



BAB II
TINJAUAN PUSTAKA
DAN LANDASAN TEORI

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Penelitian sebelumnya dilakukan oleh Iqbal Ramadhani Mukhlis (2022) dengan judul “Sistem Informasi Donor Darah Berbasis *Website* Menggunakan *Framework* CodeIgniter Pada Unit Transfusi Darah (UTD) Palang Merah Indonesia Lumajang”. Permasalahan yang diangkat dalam penelitian ini adalah kesulitan dalam mencari informasi mengenai ketersediaan darah dan data donor darah belum dikelola dengan baik serta diakses dengan cepat. Sistem ini menggunakan metode pengembangan *waterfall*, bahasa pemrograman PHP dan *framework* CodeIgniter. Hasil dari penelitian ini adalah sistem informasi berbasis *website* yang dapat mempermudah proses penyampaian informasi stok darah dan mengelola data donor yang dapat diakses dengan cepat [4].

Penelitian yang lainnya telah dilakukan oleh Rizal Rachman (2019) dengan judul “Pengembangan Sistem Informasi Donor Darah Berbasis *Website* Pada Unit Transfusi Darah Kabupaten Sumedang”. Permasalahan yang diangkat dalam penelitian ini adalah pencatatan transaksi yang masih bersifat manual dan menggunakan kertas sehingga memungkinkan besarnya risiko terjadinya *human error*, informasi mengenai persediaan darah dan kegiatan donor darah masih menggunakan media telepon. Sistem ini menggunakan metode pengembangan *waterfall*, bahasa pemrograman PHP, Javascript dan CSS serta *database* menggunakan MySQL. Hasil dari penelitian ini adalah suatu sistem informasi berbasis *website* yang dapat memudahkan pengguna sistem dalam melakukan kegiatan donor darah dan meminimalisir pengeluaran biaya [5].

Selanjutnya penelitian lainnya dilakukan oleh Yusuf Sumaryana (2019) dengan judul “(EBSIS) *Electronic Blood Stock Information System* Sebagai Pusat Informasi Stok Darah Pada Unit Transfusi Darah (UTD) Kabupaten Serang”. Permasalahan yang diangkat pada penelitian ini adalah proses pendistribusian stok darah dan kurangnya informasi mengenai kegiatan donor darah secara massal kepada masyarakat. Sistem ini menggunakan metode pengembangan *waterfall* dan bahasa pemrograman PHP. Hasil dari penelitian ini adalah sistem yang dapat

memberikan informasi stok darah dan kegiatan donor darah massal guna memenuhi kebutuhan stok darah di UTD [6].

Penelitian lainnya telah dilakukan oleh La Raufun dan Wa Ode Desi Angraini (2019) dengan judul “Rancang Bangun Aplikasi Sistem Informasi Ketersediaan Darah Pada Palang Merah Indonesia Kabupaten Buton”. Permasalahan yang diangkat dalam penelitian ini adalah manajemen informasi belum memanfaatkan teknologi informasi mengenai informasi stok darah sehingga mengalami keterlambatan informasi kepada masyarakat. Hasil dari penelitian ini adalah suatu aplikasi yang dapat memberikan informasi mengenai jumlah stok darah dan jumlah pendonor selama per tahun di PMI Kabupaten Buton [7].

Selanjutnya penelitian lainnya ditulis oleh Agnes Maria Polina dan Fenty (2018) dengan judul “Sistem Informasi Persediaan Darah Berbasis *Website* (Studi Kasus di PMI Kota Magelang)”. Permasalahan yang diangkat dalam penelitian ini adalah masyarakat yang mengalami keterlambatan dalam memperoleh informasi tentang persediaan darah di PMI. Metode pengembangan yang digunakan adalah *waterfall*, bahasa pemrograman PHP, dan *database* menggunakan MySQL. Hasil dari penelitian ini adalah sistem informasi persediaan darah berbasis *website* untuk menampilkan persediaan darah dengan update di Kota Magelang [8].

