, mengolah, dan mencari semua data Pendonor,
12. U12 : Pengeluaran : Halaman yang bisa di akses oleh *Admin* untuk melihat dan mencetak *history* pengeluaran darah di Palang Merah Indonesia (PMI).

Untuk lebih memahami perancangan *Usecase Diagram*, dapat dilihat pada gambar 4.1 berikut :



**Gambar 4.1 *Usecase Diagram* Aplikasi *Bloodbank* berbasis *Website***

Deskripsi *Usecase Diagram* pada Gambar 4.1 dapat dilihat pada:

**Tabel 4.1 Deskripsi *Usecase Diagram***

| ***Actor*** | **Deskripsi *Actor*** | ***Usecase Diagram*** | **Deskripsi *Usecase*** |
| --- | --- | --- | --- |
| *Admin* | Pengelola data Pendonor | *Login* | Pengguna melakukan *Login* agar dapat meng-akses sistem. |
| *Dashboard* | Pengguna bisa melihat berapa *stock* darah dan melihat semua datadarah pendonor serta melakukan pengambilan *stock* darah. |
|  |  | Data Pendonor | Pengguna bisa menambahkan, menerima, melihat, mengedit, dan menghapus data pendonor. |
| Pengeluaran | Pengguna bisa melihat dan mencetak riwayat pengeluran darah. |
| Pendonor | Melakukan Pengisian data Pendonor | *Login* | Pengguna melakukan *Login* agar dapat meng-akses sistem. |
| *Dashboard* | Pengguna bisa melihat berapa *stock* darah yang tersedia. |
| Formulir Donor | Pengguna melakukan pengisian Formulir donor. |
| *History* | Pengguna bisa melihat dan mecetak riwayat Formulir donor, dan melihat status donor. |
| *Profile* | Pengguna bisa melihat dan meng-*edit profile*. |
| Pengunjung | Melihat informasi yang ada | *Homepage* | Pengunjung bisa melihat berapa *stock* darah yang tersedia. |
| *Aboutpage* | Pengunjung bisa melihat beberapa informasi tentang Palang Merah Indonesia (PMI). |
| Syarat Donor | Pengunjung bisa melihat Syarat Donor Darah, Kelompok orang yang tidak boleh melakukan Donor Darah, dan Jenis Jenis Golongan Darah. |

1. Skenario *Usecase* Registrasi

**Tabel 4.2 Skenario *Usecase* Registrasi**

*Usecase* : Registrasi

*Actor* : Pendonor

Skenario : Melakukan Registrasi

Kondisi Awal : Sistem menampilkan halaman Registrasi

Kondisi Akhir : Sistem Menampilkan halaman *Login*

|  |  |
| --- | --- |
| **Skenario Normal** | |
| **Aksi *Actor*** | **Reaksi Sistem** |
| *Actor* meng-klik menu Registrasi. |  |
|  | * + - * 1. Sistem menampilkan *form* Registrasi. |
| * + - * 1. Aktor memasukkan *Fullname*, *Username, E-mail Address, Password*. |  |
|  | * 1. Sistem memproses data Registrasi. |
|  | * 1. Sistem menampilkan halaman *Login*. |

| **Skenario Gagal** | |
| --- | --- |
| **Aksi *Actor*** | **Reaksi Sistem** |
| 1. *Actor* meng-klik menu Registrasi. |  |
|  | 1. Sistem menampilkan *form* Registrasi. |
| 1. Aktor tidak memasukkan *Fullname*, *Username, E-mail Address, Password*. |  |
|  | 1. Sistem memproses Registrasi. |
|  | 1. Sistem menampilkan informasi untuk mengisi bidang ini. |

1. Skenario *Usecase Login*

**Tabel 4.3 Skenario *Usecase Login***

*Usecase* : *Login*

*Actor* : *Admin* dan Pendonor

Skenario : Melakukan *Login*

Kondisi Awal : Sistem menampilkan halaman *Login*

Kondisi Akhir : Sistem Menampilkan halaman *Dashboard*

| **Skenario Normal** | |
| --- | --- |
| **Aksi *Actor*** | **Reaksi Sistem** |
| *Actor* meng-klik menu *Login*. |  |
|  | * + - * 1. Sistem menampilkan *form* *Login*. |
| * + - * 1. *Actor* memasukkan *E-mail Address, Password*. |  |
|  | * 1. Sistem memproses data *Login*. |
|  | * 1. Sistem menampilkan halaman *Dashboard*. |

| **Skenario Gagal** | |
| --- | --- |
| **Aksi *Actor*** | **Reaksi Sistem** |
| 1. *Actor* meng-klik menu *Login*. |  |
|  | 1. Sistem menampilkan *form Login*. |
| 1. *Actor* memasukkan *E-mail Address, Password* yang salah. |  |
|  | 1. Sistem melakukan *verifikasi Login*. |
|  | 1. Sistem menampilkan informasi *E-mail Address* atau *Password* salah. |

1. Skenario *Usecase* Lihat dan Pengambilan *Stock* Darah

**Tabel 4.4 Skenario *Usecase* Lihat dan Pengambilan *Stock* Darah**

*Usecase* : Lihat dan Pengambilan *Stock* Darah

*Actor* : *Admin*

Skenario : Melakukan Lihat dan Pengambilan *Stock* Darah

Kondisi Awal : Sistem menampilkan halaman *Dashboard*

Kondisi Akhir : Sistem Menampilkan notif Berhasil

| **Skenario Normal** | |
| --- | --- |
| **Aksi *Actor*** | **Reaksi Sistem** |
| *Actor* meng-klik menu *Dashboard*. |  |
|  | * + - * 1. Sistem menampilkan Jenis dan *Stock* Darah yang tersedia. |
| * + - * 1. *Actor* meng-klik Lihat Data berdasarkan Golongan Darah yang ingin dilihat (A+, A-, B+, B-, O+, O-, AB+, AB-). |  |
|  | * 1. Sistem menampilkan Data Golongan Darah berdasarkan yang di pilih (A+, A-, B+, B-, O+, O-, AB+, AB-) secara detail. |
| * 1. *Actor* meng-klik *icon* ceklis |  |
|  | * 1. Sistem menampilkan halaman *form* Pengambilan Darah. |
| * 1. *Actor* memasukkan Nama Pengambil dan Tanggal Pengambilan |  |
|  | * 1. Sistem memproses data Pengambilan |
|  | * 1. Sistem menampilkan notif Berhasil |

| **Skenario Gagal** | |
| --- | --- |
| **Aksi *Actor*** | **Reaksi Sistem** |
| 1. *Actor* meng-klik *icon* ceklis. |  |
|  | 1. Sistem menampilkan *form* Pengambilan Darah. |
| 1. *Actor* memasukkan Nama Pengambil dan Tanggal Pengambilan yang salah. |  |
|  | 1. Sistem memproses data pengambilan. |
|  | 1. Sistem menampilkan notif Gagal. |

* + 1. Skenario *Usecase* Data Pendonor

**Tabel 4.5 Skenario *Usecase* Data Pendonor**

*Usecase* : Data Pendonor

*Actor* : *Admin*

Skenario : Mengolah dan Mencari Data Pendonor

Kondisi Awal : Sistem menampilkan halaman Data Pendonor

Kondisi Akhir : Sistem Menampilkan notif Berhasil

| **Skenario Normal** | |
| --- | --- |
| **Aksi *Actor*** | **Reaksi Sistem** |
| *Actor* meng-klik menu Data Pendonor. |  |
|  | * + - * 1. Sistem menampilkan halaman semua Data Pendonor. |
| * + - * 1. *Actor* meng-klik Tambah Data Pendonor |  |
|  | * 1. Sistem menampilkan FormulirDonor |
| * 1. *Actor* mengisi semua bidang Formulir Donor |  |
|  | * 1. Sistem memproses Tambah data Formulir Donor. |
|  | * 1. Sistem menampilkan notif Telah Berhasil ditambahkan |
| * 1. *Actor* meng-klik *icon update* |  |
|  | * 1. Sistem menampilkan Formulir Donor yang telah diisi |
| * 1. *Actor* mengganti bidang Formulir Donor yang telah diisi |  |
|  | * 1. Sistem memproses *Update* Data |
|  | * 1. Sistem menampilkan notif Telah Berhasil di-*update* |
| * 1. *Actor* meng-klik *icon* lihat pada salah satu data Pendonor |  |
|  | * 1. Sistem menampilkan detail salah satu data Pendonor |
| * 1. *Actor* meng-klik *icon* cetak |  |
|  | * 1. Sistem menampilkan detail salah satu data Pendonor dalam bentuk PDF. |
| * 1. *Actor* meng-klik *icon* centang |  |
|  | * 1. Sistem memproses data dan mengirim data tersebut ke Jenis dan *Stock* Darah yang tersedia. |
|  | * 1. Sistem menampilkan notif Data telah diterima |
| * 1. *Actor* meng-klik *icon* silang |  |
|  | * 1. Sistem menghapus salah satu data Pendonor |
|  | * 1. Sistem menampilkan notif Data telah berhasil dihapus |
| * 1. *Actor* melakukan pencarian berdasarkan (Nama, Golongan Darah, No. Kantong, Alamat) di-*Search Bar*. |  |
|  | * 1. Sistem mencari berdasarkan (Nama, Golongan Darah, No. Kantong, Alamat) yang di ketik oleh *Actor*. |
|  | * 1. Sistem menampilkan hasil dari pencarian *Actor*. |

| **Skenario Gagal** | |
| --- | --- |
| **Aksi *Actor*** | **Reaksi Sistem** |
| 1. *Actor* meng-klik Tambah Data Pendonor. |  |
|  | 1. Sistem menampilkan Formulir Donor. |
| 1. *Actor* tidak mengisi Formulir Donor secara lengkap |  |
|  | 1. Sistem memproses Formulir Donor. |
|  | 1. Sistem menampilkan informasi harap isi tabel ini. |
| 1. *Actor* tidak melakukan pencarian berdasarkan (Nama, Golongan Darah, No. Kantong, Alamat) atau mengetik *keyword* secara acak di-*Search Bar*. |  |
|  | 1. Sistem menampilkan notif Data Tidak ditemukan. |

* + 1. Skenario *Usecase* Pengeluaran

**Tabel 4.6 Skenario *Usecase* Pengeluaran**

*Usecase* : Pengeluaran

*Actor* : *Admin*

Skenario : Mengolah, Mencari dan Mencetak Pengeluaran

Kondisi Awal : Sistem menampilkan halaman Pengeluaran

Kondisi Akhir :

| **Skenario Normal** | |
| --- | --- |
| **Aksi *Actor*** | **Reaksi Sistem** |
| *Actor* meng-klik menu Pengeluaran. |  |
|  | * + - * 1. Sistem menampilkan halaman semua Data Pengeluran. |
| * + - * 1. *Actor* meng-klik *icon* lihat |  |
|  | * 1. Sistem menampilkan salah satu data Pendonor secara detail |
| * 1. *Actor* meng-klik *icon* cetak |  |
|  | * 1. Sistem mencetak semua data Pengeluaran |
| * 1. *Actor* melakukan pencarian berdasarkan (Nama, Golongan Darah, No. Kantong) di-*Search Bar*. |  |
|  | * 1. Sistem mencari berdasarkan (Nama, Golongan Darah, No. Kantong) yang di ketik oleh *Actor*. |
|  | * 1. Sistem menampilkan hasil dari pencarian *Actor*. |

| **Skenario Gagal** | |
| --- | --- |
| **Aksi *Actor*** | **Reaksi Sistem** |
| 1. *Actor* tidak melakukan pencarian berdasarkan (Nama, Golongan Darah, No. Kantong) atau mengetik *keyword* secara acak di-*Search Bar*. |  |
|  | 1. Sistem menampilkan notif Data Tidak ditemukan. |

* + 1. Skenario *Usecase* Formulir Pendonor

**Tabel 4.7 Skenario *Usecase* Formulir Pendonor**

*Usecase* : Formulir Pendonor

*Actor* : Pendonor

Skenario : Mengisi Formulir Donor

Kondisi Awal : Sistem menampilkan Formulir Donor

Kondisi Akhir : Sistem menampilkan Halaman *History*

| **Skenario Normal** | |
| --- | --- |
| **Aksi *Actor*** | **Reaksi Sistem** |
| *Actor* meng-klik menu Formulir Donor. |  |
|  | * + - * 1. Sistem menampilkan Formulir Donor. |
| * + - * 1. *Actor* mengisi semua tabel di Formulir Donor |  |
|  | * 1. Sistem memproses data Formulir Donor. |
|  | * 1. Sistem menampilkan halaman *History* Donor dan menampilkan notif Telah berhasil ditambahkan. |

| **Skenario Gagal** | |
| --- | --- |
| **Aksi *Actor*** | **Reaksi Sistem** |
| 1. *Actor* mengisi semua tabel di Formulir Donor |  |
|  | 1. Sistem menampilkan informasi harap isi tabel ini. |

* + 1. Skenario *Usecase History*

**Tabel 4.8 Skenario *Usecase History* Pendonor**

*Usecase* : *History* Pendonor

*Actor* : Pendonor

Skenario : Melihat *History* Donor

Kondisi Awal : Sistem menampilkan Halaman *History*

Kondisi Akhir :

| **Skenario Normal** | |
| --- | --- |
| **Aksi *Actor*** | **Reaksi Sistem** |
| *Actor* meng-klik menu *History*. |  |
|  | * + - * 1. Sistem menampilkan *History* dan Status Donor. |
| * + - * 1. *Actor* meng-klik *icon*  lihat |  |
|  | * 1. Sistem menampilkan salah satu data secara detail |

* + 1. Skenario *Usecase Profile*

**Tabel 4.9 Skenario *Usecase Profile* Pendonor**

*Usecase* : *Profile* Pendonor

*Actor* : Pendonor

Skenario : Mengolah *Profile* Pendonor

Kondisi Awal : Sistem menampilkan Halaman *Profile*

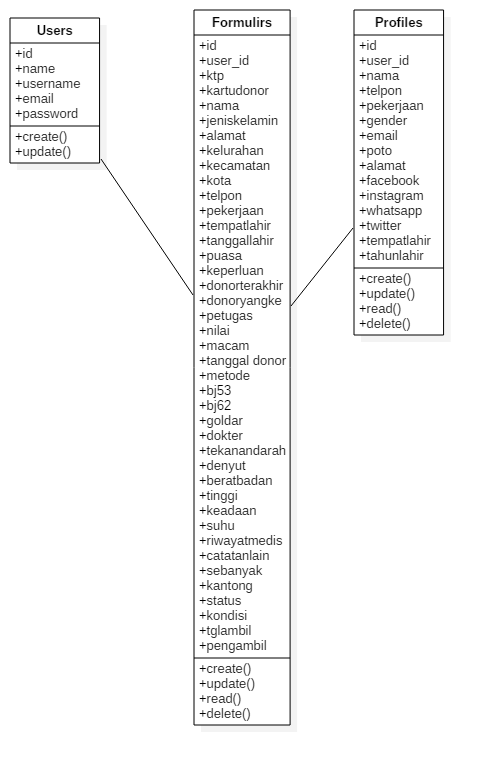
Kondisi Akhir : Sistem Menampilkan notif Berhasil

| **Skenario Normal** | |
| --- | --- |
| **Aksi *Actor*** | **Reaksi Sistem** |
| *Actor* meng-klik menu *Profile*. |  |
|  | * + - * 1. Sistem menampilkan *Profile page*. |
| * + - * 1. *Actor* meng-klik *Edit Profile* |  |
|  | * 1. Sistem menampilkan Formulir *Edit Profile* |
| * 1. *Actor* mengisi tabel Formulir yang masih kosong dan memasukkan poto dengan jenis *file* jpg,jpeg dan ukuran maximal 2mb. |  |
|  | * 1. Sistem memproses data *Profile* yang telah di isi |
|  | * 1. Sistem menampilkan notif *Profile* telah di­-*update* |

| **Skenario Gagal** | |
| --- | --- |
| **Aksi *Actor*** | **Reaksi Sistem** |
| 1. *Actor* mengisi tidak memasukkan poto dengan jenis *file* jpg,jpeg dan ukuran poto melebihi 2mb (*megabyte*). |  |
|  | 1. Sistem menampilkan notif jenis *file* poto harus jpg, jpeg dan ukuran maximal poto tidak lebih dari 2mb (*megabyte*). |

1. ***Class Diagram***

*Class Diagram* pada sistem dapat dilihat pada Gambar 4.2

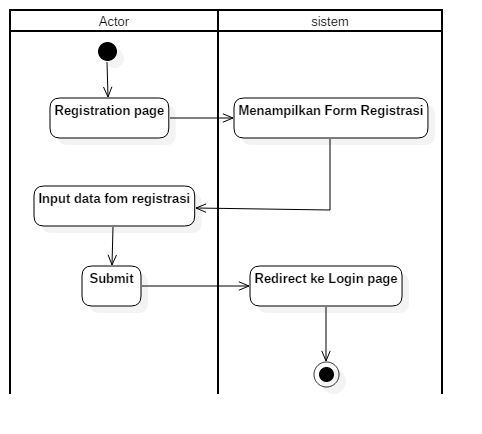


**Gambar 4.2 *Class Diagram***

1. ***Activity Diagram***

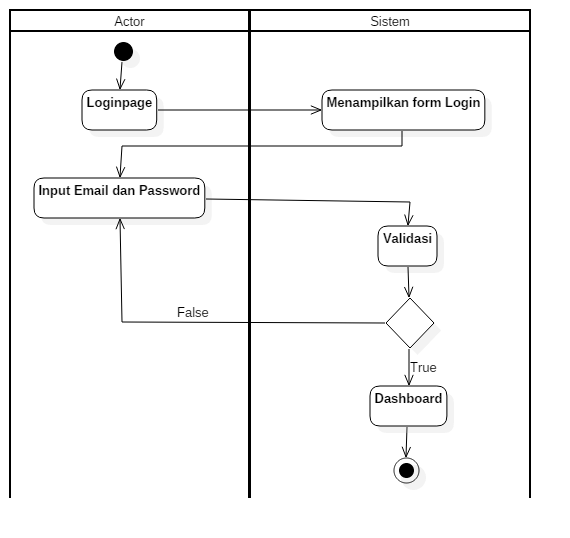
Diagram ini digunakan untuk menentukan apa saja aktifitas antara aktor dengan sistem.

* + 1. *Activity Diagram* Registrasi



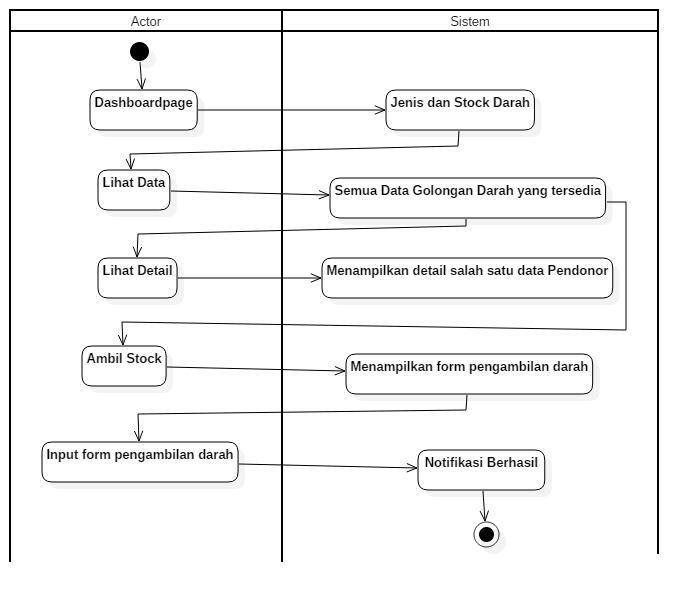
**Gambar 4.3 *Activity Diagram* Registrasi**