Tabel 2. 1 Penelitian Terdahulu

Komponen	Penelitian				
	Iqbal Ramadhani Mahlis	Rizki Iscahmana	Yuar Samudryana	La Rantien dan Wa Ode Desti Agrianti	Agnes Maria Polina dan Ferry
Judul Penelitian	Sistem Informasi Donor Darah Berbasis Sistem Website Manajemen Farmasi Codingan Pada Unit Transfusi Darah (UTD) Palang Merah Indonesia Lampung	Pengembangan Sistem Informasi Donor Darah Berbasis Web Pada Unit Transfusi Darah Kabupaten Sumsel	(EBIS) Electronic Blood Stock Information System Sebagai Pusat Informasi Stock Darah Pada Unit Transfusi Darah (UTD) Kabupaten Serang	Rancang Bangun Aplikasi Sistem Informasi Kesehatan Masyarakat Kecamatan Kedebe Kabupaten Bantou	Sistem Informasi Persediaan Darah Berbasis Website (Studi Kasus di PMI Kota Magelang)
Pemmasalahan	Kesulitan dalam mencari informasi mengenai ketersediaan darah dan data donor darah belum dikelola dengan baik serta diakses dengan cepet	Persialat transaksi yang masih bersifat manual dan menggunakan kertas sehingga mengakibatkan besarnya risiko terjadinya <i>human error</i> , informasi mengenai persediaan darah dan kegiatan donor darah masih menggunakan media telepon	Proses pendistribusian stok darah dan kurangnya informasi mengenai kegiatan donor darah secara masal kepada masyarakat	Memangenen informasi belum memanfaatkan teknologi informasi mengenai informasi stok darah sehingga mengalami keterbatasan informasi kepada masyarakat	Masyarakat yang mengalami keterbatasan informasi dalam memperoleh informasi tentang persediaan darah di PMI
Metode	Wawancara	Wawancara	Wawancara	Wawancara	Wawancara
Penggunaan Sistem Tahun Penelitian	<i>Framework</i> Codingan; Bahasa pemrograman PHP; 2012	Bahasa pemrograman PHP, Javascrpt, CSS, <i>database</i> MS-SQL; 2019	Bahasa pemrograman PHP; 2019	Bahasa pemrograman PHP; 2019	Bahasa pemrograman PHP, <i>database</i> MS-SQL; 2018
Hasil	Sistem informasi berbasis website yang dapat mempermudah proses penyampaian informasi stok darah dan mengelola data donor yang dapat diakses dengan cepet	Sistem informasi berbasis website yang dapat memudahkan pengguna sistem dalam melakukan kegiatan donor darah dan meminimalisir pengeluaran biaya	Sistem yang dapat memberikan informasi data dan kegiatan donor darah masal masyarakat kebutuhan stok darah di UTD	Suatu aplikasi yang dapat memberikan informasi mengenai jumlah stok darah dan jumlah pendonor selama per tahun di PMI Kabupaten Bantou	Sistem informasi persediaan darah berbasis website untuk mempermudah persediaan darah dengan update di Kota Magelang

Dari penelitian yang akan dilakukan memiliki perbedaan dengan penelitian-penelitian sebelumnya. Perbedaan tersebut diantaranya yaitu penggunaan *framework* Laravel dalam pembangunan sistem ini yang mana menggunakan bahasa pemrograman PHP (*Hypertext Preprocessor*), Javascript, dan CSS serta MySQL sebagai aplikasi *database*. Terdapat perbedaan yang terletak pada menu pendaftaran donor darah yang dapat diakses kapanpun oleh pendonor, menu untuk mengelola pencatatan donor darah dari proses *medical check up* sampai stok darah, menu mengakses formulir dan mengelola permintaan darah untuk Instansi Medis yang sedang membutuhkan darah dilengkapi dengan notifikasi *email*.

2.2 Landasan Teori

2.2.1 Sistem Informasi

Sistem merupakan sebuah jaringan kerja dari prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan sasaran tertentu. Informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengelolaan transaksi harian, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi tertentu dengan laporan-laporan yang dibutuhkan [3].

Sistem informasi merupakan suatu kombinasi antara perangkat lunak (*software*), perangkat keras (*hardware*), manusia (*brainware*), jaringan komputer, komunikasi data, dan basis data yang dapat mengumpulkan, mengubah serta menyebarkan suatu informasi dalam organisasi tertentu [9]. Sistem informasi terdiri dari data sebagai input lalu menghasilkan laporan sebagai *output* sehingga dapat diterima oleh sistem yang lain dan kegiatan strategi dalam sebuah organisasi dalam melakukan tindakan atau mengambil keputusan.

Sistem informasi memiliki tujuan guna menciptakan suatu wadah komunikasi yang dipandang efisien dalam bidang bisnis. Adapun komponen-komponen dari sistem informasi adalah sebagai berikut [10]:

1. Blok masukan berupa metode-metode dan media untuk menangkap data yang akan dimasukkan, dapat berupa dokumen-dokumen dasar.
2. Blok model terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data input dan data tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan.

3. Blok keluaran merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.
4. Blok teknologi merupakan kotak alat dalam sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian sistem secara keseluruhan.

2.2.2 Palang Merah Indonesia (PMI)

PMI merupakan organisasi yang memiliki sifat independen dan melaksanakan tugasnya di bidang sosial kemanusiaan. Tugas pokok dari PMI sendiri telah diatur dalam Peraturan Pemerintah No. 18/ 1980 Bab IV, pasal 6, ayat (1) yaitu “Pengelolaan dan pelaksanaan usaha transfusi darah ditugaskan kepada Palang Merah Indonesia atau instansi lain yang ditetapkan oleh Menteri Kesehatan”. Tugas pokok lain dari PMI seperti kesiapsiagaan bantuan dan penanggulangan bencana, pelatihan pertolongan pertama untuk sukarelawan, pelayanan kesehatan dan kesejahteraan masyarakat. Dalam hal penyediaan dan pengolahan darah di perkotaan, PMI membuat suatu unit khusus untuk melaksanakan tugas tersebut yaitu Unit Donor Darah atau UDD [11].

A. Unit Donor Darah (UDD)

Unit Donor Darah (UDD) didirikan oleh PMI untuk melaksanakan tugas khusus dalam hal pelayanan darah, salah satunya adalah UDD PMI Cilacap. Pelayanan darah yang difasilitasi oleh UDD PMI meliputi penyediaan darah, pengolahan darah, dan pelayanan donor darah. Unit Donor Darah (UDD) PMI merupakan sebuah lembaga yang menyelenggarakan, mengatur, memfasilitasi kegiatan donor darah, dan persediaan darah [12].

B. Pelayanan Darah

Menurut Pasal 86 Undang-Undang Nomor 36 Tahun 2009 tentang Kesehatan dan PP No. 7 Tahun 2011 tentang Pelayanan Darah, definisi dari pelayanan darah adalah upaya pelayanan kesehatan yang memanfaatkan darah manusia sebagai bahan dasar dengan tujuan kemanusiaan dan tidak untuk tujuan komersial. Pelayanan penyedia darah di Indonesia dilaksanakan oleh Bank Darah Rumah Sakit (BDRS) dan Unit Tranfusi Darah (UTD) atau Unit Donor Darah (UDD).

C. Donor Darah

Donor darah merupakan suatu proses yang dilakukan untuk mengambil darah dari seseorang (pendonor) tanpa adanya paksaan atau sukarela, selanjutnya darah yang sudah diambil disimpan di bank darah yang disebut sebagai persediaan darah, kemudian darah tersebut akan digunakan untuk kepentingan tranfusi darah [17].