* + 1. *Activity Diagram* Login



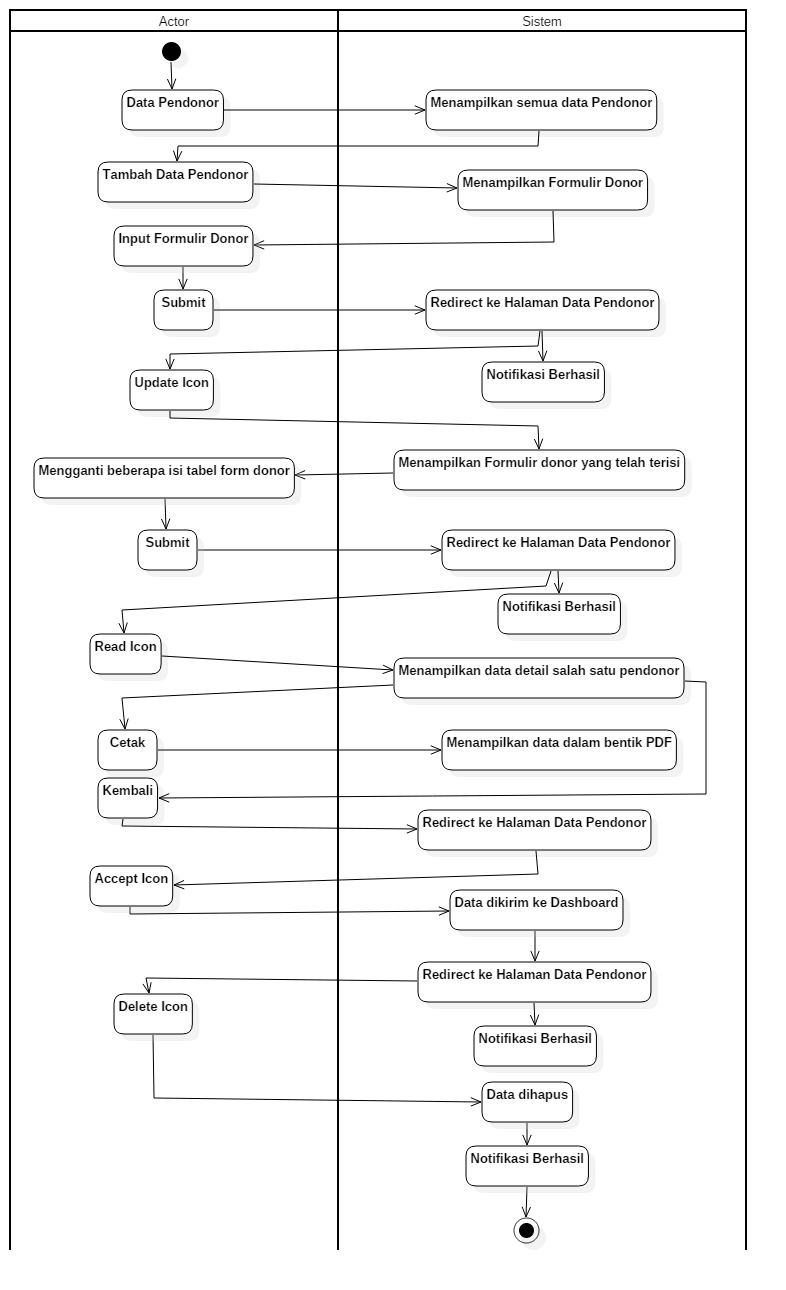
**Gambar 4.4 *Activity Diagram* Login**

* + 1. *Activity Diagram Dashboard Admin*



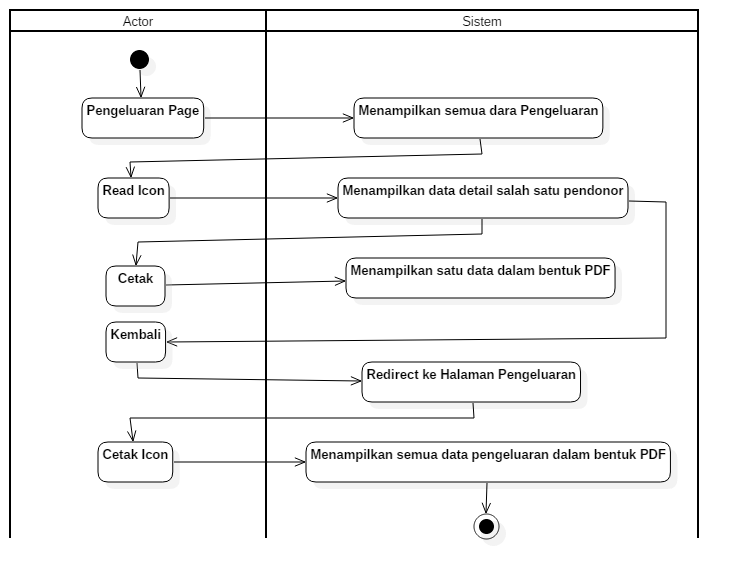
**Gambar 4.5 *Activity Diagram Dashboard Admin***

* + 1. *Activity Diagram* Data Pendonor



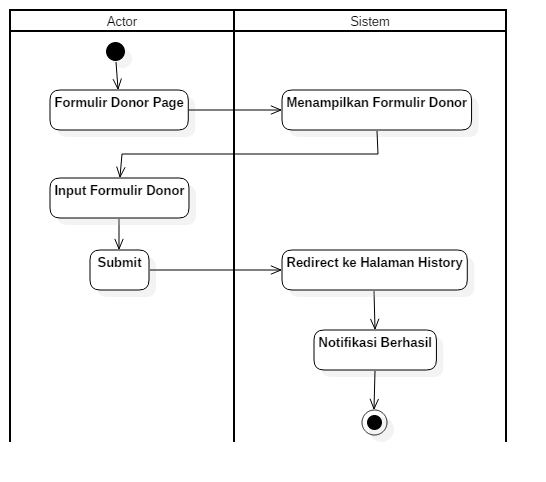
**Gambar 4.6 *Activity Diagram* Data Pendonor**

* + 1. *Activity Diagram* Pengeluaran



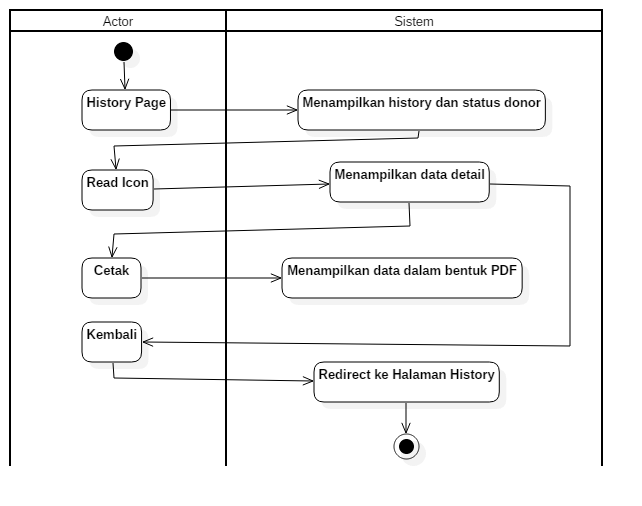
**Gambar 4.7 *Activity Diagram* Pengeluaran**

* + 1. *Activity Diagram* Formulir Donor

**

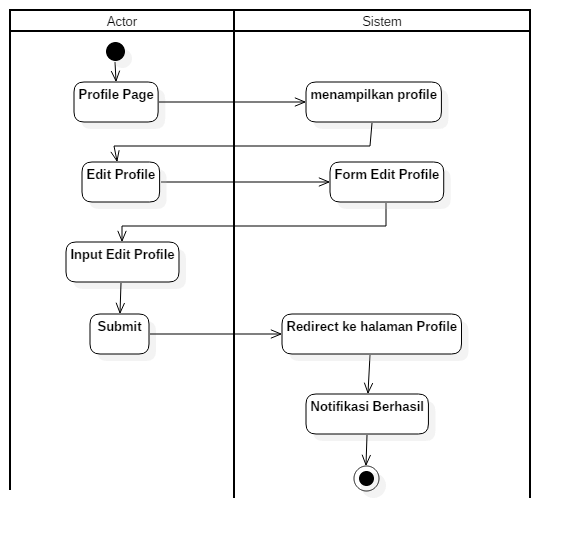
**Gambar 4.8 *Activity Diagram* Formulir Donor**

* + 1. *Activity Diagram* *History*

**

**Gambar 4.9 *Activity Diagram History***

* + 1. *Activity Diagram* *Profile*



**Gambar 4.10 *Activity Diagram Profile***

* 1. **Struktur Aplikasi yang digunakan**

Pada Aplikasi *Bloodbank* berbasis *website* Kabupaten Kampar menyediakan berbagai informasi dan data yang sesuai di Palang Merah Indonesia. Berikut halaman yang deindex paling pertama diakses oleh *user* dan menghubungkan ke halaman satu ke halaman lainnya.

* + - 1. *Homepage*

Merupakan halaman yang pertama kali diakses oleh *user* dan halaman ini memberikan informasi Jenis dan *Stock* Darah yang tersedia di Palang Merah Indonesia (PMI) Kabupaten Kampar,

* + - 1. *Aboutpage*

Halaman ini memberikan beberapa informasi tentang Palang Merah Indonesia (PMI) Kabupaten Kampar seperti (Tentang, Alamat, *Costumer Service*, dll),

* + - 1. Syarat Donor

Halaman ini memberikan informasi tentang (Syarat Donor Darah, Kelompok Orang yang Tidak Boleh Melakukan Donor Darah, Jenis Jenis Golongan Darah),

* + - 1. *Dashboard Admin*

Halaman ini hanya bisa diakses oleh *Admin* yang mana halaman ini menampilkan detail *stock* darah yang tersedia, dan melakukan pengambilan darah,

* + - 1. Data Pendonor

Halaman ini hanya bisa diakses oleh *Admin* yang mana halaman ini untuk mengolah (Menambah, Meng-*update*, Melihat, Mencari, Menerima, Mencetak, dan Menghapus) data dari Formulir Pendonor,

* + - 1. Pengeluaran

Halaman ini hanya bisa diakses oleh *Admin* yang mana halaman ini untuk melihat, dan mencetak Pengeluaran Darah di Palang Merah Indonesia (PMI),

* + - 1. Formulir Donor

Halaman ini bisa di akses oleh Pendonor untuk mengisi Formulir Donor,

* + - 1. *History* Donor

Halaman ini bisa di akses oleh Pendonor untuk melihat dan mencetak riwayat serta status donor,

* + - 1. *Profilepage*

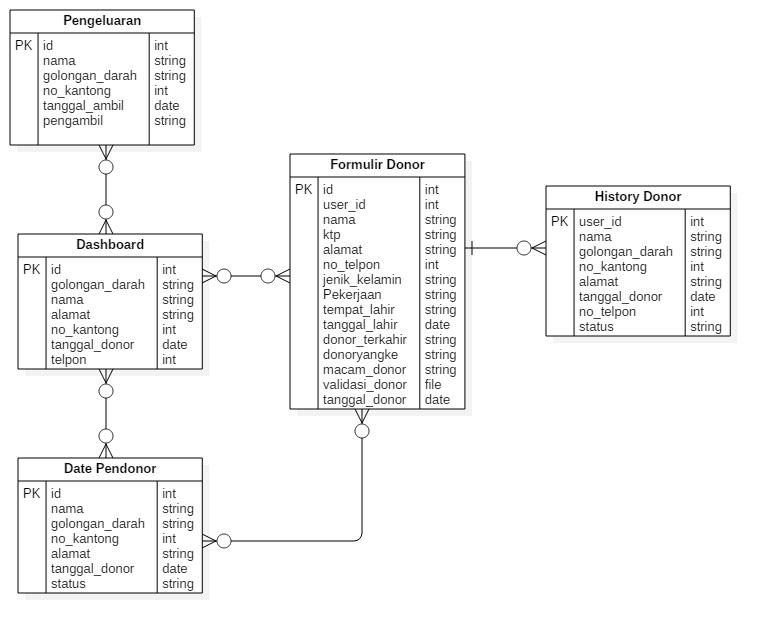
Halaman ini bisa di akses oleh Pendonor untuk melihat dan meng-*edit* halaman *Profile*.

* 1. **Rancangan**

Basis data sangat diperlukan untuk mengakses file didalam program, karena dari perancangan basis data bisa menemukan data-data yang dibutuhkan dalam sistem, sehingga informasi yang dihasilkan bisa terpenuhi dengan baik dan sesuai dengan yang dibutuhkan. Perancangan pertama yang lakukan adalah perancangan ERD dan perancangan *database*.

* + - 1. *Entity Relationship Diagram* (ERD)

Pada perancangan ini kita bisa mengetahui hubungan antara satu entity dengan entity yang lain dan nantinya berguna sebagai dasar dalam perancangan program. Hubungan entitas dapat dilihat pada gambar dibawah ini.