Manfaat dari donor darah antara lain yaitu dapat meningkatkan produksi sel darah merah, membantu menjaga kesehatan jantung, dan melancarkan sirkulasi darah. Tujuan utama dari kegiatan donor darah yaitu menambahkan darah ke tubuh korban yang mengalami kekurangan darah karena sebab tertentu, contohnya pendarahan ketika kecelakaan, ibu melahirkan, dan operasi [18]. Donor darah sendiri dibedakan menjadi empat macam, yaitu [13]:

1. Donor Darah Sukarela

Donor darah sukarela adalah kondisi dimana seseorang mendonorkan darahnya dengan sukarela tanpa tahu siapa yang akan menerima darah tersebut. Donor darah jenis ini tidak dibayar, karena niat si pendonor sendiri secara sukarela.

2. Donor Darah Pengganti

Donor darah pengganti adalah kondisi dimana seseorang mendonorkan darahnya kepada orang lain yang merupakan anggota keluarga atau kerabat, pendonor sudah mengetahui siapa yang akan menerima darah tersebut. Donor darah jenis ini biasanya tidak dibayar oleh unit donor darah (UDD) atau Instansi medis, tetapi kemungkinan akan diberi imbalan oleh keluarga pasien.

3. Donor Bayaran

Donor bayaran adalah orang yang memberikan darah dengan imbalan pembayaran atau keuntungan lainnya, yang digunakan untuk memenuhi kebutuhan hidup pokok atau dapat dijual atau ditukar dengan uang tunai, atau dipindahkan kepada orang lain.

4. Donor Plasma Khusus

Donor plasma khusus adalah seseorang yang melakukan *plasmapheresis* dengan tujuan menjadi pendonor untuk memenuhi kebutuhan bahan baku dalam pembuatan derivat plasma melalui proses fraksionasi. Fraksionasi plasma adalah proses pemisahan komponen-komponen plasma darah menjadi berbagai produk yang berguna, seperti faktor pembekuan, imunoglobulin, albumin, dan produk lainnya. Donor plasma khusus berperan penting dalam menyediakan bahan baku ini untuk keperluan medis dan farmasi.

D. Uji Cocok Silang Serasi

Pemeriksaan uji silang serasi adalah suatu prosedur untuk memastikan kesesuaian darah antara pasien dan donor sebelum transfusi darah dilakukan. Pemeriksaan ini bertujuan untuk menentukan apakah antigen dari sel darah donor cocok dengan antibodi yang ada di serum pasien, serta apakah antigen dari sel darah pasien cocok dengan antibodi yang ada di serum donor. Metode yang digunakan dalam pemeriksaan uji silang serasi adalah *gel test*, di mana suspensi sel darah dan serum atau plasma dari pasien dan donor ditempatkan dalam *microtube*. Jika darah kompatibel, hasil uji silang akan menunjukkan hasil jernih pada *gel test* yang menandakan hasil negatif, sedangkan jika darah tidak cocok, hasil uji silang akan menunjukkan hasil keruh pada *gel test* yang menandakan hasil positif. Hal ini dilakukan untuk memastikan keselamatan dan kelayakan transfusi darah bagi pasien [14].

2.2.3 Darah

Istilah darah berasal dari bahasa Yunani yaitu *hemo*, *hemato*, dan *haima*. Darah merupakan cairan yang terdapat pada semua makhluk hidup kecuali tumbuhan yang memiliki fungsi yaitu mengirimkan zat dan oksigen yang dibutuhkan oleh jaringan tubuh, pertahanan tubuh manusia terhadap bakteri atau virus, dan mengangkut bahan kimia hasil metabolisme.

Darah manusia yaitu cairan yang ada di dalam tubuh manusia yang berfungsi untuk mengangkut oksigen yang dibutuhkan oleh seluruh sel-sel tubuh. Selain itu, darah juga berfungsi untuk menyuplai jaringan tubuh dengan nutrisi dan mempertahankan tubuh dari berbagai macam penyakit [15].

A. Golongan Darah

Golongan darah adalah suatu ciri khusus darah dari seorang individu karena adanya perbedaan jenis karbohidrat dan protein yang dimiliki pada permukaan membran sel darah merah. Bisa juga dikatakan, golongan darah ditentukan oleh jumlah zat (antigen) yang terkandung dalam sel darah merah individu. Penggolongan darah ABO ditentukan berdasarkan jenis antigen dan antibodi yang terkandung dalam darah [16]. Macam-macam golongan darah antara lain:

1. Golongan Darah A

Individu dengan golongan darah A memiliki sel darah merah dengan antigen A di permukaan membran selnya dan menghasilkan antibodi terhadap antigen B dalam serum darahnya. Dengan demikian, orang

dengan golongan darah A negatif hanya dapat menerima darah dari orang lain dengan golongan darah A negatif atau O negatif.