**Gambar 4.11 *Entity Relationship Diagram* (ERD)**

1. Perancangan *Database*

Pada perancangan *database*, sistem ini menggunakan 3 (tiga) tabel yaitu *users,* formulirs, *profiles*. Berikut perancangan *database* setiap tabelnya:

* + 1. Tabel *users*

Nama *database* : *project*

Nama tabel : *users*

*Primary key* : id

**Tabel 4.10 Perancangan tabel *users***

| **No** | **Atribut/*field*** | ***Type* data** | ***Length/*Values** |
| --- | --- | --- | --- |
| 1 | id | bigint | 20 |
| 2 | name | varchar | 255 |
| 3 | username | varchar | 255 |
| 4 | E-mail | varchar | 255 |
| 5 | password | varchar | 255 |
| 6 | roles | tinyint | 1 |

* + 1. Tabel *profiles*

Nama *database* : *project*

Nama tabel : *profiles*

*Primary key* : id

**Tabel 4.11 Perancangan tabel *profiles***

| **No** | **Atribut/*field*** | ***Type* data** | ***Length/*Values** |
| --- | --- | --- | --- |
| 1 | id | bigint | 20 |
| 2 | user\_id | bigint | 20 |
| 3 | nama | varchar | 255 |
| 4 | telpon | varchar | 255 |
| 5 | pekerjaan | varchar | 255 |
| 6 | gender | varchar | 255 |
| 7 | email | varchar | 255 |
| 8 | poto | varchar | 255 |
| 9 | alamat | varchar | 255 |
| 10 | facebook | varchar | 255 |
| 11 | instagram | varchar | 255 |
| 12 | whatsapp | varchar | 255 |
| 13 | twitter | varchar | 255 |
| 14 | tempatlahir | varchar | 255 |
| 15 | tahunlahir | date | - |

* + 1. Tabel *formulirs*

Nama *database* : *project*

Nama tabel : *formulirs*

*Primary key* : id

**Tabel 4.12 Perancangan tabel *formulirs***

| **No** | **Atribut/*field*** | ***Type* data** | ***Length/*Values** |
| --- | --- | --- | --- |
| 1 | id | bigint | 20 |
| 2 | user\_id | bigint | 20 |
| 3 | ktp | varchar | 255 |
| 4 | kartudonor | varchar | 255 |
| 5 | nama | varchar | 255 |
| 6 | jeniskelamin | varchar | 255 |
| 7 | alamat | varchar | 255 |
| 8 | kelurahan | varchar | 255 |
| 9 | kecamatan | varchar | 255 |
| 10 | kota | varchar | 255 |
| 11 | telpon | varchar | 255 |
| 12 | pekerjaan | varchar | 255 |
| 13 | tempatlahir | varchar | 255 |
| 14 | tanggallahir | date | - |
| 15 | puasa | varchar | 255 |
| 16 | keperluan | varchar | 255 |
| 17 | donorterakhir | date | - |
| 18 | donoryangke | varchar | 255 |
| 19 | validasi | varchar | 255 |
| 20 | validasipoto | varchar | 255 |
| 21 | riwayatdonor | varchar | 255 |
| 22 | petugas | varchar | 255 |
| 23 | nilai | varchar | 255 |
| 24 | macam | varchar | 255 |
| 25 | tanggaldonor | date | - |
| 26 | metode | varchar | 255 |
| 27 | bj53 | varchar | 255 |
| 28 | bj62 | varchar | 255 |
| 29 | goldar | varchar | 255 |
| 30 | dokter | varchar | 255 |
| 31 | tekanandarah | varchar | 255 |
| 32 | denyut | varchar | 255 |
| 33 | beratbadan | varchar | 255 |
| 34 | tinggi | varchar | 255 |
| 35 | keadaan | varchar | 255 |
| 36 | suhu | varchar | 255 |
| 37 | riwayatmedis | text | - |
| 38 | catatanlain | text | - |
| 39 | sebanyak | varchar | 255 |
| 40 | kantong | varchar | 255 |
| 41 | status | varchar | 255 |
| 42 | kondisi | varchar | 255 |
| 43 | tglambil | date | - |
| 44 | pengambil | varchar | 255 |

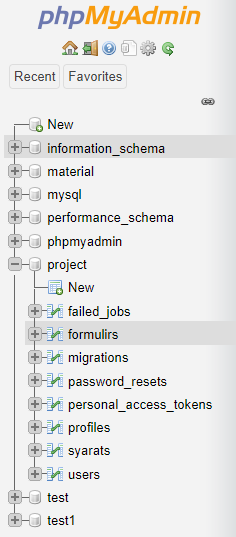
* 1. **Implementasi**

Implementasi sistem yaitu merupakan beberapa tahap penerapan sistem yang akan dilakukan jika perancangan sistem telah siap untuk dibuat dan di operasikan. Berikut ada beberapa tahap dalam implementasi yaitu:

* + - 1. Implementasi *database*

Pembuatan dari *database* ini dilakukan dengan cara menggunakan *database* MySQL, berikut adalah tampilan database dari Aplikasi *Bloodbank* Berbasis *Website* di Kabupaten Kampar (Palang Merah Indonesia (PMI)).

1. *Database Project*
2. Merupakan nama dari *database* yang telah digunakan untuk Aplikasi *Bloodbank* Berbasis *Website* di Kabupaten Kampar. Berikut adalah *database Project* pada Gambar 4.12.

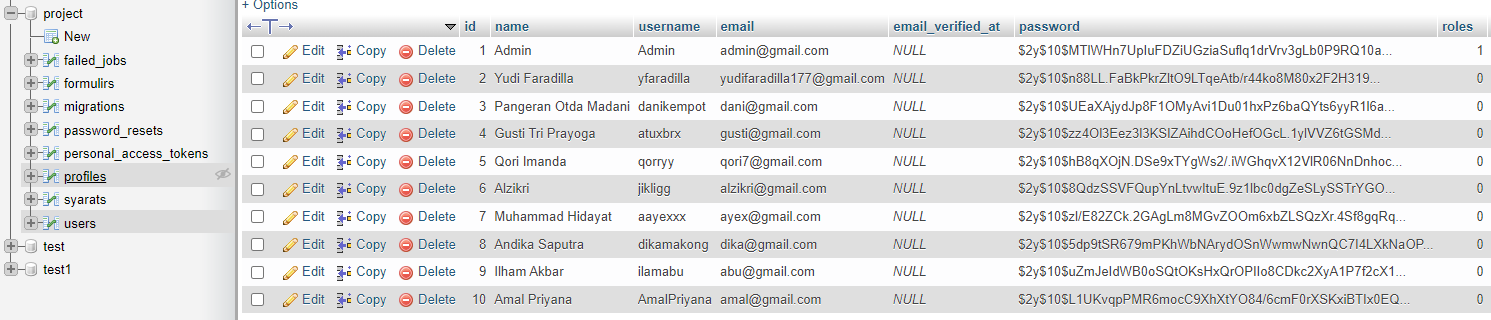


**Gambar 4.12 *Database Project***

Pada gambar 4.12 menampilkan *database Project* yang memiliki 3 (tiga) struktur tabel dimana setiap tabel memiliki peran yang khusus pada setiap sistem yang terdiri dari *users*, *profiles*, formulirs.

1. Struktur tabel *users*

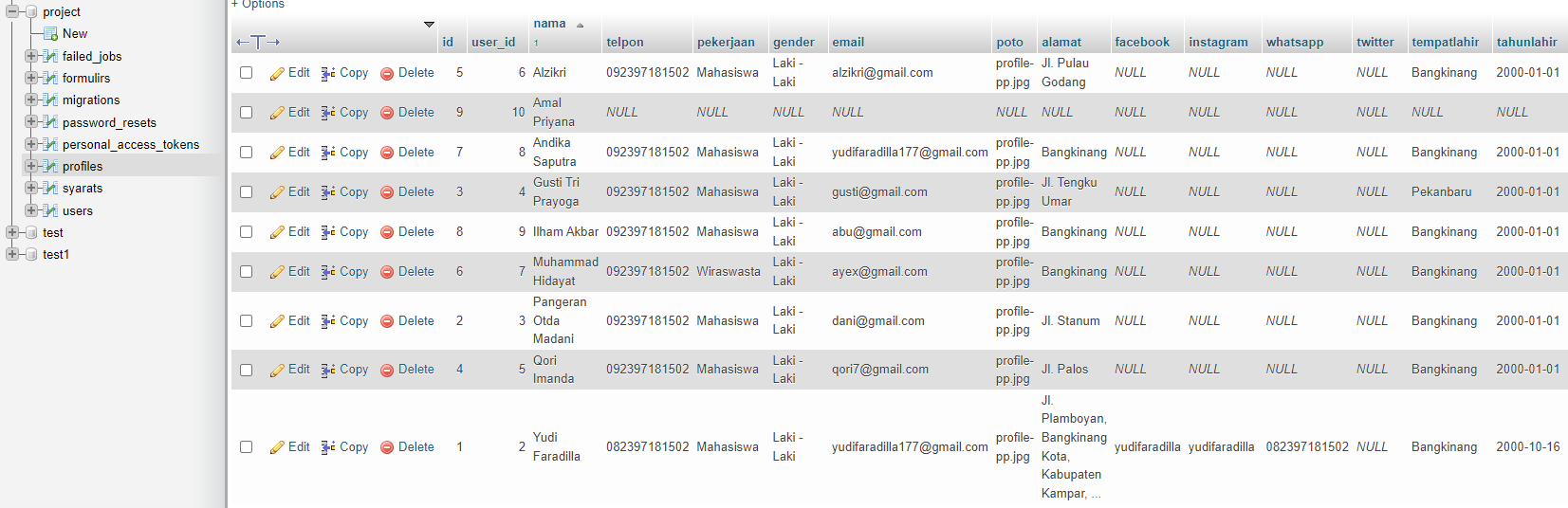
Ada beberapa struktur tabel *users* berisikan *field* yaitu seperti id, *name, username, email, password, roles.* Berikut adalah struktur tabel *users* pada Gambar 4.13.

****

**Gambar 4.13 Struktur Tabel *Users***

1. Struktur tabel *profiles*

Pada struktur tabel *profiles* berisikan *field* seperti id, user\_id, nama, telpon, pekerjaan, gender, email, poto, alamat, facebook, instagram, whatsapp, twitter, tempatlahir, tahunlahir. Berikut adalah struktur tabel *profiles* pada Gambar 4.14.



**Gambar 4.14 Struktur Tabel *Profiles***

1. Struktur tabel formulirs

Pada struktur tabel formulirs berisikin *field* seperti id, user\_id, ktp, kartudonor, nama, jeniskelamin, alamat, kelurahan, kecamatan, kota, telpon, pekerjaan, tempatlahir, tanggallahir, puasa, keperluan, donorterakhir, donoryangke, validasi, validasipoto, riwayatdonor, petugas, nilai, macam, tanggaldonor, metode, bj53, bj62, goldar, dokter, tekanandarah, denyut, beratbadan, tinggi, keadaan, suhu, riwayatmedis, catatanlain, sebanyak, kantong, status, kondisi, tglambil, pengambil. Berikut adalah struktur tabel formulirspada Gambar 4.15.