2. Golongan Darah B

Individu dengan golongan darah B memiliki sel darah merah dengan antigen B di permukaan membrane selnya dan menghasilkan antibodi terhadap antigen A dalam serum darahnya. Dengan demikian, orang dengan golongan darah B negatif hanya dapat menerima darah dari orang lain dengan golongan darah B negatif atau O negatif.

3. Golongan Darah O

Individu dengan golongan darah O memiliki sel darah tanpa antigen, tapi memproduksi antibodi terhadap antigen A dan B. Oleh karena itu, orang dengan golongan darah O negatif dapat mendonorkan darahnya pada orang lain dengan golongan darah ABO apapun dan disebut donor universal. Namun, orang dengan golongan darah O negatif hanya dapat menerima darah dari sesama O negatif.

4. Golongan Darah AB

Individu dengan golongan darah AB memiliki sel darah merah dengan antigen A dan B serta tidak menghasilkan antibodi terhadap antigen A maupun B dalam serum darahnya. Oleh karena itu, orang dengan golongan darah AB positif dapat menerima darah dari orang dengan golongan darah ABO apapun dan disebut resipien universal. Namun, orang dengan golongan darah AB positif tidak dapat mendonorkan darah kecuali pada sesama AB positif.

B. Produk Darah

Produk darah adalah setiap substansi terapeutik yang dibuat dari darah manusia. Dari produk darah dibuat menjadi komponen darah. Komponen darah berawal dari darah lengkap (*Whole Blood*) merupakan darah dari donor yang dikumpulkan dalam sebuah wadah berisi larutan pengawet antikoagulan, dan belum dipisahkan komponennya [16]. Adapun jenis komponen darah adalah sebagai berikut [17]:

1. Darah Lengkap (*Whole Blood*)

Digunakan untuk transfusi tanpa pengolahan lebih lanjut. WB merupakan bahan baku untuk pengolahan menjadi komponen darah lain.

2. *Packed Red Cell* (PRC)

Diperoleh dari WB dengan membuang sebagian besar volume plasma dari darah lengkap. PRC mungkin mengandung sejumlah besar leukosit dan trombosit tergantung metoda sentrifugasi.

3. *Fresh Frozen Plasma* (FFP)

Diperoleh dari WB yang ditampung ke dalam sistem kantong darah steril dengan kantong transfer yang terintegrasi. FFP dipisahkan setelah sentrifugasi dengan putaran cepat dari WB atau *platelet rich* plasma dan dibekukan dengan cepat hingga ke intinya yang akan menjaga fungsi dari faktor koagulasi labil.

4. *Cryoprecipitate/ Anti Hemophilic Factor* (AHF)

Komponen darah yang berisi fraksi krioglobulin plasma. Diperoleh dari FFP asal WB yang diproses lebih lanjut dan dikonsentrasikan.

5. *Thrombocyt Concentrate* (TC)

Komponen darah yang berisi fraksi krioglobulin plasma. Diperoleh dari FFP asal WB yang diproses lebih lanjut dan dikonsentrasikan.

C. **Rhesus Darah**

Dalam sistem penggolongan darah, terdapat pula sistem penggolongan darah rhesus (faktor Rh), yaitu penggolongan darah yang hasilnya positif atau negatif setelah mengetahui penggolongan darah A, B, AB, O. Seseorang yang memiliki rhesus positif maka dia memiliki antigen rhesus (faktor Rh) dalam sel darah merahnya, dan seorang yang memiliki rhesus negatif tidak ditemukan adanya antigen rhesus (faktor Rh) dalam sel darah merahnya.

Berdasarkan data Kemenkes RI tahun 2018, di Indonesia jumlah pemilik rhesus negatif kurang dari 1% penduduk ini dari atau sekitar 1,2 juta orang. Pasangan dengan golongan rhesus berbeda akan menyebabkan terbentuk antibodi yang berbeda, hal ini akan mengancam nyawa janin yang dikandung oleh ibu karena ketidakcocokan golongan darah rhesus ibu dan janin. Data dari Instansi Medis Umum Vancouver Columbia menyatakan bahwa golongan darah ibu yang paling banyak mengalami abortus spontan yaitu golongan darah O, 52.0%, kemudian golongan darah A, 37.1%, B, 9.2%, AB, 1.7%, yang disebabkan karena ketidakcocokan rhesus ibu dengan rhesus janin [18].

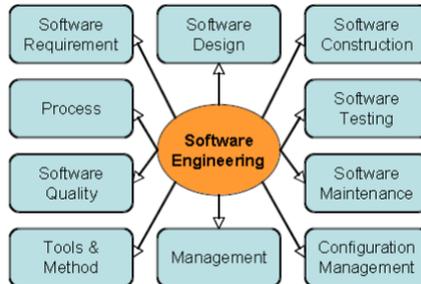
2.2.4 **Rekayasa Perangkat Lunak**

Perangkat lunak (*Software*) adalah sebuah perintah program dalam sebuah komputer, yang apabila dieksekusi oleh *user* akan memberikan

fungsi dan untuk bekerja seperti yang diharapkan oleh user. Rekayasa Perangkat Lunak (RPL) merupakan proses kegiatan perangkat lunak guna mengembangkan, memelihara, dan membangun kembali dengan menggunakan prinsip rekayasa untuk menghasilkan perangkat lunak yang dapat bekerja lebih efektif dan efisien untuk *user*. Rekayasa perangkat lunak lebih fokus pada bagaimana membuat perangkat lunak yang memenuhi kriteria berikut [19]:

1. Dapat terus dipelihara setelah perangkat lunak selesai dibuat seiring berkembangnya teknologi dan lingkungan (*maintainability*)
2. Dapat diandalkan dengan proses bisnis yang dijalankan dan perubahan yang terjadi (*dependability* dan *robust*)
3. Efisien dari segi sumber daya dan penggunaan
4. Kemampuan untuk dipakai sesuai dengan kebutuhan (*usability*)

Adapun ruang lingkup dari RPL dapat dilihat pada Gambar 2.1 sebagai berikut:



Gambar 2. 1 Ruang Lingkup RPL

Berikut merupakan penjelasan mengenai ruang lingkup RPL:

1. *Software Requirement* merupakan kegiatan yang dilakukan untuk mengidentifikasi dan menganalisis kebutuhan perangkat lunak. Hasil akhir tahapan ini adalah spesifikasi dan model perangkat lunak.
2. *Software Design* adalah tahapan perancangan arsitektur, komponen, antar muka, dan karakteristik lain dari perangkat lunak.
3. *Software Construction* berhubungan dengan detail pengembangan perangkat lunak, termasuk algoritma, pengkodean, pengujian dan pencarian kesalahan.
4. *Software Testing* meliputi pengujian pada keseluruhan perilaku perangkat lunak.

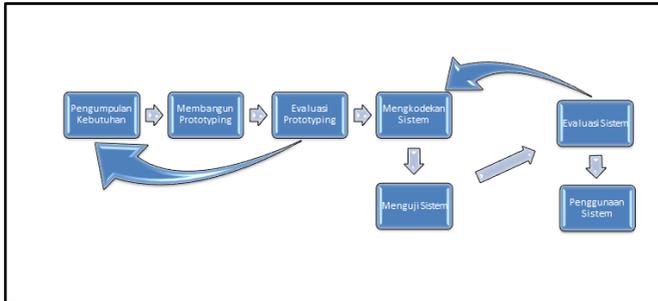
5. *Software Maintenance* mencakup upaya-upaya perawatan ketika perangkat lunak telah dioperasikan.
6. *Software Configuration Management* berhubungan dengan usaha perubahan konfigurasi perangkat lunak untuk memenuhi kebutuhan tertentu.
7. *Software Engineering Management* berkaitan dengan pengelolaan dan pengukuran RPL, termasuk perencanaan proyek perangkat lunak.
8. *Software Engineering Tools and Methods* mencakup kajian teoritis tentang alat bantu dan metode RPL.
9. *Software Engineering Process* berhubungan dengan definisi, implementasi pengukuran, pengelolaan, perubahan dan perbaikan proses RPL.
10. *Software Quality* menitik beratkan pada kualitas dan daur hidup perangkat lunak.