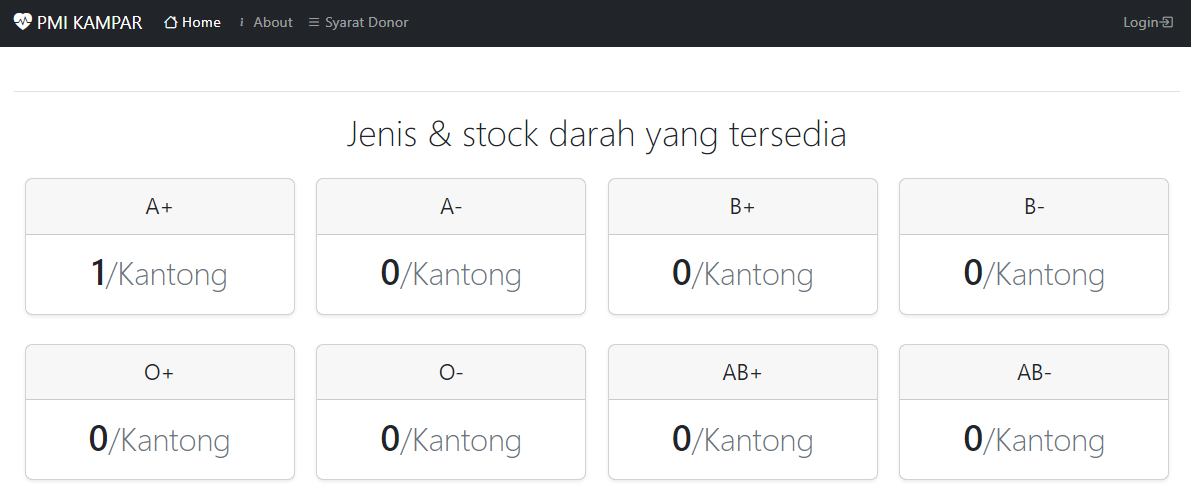
**Gambar 4.15 Struktur Tabel Formulirs**

* + - 1. Implementasi Sistem

Tahapan ini merupakan hasil implementasi dari analisa dan perancangan yang telah dibangun. Pada tahap ini memperlihatkan tampilan dengan 2 (dua) *actor* yang terlibat. Berikut tampilan *User Interface* (UI) Aplikasi *Bloodbank* Berbasis *Website* di Kabupaten Kampar (Palang Merah Indonesia (PMI)).

1. *Homepage*

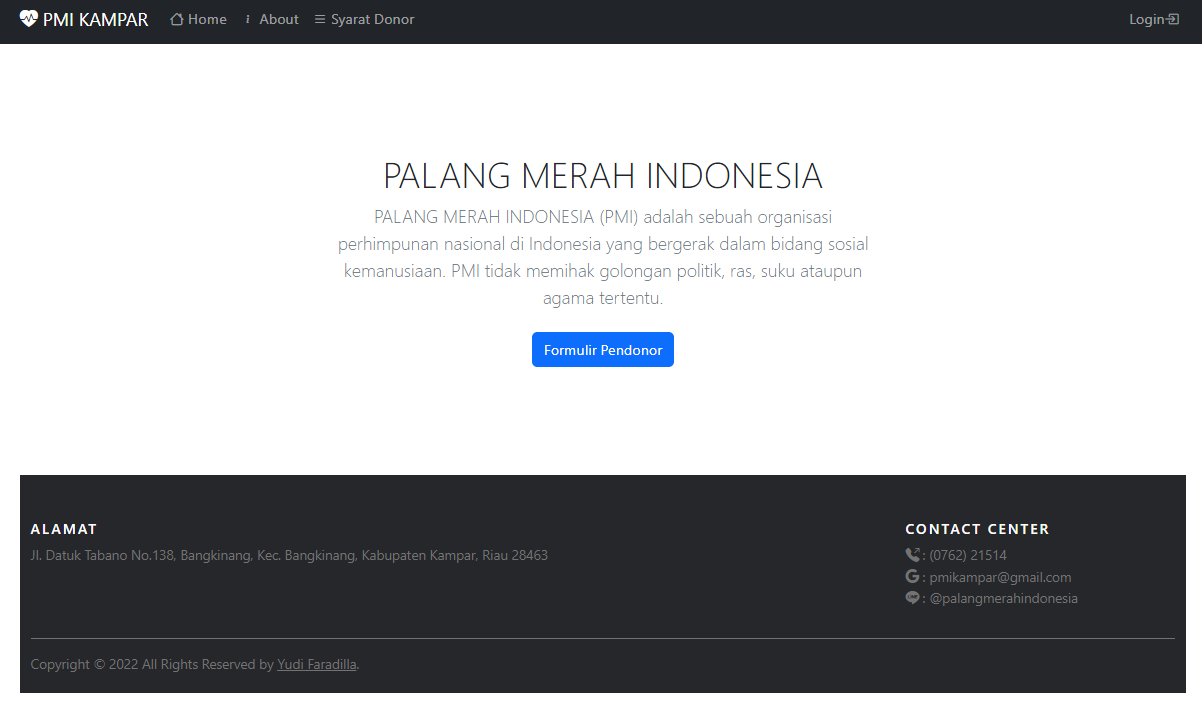
Pada halaman *home*, terdapat informasi tentang Jenis & *Stock* Golongan Darah yang Tersedia di Palang Merah Indonesia (PMI). Berikut tampilan halaman *home* yang dapat dilihat pada Gambar 4.16.



**Gambar 4.16 Tampilan *Homepage***

1. *Aboutpage*

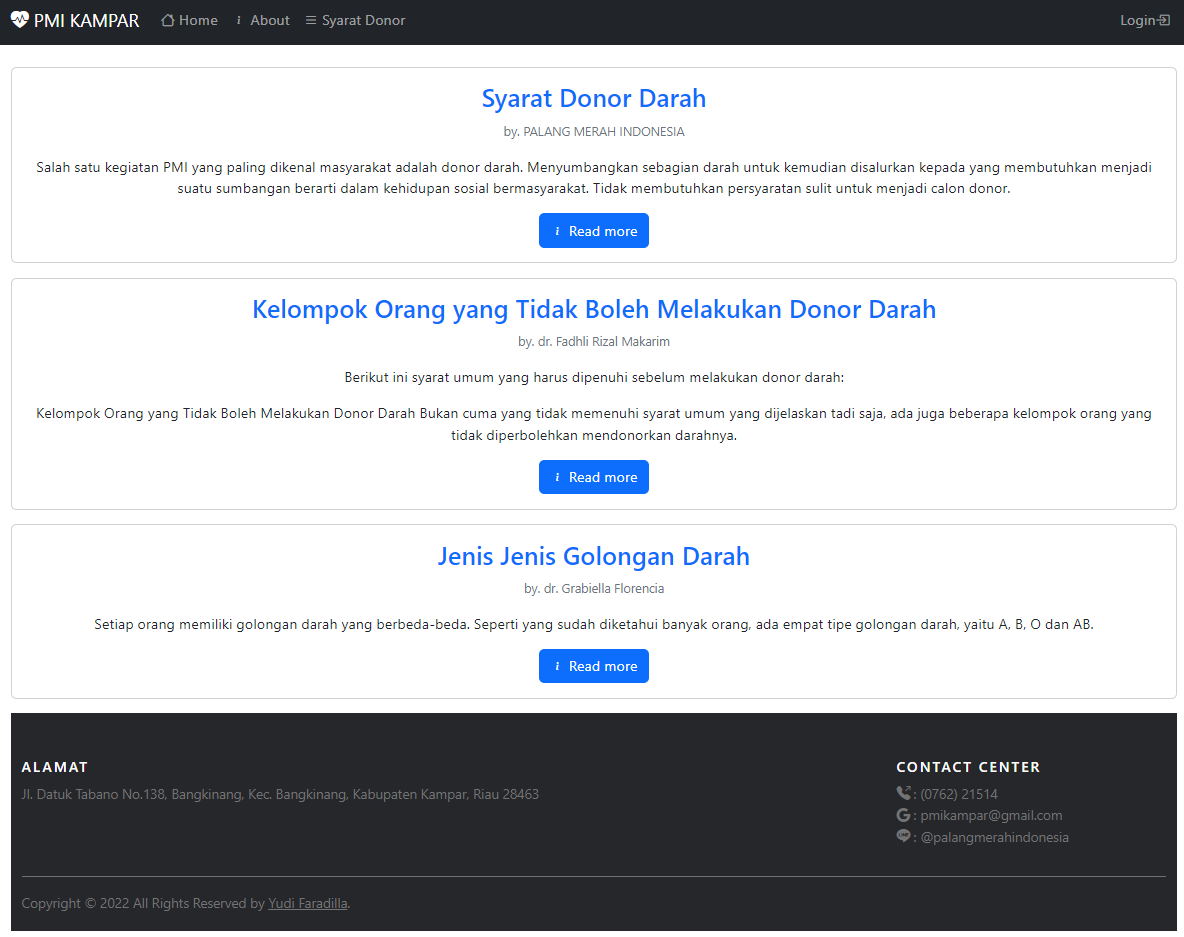
Pada halaman *about*, terdapat informasi di Palang Merah Indonesia (PMI) seperti ( Tentang, Alamat, *Costumer Service,* dll). Berikut tampilan halaman *about* dapat dilihat pada Gambar 4.17.



**Gambar 4.17 Tampilan *Aboutpage***

1. Syarat Donor

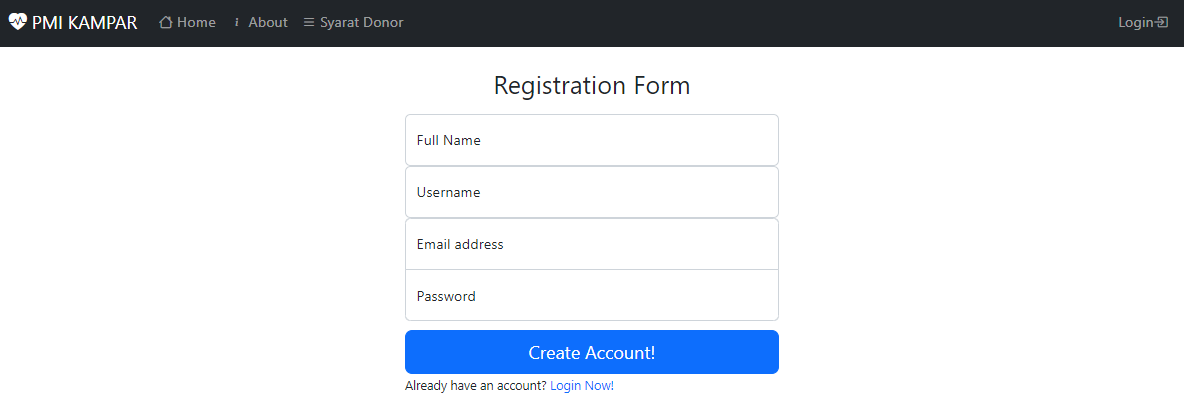
Pada halaman Syarat Donor, terdapat informasi seperti (Syarat Donor Darah, Kelompok Orang yang Tidak Boleh Melakukan Donor Darah, Jenis Jenis Golongan Darah). Berikut tampilan halaman Syarat Donor dapat dilihat pada Gambar 4.18.



**Gambar 4.18 Tampilan Syarat Donor**

1. *Registrationpage*

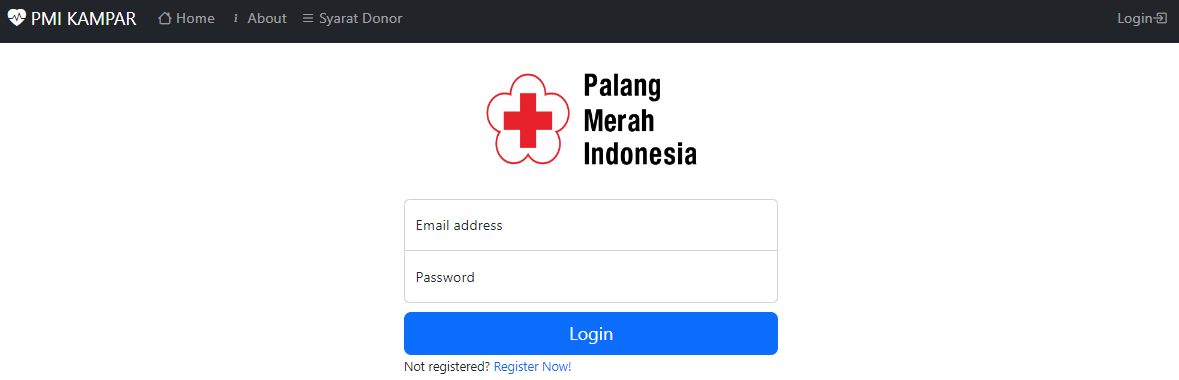
Pada halaman registrasi, terdapat 4 (empat) tabel yang harus diisi yaitu, *Fullname, Username, Email Address, Password*. Berikut adalah tampilan halaman Registrasi yang dapat dilihat pada Gambar 4.19



**Gambar 4.19 Tampilan *Registrationpage***

1. *Loginpage*

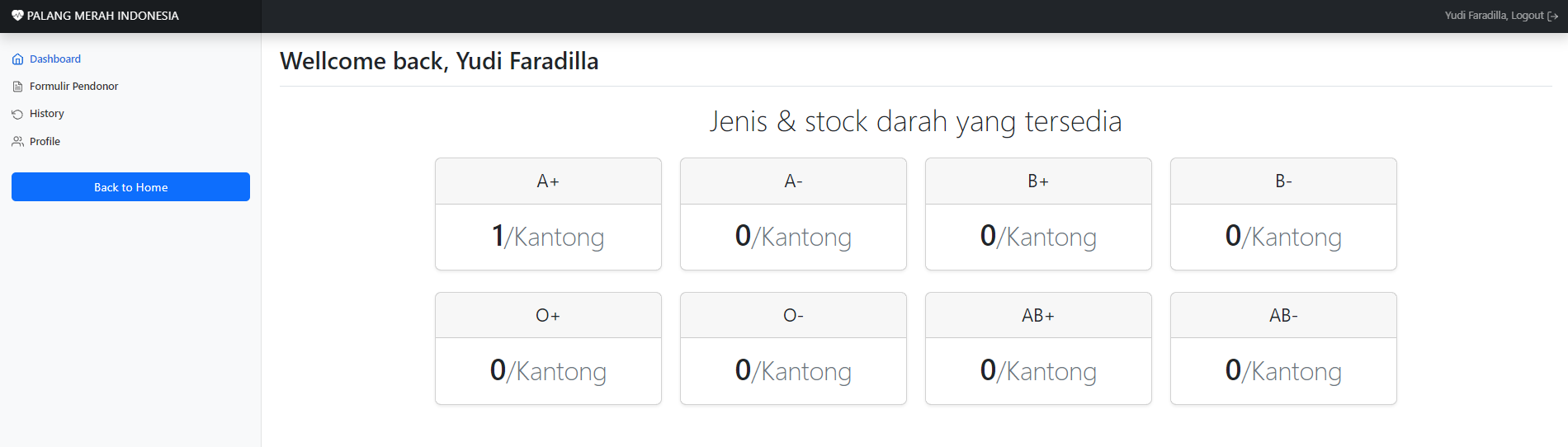
Pada halaman *Login* ini *users* harus memasukkan *Email Address* dan *Password* yang telah terdaftar. Berikut halaman *Login* yang dapat dilihat pada Gambar 4.20.



**Gambar 4.20 Tampilan *Loginpage***

1. *Dashboard*

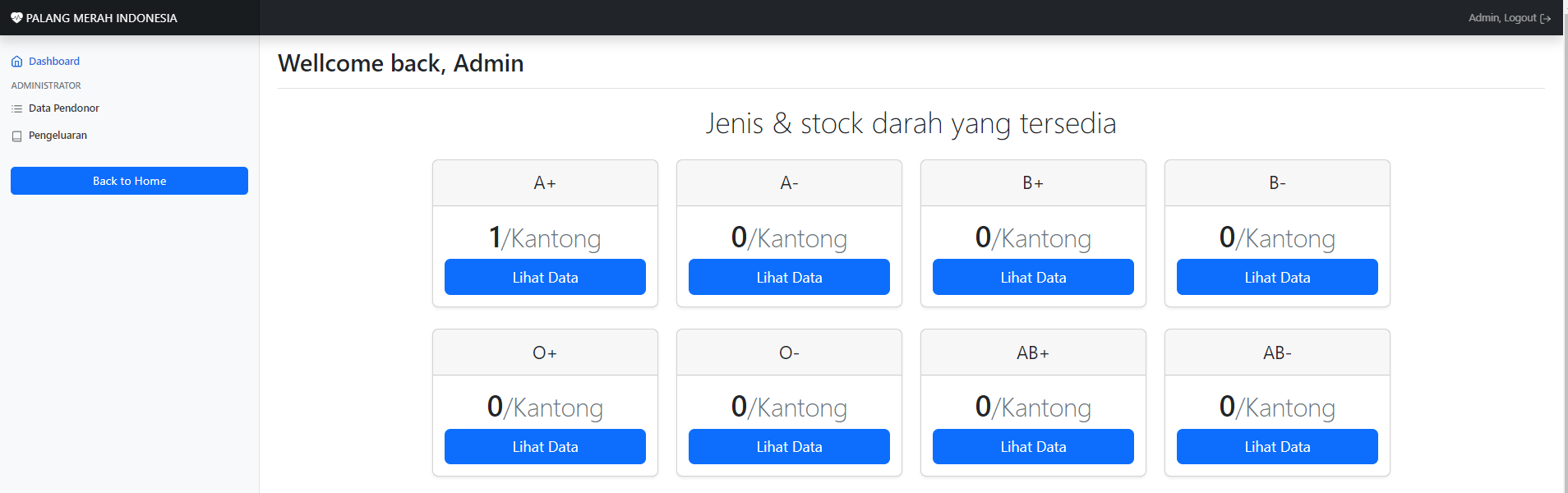
Pada halaman *dashboard*, terdapat informasi tentang Jenis & *Stock* Golongan Darah yang Tersedia di Palang Merah Indonesia (PMI). Berikut tampilan halaman *dashboard* yang dapat dilihat pada Gambar 4.21.



**Gambar 4.21 Tampilan *Dashboard***

1. *Dashboard Admin*

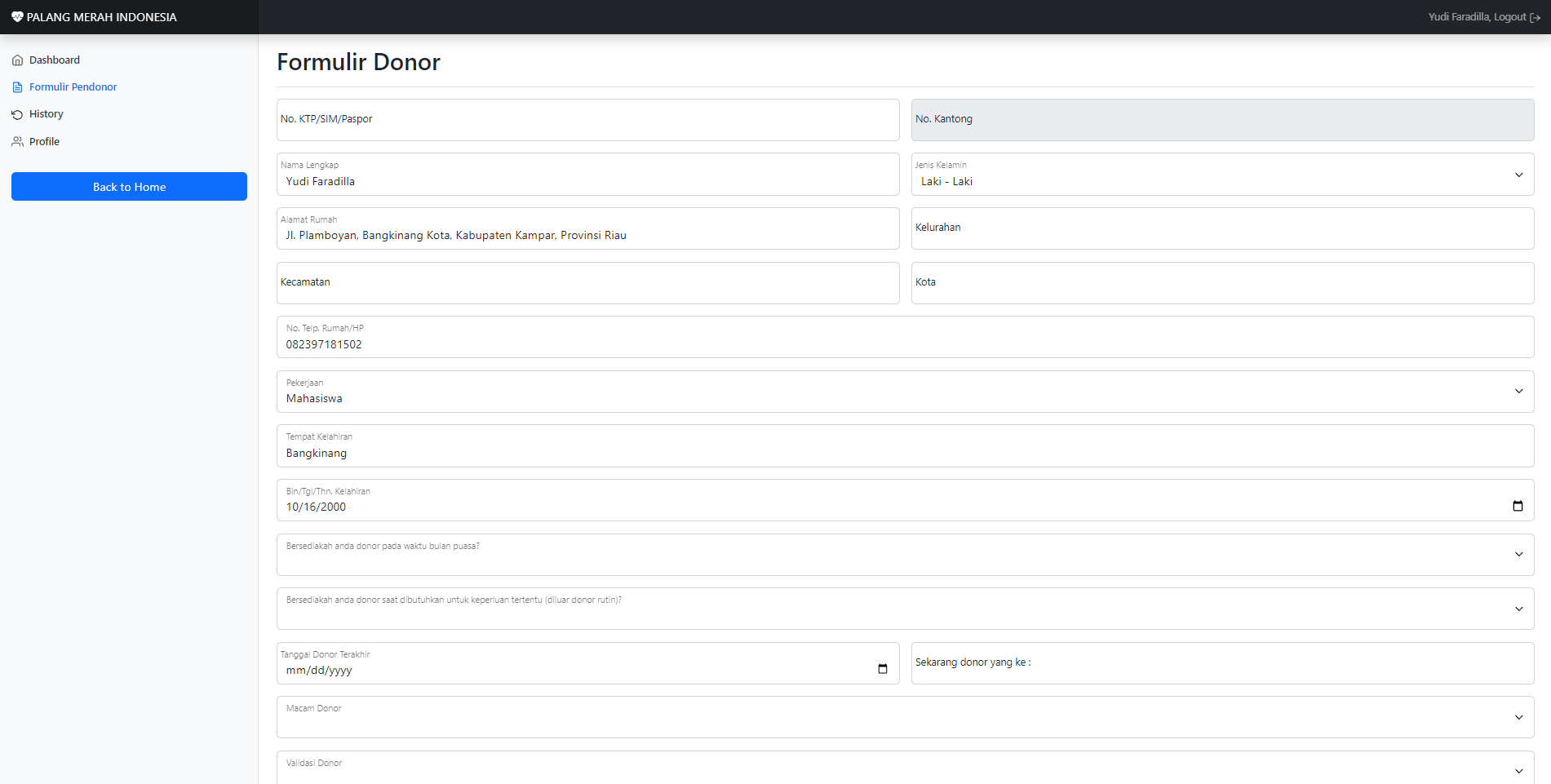
Pada halaman *dashboard admin*, terdapat informasi tentang Jenis & *Stock* Golongan Darah yang Tersedia di Palang Merah Indonesia (PMI), *admin* dapat melihat data secara detail, dan melakukan pengambilan stock darah. Berikut tampilan *dashboard admin* yang dapat dilihat pada gambar 4.22.