A. Metode Pengembangan Sistem

Pada rekayasa perangkat lunak, terdapat beberapa metode yang digunakan dalam pengembangan sistem, salah satu metode yang digunakan dalam pengembangan sistem adalah metode *prototype*. Metode *prototype* ini diawali dengan dikumpulkannya kebutuhan pengguna dari sistem yang akan dikembangkan, setelah itu dibuat sebuah rancangan kilat, yang kemudian dilakukan evaluasi oleh klien.

Metode *prototype* merupakan metode pengembangan *software* yang memungkinkan terdapat interaksi antara *software developer* dengan pemakai sistem, sehingga bisa menangani ketidaksesuaian antara *developer* dengan pemakai [20]. *Prototype* bisa dipakai sebagai penyambung ketidakpahaman klien tentang hal teknis dan membantu menjelaskan rincian kebutuhan yang dibutuhkan klien kepada *developer* [21].

Metode *prototype* adalah suatu metode yang memungkinkan pengembang membuat suatu model perangkat lunak, metode *prototype* ini bisa diterapkan apabila pihak klien tidak dapat menjelaskan informasi secara maksimal perihal kebutuhan yang diinginkannya [22]. Dengan berdasar pada pendapat para ahli, dapat disimpulkan bahwa dengan menggunakan metode *prototype* ini, memungkinkan klien untuk lebih mengetahui mengenai hal teknis sistem dan bagi pengembang untuk lebih memahami kebutuhan pengguna secara rinci. Sehingga dengan menggunakan metode *prototype* ini dapat menangani ketidaksesuaian yang terjadi antara pengembang dengan klien.



Gambar 2.2 Model *Prototype* menurut Pressman

Berikut adalah tahap-tahap dalam metode *prototyping* [23]:

1. Pengumpulan kebutuhan
Pada tahap ini pelanggan dan pengembang bersama-sama mendefinisikan format dan kebutuhan perangkat lunak secara keseluruhan, mengidentifikasi semua kebutuhan, dan garis besar sistem yang akan dibuat.
2. Membangun *prototyping*
Yang dilakukan pada tahap ini adalah membuat desain sementara yang berpusat pada penyajian kepada pelanggan, misalnya; dengan membuat contoh *input* dan *output*.
3. Evaluasi *prototyping*
Jika *prototype* yang sudah dibangun sesuai dengan keinginan pelanggan maka dilakukan evaluasi. Jika sudah sesuai akan dilanjutkan ketahap pengkodean (*coding*), namun jika belum sesuai maka akan diperbaiki dengan mengulang kembali tahap 1, tahap 2, dan tahap 3.
4. Mengodekan sistem
Tahap ini lakukan jika *prototype* yang dibuat sudah diterima dan disepakati yang dilanjutkan dengan pengkodean dengan bahasa pemrograman yang sesuai
5. Menguji sistem
Sistem yang telah selesai dibuat kemudian akan dilakukan pengujian dengan menggunakan metode *black-box*. Tujuannya adalah untuk mengetahui apakah sistem yang dibuat sudah dapat berfungsi dengan baik dan benar sesuai yang diharapkan.
6. Evaluasi sistem

Tahap ini dilakukan untuk mengetahui apakah sistem yang dibangun telah sesuai dengan yang diharapkan. Jika sudah memenuhi dapat dilakukan tahap selanjutnya, namun jika tidak maka perlu dilakukan pengulangan pada tahap 4 dan tahap 5.