**Gambar 4.22 Tampilan *Dashboard Admin***

1. Formulir Donor

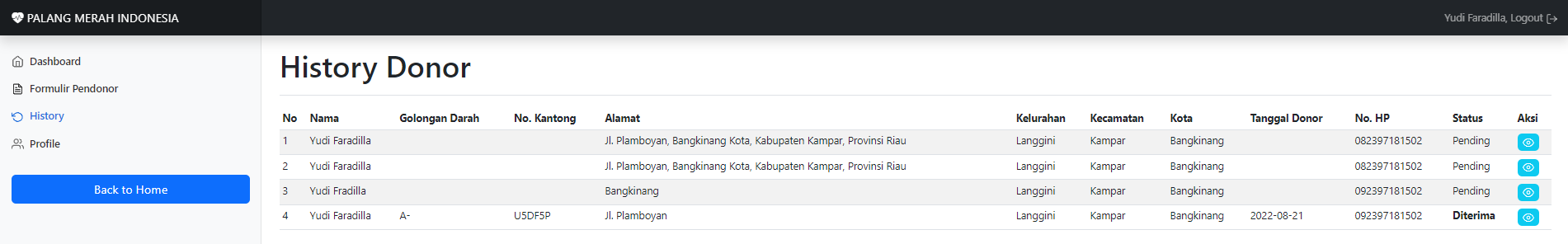
Pada halaman Formulir Donor, Pendonor bisa melakukan pengisian formulir donor seperti No. KTP/SIM/Paspor, Nama Lengkap, Jenis Kelamin, Alamat Rumah, Nomor Telpon, Pekerjaan, Tempat Kelahiran, Tanggal Lahir, , Tanggal Donor Terakhir, dll. Berikut tampilan halaman Formulir Donor yang dapat dilihat pada Gambar 4.23.



**Gambar 4.23 Tampilan Formulir Donor**

1. *Historypage*

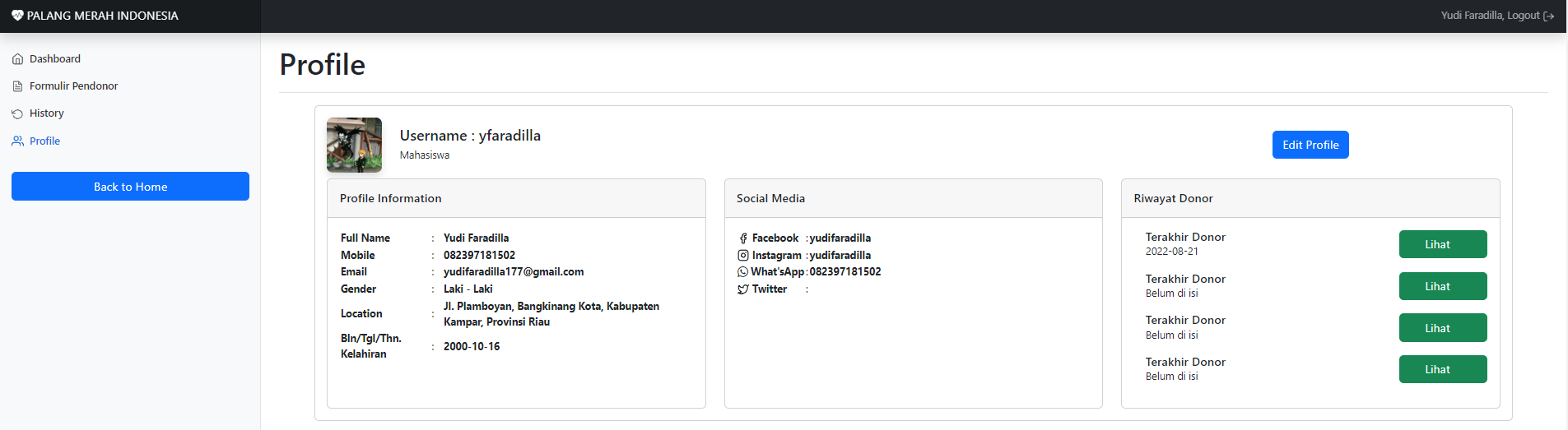
Pada halaman *History* Pendonor bisa melihat status donor dan mencetak Formulir donor yang sudah di-*input* / sudah diisi. Berikut tampilan halaman *History* yang dapat dilihat pada Gambar 4.24.



**Gambar 4.24 Tampilan *Historypage***

1. *Profilepage*

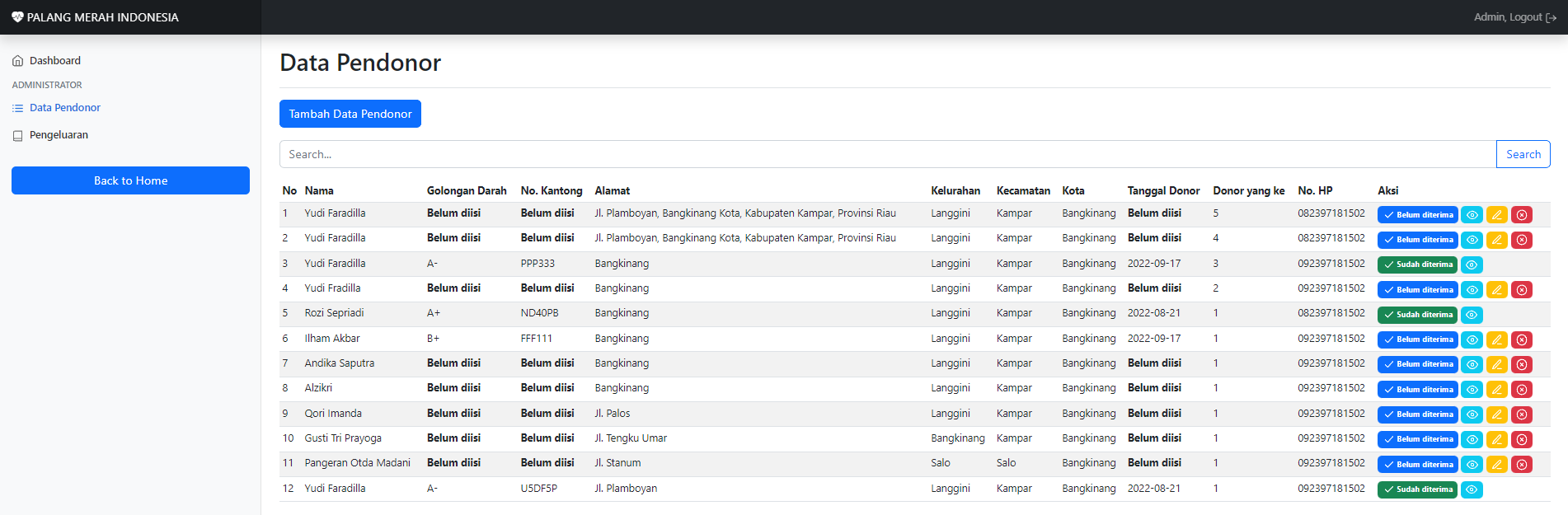
Pada *profilepage* Pendonor dapat mlihat dan meng-*update ­Profile* sesuai dengan yang diinginkan. Berikut tampilan halaman *profilepage* yang dapat dilihat pada gambar 4.25.



**Gambar 4.25 Tampilan *Profilepage***

1. Data Pendonor

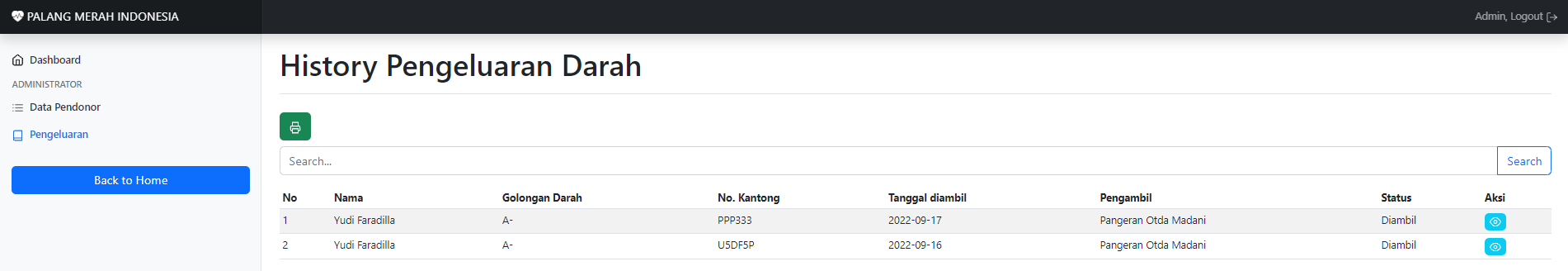
Pada halaman Data Pendonor *Admin* dapat mengolah seperti (Menambahkan, Meng-*update*, Melihat, Mencari, Menerima, dan Menghapus) data dari Formulir Donor. Berikut adalah tampilan dari halaman Data Pendonor yang dapat dilihat pada Gambar 4.26.



**Gambar 4.26 Tampilan Data Pendonor**

1. Pengeluaran

Pada halaman Pengeluaran *Admin* dapat melihat, mencetak, dan mencari riwayat pengeluaran darah di Palang Merah Indonesia (PMI). Berikut adalah tampilan Pengeluaran yang dapat dilihat pada Gambar 4.27.

******

**Gambar 4.27 Tampilan Pengeluaran**

* 1. **Pengujian Sistem**

Sebelum *website* diterapkan kepada pengguna, makan fitur-fitur harus bebas dari kesalahan-kesalahan dan bisa dijalankan. Untuk itu harus diuji terlebih dahulu untuk menentukan kesalahan yang terjadi. Pengujian dilakukan menggunakan tenik pengujian *Black Box* (*Black Box Testing*).

* + - 1. *Black Box Testing*

Pengujian ini bertujuan untuk menemukan kesalah seperti terjadi bug saat dijalankan, fitur tidak merespon seperti yang diinginkan.

Berikut merupakan fitur identifikasi pengujuan pada Aplikasi *Bloodbank* berbasis *Website* Kabupaten Kampar (Palang Merah Indonesia (PMI)) yang akan diuji:

1. *Registrationpage*
2. *Loginpage*
3. Formulir Donor
4. *History*
5. *Profilepage*
6. Pengambilan *Stock* Darah
7. Data Pendonor
8. Pengeluaran

Berikut identifikasi pengujian *website* menggukan *Black Box Testing* dapat dilihat pada Tabel 4.13.

**Tabel 4.13 *Black Box Testing***

| **No** | **Deskripsi Pengujian** | **Prosedur Pengujian** | **Data Masukan** | **Hasil Yang diharapan** | **Kriteria Hasil Evaluasi** | |
| --- | --- | --- | --- | --- | --- | --- |
| **Benar** | **Salah** |
| 1. | Registrasi | Membuka halaman registrasi |  | Sistem menampilkan halaman registrasi | Tanda centang, Wingdings font, kode karakter 252 desimal. |  |
|  |  | Submit registrasi | Input data registrasi | Sistem menyimpan data dan *redirect* ke halaman *login* | Tanda centang, Wingdings font, kode karakter 252 desimal. |  |
| 2 | *Login* | Klik menu *Login* dan isi form *Login* | *Input Email Address* dan *Password* | Berhasil *login* dan *redirect* ke halaman *dashboard* | Tanda centang, Wingdings font, kode karakter 252 desimal. |  |
| 3 | Formulir Donor | Membuka form donor dan mengisi form donor | *Input* semua tabel di form donor | Sistem menyimpan data, dan *redirect* ke halaman *History* | Tanda centang, Wingdings font, kode karakter 252 desimal. |  |
| 4 | *Profile* | Membuka halaman *profile* dan mengganti data *profile* | *Input* semua tabel di tabel form edit profile dan foto *profile* | Sistem menyimpan dan menampilkan data yang telah di-*edit* | Tanda centang, Wingdings font, kode karakter 252 desimal. |  |
| 5 | *Stock* Darah | Membuka halaman *dashboard* dan melihat data berdasar golongan darah yang di pilih, serta melakukan pengambilan darah | *Input* form pengambilan darah | Sistem menampilkan golongan darah sesuai dengan yang di cari dan menyimpan data pengambilan | Tanda centang, Wingdings font, kode karakter 252 desimal. |  |
| 6 | Data Pendonor | Membuka halaman data pendonor dan mengolah data (menambakan, meng-*edit*, melihat, mencari, menerima dan, menghapus) data Formulir Donor | *Input* form Tambah data Pendonor dan form *edit* | Sistem menampilkan semua formulir donor dan bisa melakukan (tambah, edit, lihat, cari, terima, dan hapus) | Tanda centang, Wingdings font, kode karakter 252 desimal. |  |
| 7 | Pengeluaran | Menampilkan semua pengeluaran darah dan mencetak pengeluaran |  | Sistem menampilkan semua pengeluaran dan mecetak pengeluaran ke *format* PDF | Tanda centang, Wingdings font, kode karakter 252 desimal. |  |

Hasil pengujian Aplikasi *Bloodbank* berbasis *Website* Kabupaten Kampar (Palang Merah Indonesia (PMI)) menggunakan *Black Box Testing* pada tabel diatas dapat disimpulkan bahwa kebutuhan fungsional dari sistem ini dengan melakukan fitur sudah berjalan sesuai dengan yang diharapkan dengan tingkat keberhasilan 100%.

**BAB V**

**KESIMPULAN DAN SARAN**

1. **Kesimpulan**

Berdasarkan hasil analisis dan pembahasan data, penulis memperoleh kesimpulan yang dapat diambil dari bab-bab sebelumnya dan penelitian mengenai Aplikasi *Blood Bank* berbasis *Website* Kabupaten Kampar (Palang Merah Indonesia (PMI)) maka dapat disimpulan sebagai berikut:

* + - * 1. Aplikasi *Blood Bank* berbasis *Website* di Kabupaten Kampar (Palang Merah Indonesia (PMI)) mempermudah Unit Donor Darah di Palang Merah Indonesia (PMI) dalam mengolah dan melakukan pencarian data Pendonor,
        2. Aplikasi *Blood Bank* berbasis *Website* Kabupaten Kampar (Palang Merah Indonesia (PMI)) dapat mempermudah dan membantu masyarakat dalam melakukan pencarian informasi seperti (*Stock* darah yang Tersedia, Syarat Donor Darah, Kelompok Orang yang Tidak Boleh Melakukan Donor, dan Jenis Jenis Golongan Darah) di Palang Merah Indonesia (PMI).

1. **Saran**

Saran untuk pengembangan Aplikasi *Blood Bank* berbasis *Website* Kabupaten Kampar (Palang Merah Indonesia (PMI)) sebagai berikut:

Sistem ini tidak menyediakan menu pengambilan *stock* darah secara *online*, disarankan untuk penelitian selanjutnya disediakan menu untuk pengambilan *stock* darah secara *online*,

Sistem ini tidak menyediakan menu *interface* untuk berkomunikasi langsung antara pendonor dan staf, disarankan untuk penelitian selanjutnya disediakan menu *interface* antara pendonor dan staf Palang Merah Indonesia (PMI).

**DAFTAR PUSTAKA**

Aminudin. (2015). Cara Efektif Belajar Framework Laravel. *Ilmu Teknologi Informasi*, *1*(1), 203.

Begawan Raka Sakti, Wina Witanti, A. I. H. (2021). *Sistem Informasi Bank Darah dengan Location Based Service untuk Meningkatkan Efisiensi Pencarian Golongan Darah (Studi Kasus: UTD PMI Cimahi)*.

Bonnie Soeherman, M. P. (2008). *Designing information system*. Elex Media Komputindo.

Booch, G., Rumbaugh, J., & Jacobson, I. (2005). *The Unified Modeling Language*. Addison Wesley Professional.

Brady, M., & Loonam, J. (2010). Exploring the use of entityrelationship diagramming as a technique to support grounded theory inquiry. *Qualitative Research in Organizations and Management: An International Journal*, *5*(3), 224–237. https://doi.org/10.1108/17465641011089854

Cholifah, W. N., Yulianingsih, Y., & Sagita, S. M. (2018). Pengujian Black Box Testing pada Aplikasi Action & Strategy Berbasis Android dengan Teknologi Phonegap. *STRING (Satuan Tulisan Riset Dan Inovasi Teknologi)*, *3*(2), 206. https://doi.org/10.30998/string.v3i2.3048

Depkes RI. (2009). *Donor Darah, Hidup Sehat*.

Dhimas Bayususetyo, Rukun Santoso, T. (2017). *KLASIFIKASI CALON PENDONOR DARAH MENGGUNAKAN METODE NAÏVE BAYES CLASSIFIER (Studi Kasus : Calon Pendonor Darah di Kota Semarang)*. *6*, 193–200.

Fathansyah. (2015). *Basis Data*. Informatika Bandung.

Findawati, Y. (2018). Buku Ajar Rekayasa Perangkat Lunak. In *Buku Ajar Rekayasa Perangkat Lunak*. https://doi.org/10.21070/2018/978-602-5914-09-6

Gulo, W., & Hardiwati, Y. (2002). *Metodologi penelitian*. Grasindo.

Gustaman, A. B., & Suji. (2013). Kualitas Pelayanan Pendonoran Darah pada Unit Donor Darah Palang Merah Indonesia Kabupaten Jember ( Service Quality of blood Transfusion at Unit Donor Darah Palang Merah Indonesia in Jember Regency ). *Universitas Jember (UNEJ )*, 1–5. https://repository.unej.ac.id/bitstream/handle/123456789/58987/Arief Gustaman.pdf?sequence=1

Gustaman, R. A., Hidayat, E. W., & Hiron, N. (2016). Sistem Informasi Pelayanan Donor Darah Berbasis Web ( Studi Kasus : Pmi Tasikmalaya ). *Seminar Nasional Teknologi Informasi Dan Multimedia*, *X*(2), 6–7.

Kemenkes RI. (2014). *Situasi Donor Darah di Indonesia*.

M. Shalahuddin, Rosa A.S. (2014). *Rekayasa Perangkat Lunak: Terstruktur dan Berorientasi Objek*. Informatika Bandung.

MADCOMS. (2016). *Manajemen sistem jaringan komputer dengan mikrotik routeros*. CV. Andi Offset ( Penerbit Andi).

Mulyanto, A. (2008). *Sistem Informasi Konsep dan Aplikasi*. Pustaka Belajar.

Nahado, M. R. (2015). *Bumbu Bumbu Membuat Website: Cara Membuat Website Cantik dan Mudah* (F. D. Mulya (ed.)). Bang Roy Han.

Parulian, O. S. (2017). *3 Days With Mysql for your Applications*.

Permenkes RI. (2015). *Standar Pelayanan Transfusi Darah*.

Prayitno, H. (2005). Identifikasi Problema Pelayanan Kesehatan pada Unit Transfusi Darah (UTD) PMI Cabang Jember dan Upaya Pemecahannya. *Jurnal Aspirasi*, 88.

Priyanto Hidayatullah, Jauhari Khairul Kawistara. (2017). *Pemrograman WEB*. Informatika.

Rachman Andi, R., Beny, & Fernando, E. (2017). Perancangan E-Commerce Berbasis Website Pada Toko Dunia Palembang. *Jurnal Ilmiah Processor*, *12*(2), 1102–1117.

Rika Andriyani, Ani Triana, Widya Juliarti. (2015). *Buku Ajar Biologi Reproduksi dan Perkembangan*. Deepublish.

Robert J. Verzello/John Reuter III. (1982). *Data Processing: System and concepts*. McGraw-Hill Kogakusha.

Ronald, A. S. (2004). *Tinjauan Klinis Hasil Pemeriksaan Laboratorium*. Gramedia Pustaka Utama.

Sadikin, M. (2001). *Biokimia Darah*. Widya Medika.

Sidratul munti, N. yona. (2019). Perancangan Sistem Pakar Diagnosa Penyakit Lupus Eritmatosus Sistem(LES) Dengan Metode Forward Chaining Menggunakan Pemrograman PHP dan MySQL. *Jurnal Fasilkom*, *9*(2), 407–428. https://doi.org/10.37859/jf.v9i2.1404

Simarmata, J. (2010). *Rekayasa Web* (N. WK (ed.)). ANDI.

Sofia, Budhi Prianto, H. (2010). *Panduan Mahir Akses Internet* (D. A. Santoso (ed.)). Kriya Pustaka.

Sri Mulyani. (2016). *Sistem Informasi Manajemen Rumah Sakit: Analisis dan Perancangan*.

Sugiyono. (2005). *Memahami Penelitian Kunlitatif*. CV.Alfabeta.

Sunyoto, A. (2007). *AJAX Membangun Web dengan Teknologi Asynchronouse Javascript dan XML*. ANDI.

Susilo, M. (2018). Rancang Bangun Website Toko Online Menggunakan Metode Waterfall. *InfoTekJar (Jurnal Nasional Informatika Dan Teknologi Jaringan)*, *2*(2), 98–105. https://doi.org/10.30743/infotekjar.v2i2.171

Tim EMS. (2016). *All in one WEB programming*. Kompas Gramedia.

Trianto, E. A., & Yulianeu, A. (2018). Perancangan Sistem Informasi Pembayaran Abodemen di UPTD Pasar Rajadesa. *Jumantaka*, *1*(1), 11–20. https://bit.ly/2w8Qmlm

Vilma, A., Azriadi, E., & Gusman, D. (2022). *SISTEM APLIKASI PEMESANAN BIBIT IKAN BERBASIS WEB ( PROGRAMMING ) Studi Kasus ( Kabupaten Kampar ) Pendahuluan*. *1*(1), 19–28.

Wahana Komputer. (2014). *Mobile App Development With PhoneGap*. Andi Publisher.

Whitten, J. L., Dittman, K. C., & Bentley, L. D. (2004). *Systems Analysis and Design Methods*. McGraw-Hill.

Zaki, SmitDev Community, A. (2008). *36 menit belajar komputer PHP dan MySQL*. Elex Media Komputindo.