7. Penggunaan sistem

Sistem yang dibangun siap digunakan oleh pelanggan setelah tahap pengujian berhasil dan diterima oleh pelanggan.

B. Metode Pengujian Sistem

Tahap pengujian sistem ini menggunakan metode *black box testing*. *Black box testing* merupakan metode yang hanya berfokus untuk mengamati pada *input* dan *output* sistem dan menunjukkan fungsi sistem tentang cara beroperasinya tanpa mengetahui struktur kode dari sistem. Metode ini digunakan untuk mengetahui apakah perangkat lunak sudah berfungsi dengan benar. Pengujian *black box testing* bertujuan untuk menunjukkan fungsi dari sistem tentang cara bekerja, apakah input dan output sudah sesuai dengan yang diharapkan [24].

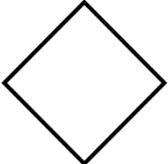
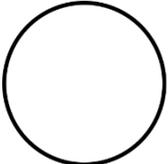
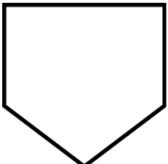
Salah satu klasifikasi *black-box testing* yaitu pengujian usabilitas atau *Usability Testing*. Pengujian ini dikenal juga sebagai pengujian untuk keakraban pengguna (*testing for user friendliness*). Pengujian usabilitas juga merupakan proses yang bekerja dengan pengguna akhir secara langsung maupun tidak langsung guna menilai bagaimana pengguna merasakan perangkat. Proses ini akan membongkar area kesulitan pengguna dalam menggunakan sistem yang telah dibuat.

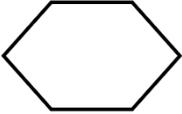
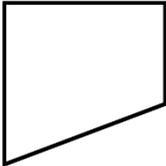
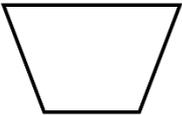
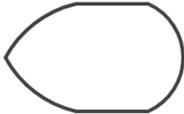
C. Tools/Alat Bantu Penelitian

1. Flowchart

Flowchart atau diagram alir merupakan diagram yang merepresentasikan langkah-langkah instruksi atau algoritma yang berurutan dalam sebuah sistem. *Flowchart* digambarkan dengan menggunakan simbol, dimana setiap simbol tersebut mewakili proses tertentu. Simbol-simbol tersebut kemudian dihubungkan satu sama lain menggunakan garis penghubung untuk membentuk suatu proses atau langkah-langkah sistem. Proses dalam *flowchart* digambarkan dari halaman atas ke bawah dan dari kiri ke kanan [22]. Simbol-simbol *flowchart* dapat dilihat pada Tabel 2.2.

Tabel 2. 2 Simbol *Flowchart*

No	Gambar Simbol	Nama	Keterangan
1.		Terminal	Menyatakan permulaan atau akhir suatu program.
2.		Arus/ <i>Flow</i>	Untuk menyatakan jalanya arus suatu proses.
3.		Proses	Menunjukkan sebuah kegiatan atau proses yang dilakukan sistem.
4.		<i>Decision</i> /Logika	Menunjukan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan (ya/tidak).
5.		<i>Input</i> / <i>output</i>	Menyatakan proses <i>input/output</i> tanpa tergantung dengan jenis peralatannya.
6.		<i>Connector</i>	Menyatakan sambungan dari suatu proses ke proses lainnya dalam halaman/lembar yang sama.
7.		<i>Offline</i> / <i>Connector</i>	Menyatakan sambungan dari suatu proses ke proses lainnya dalam halaman/lembar yang berbeda.

8.		<i>Predefined Proses</i>	Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
9.		Manual <i>input</i>	Memasukan data secara manual dengan menggunakan <i>online keyboard</i> .
10.		Manual	Menyatakan suatu Tindakan (proses) yang tidak dilakukan oleh komputer (manual).
11.		<i>Document</i>	Mencetak laporan ke printer.
12.		<i>Display</i>	Menyatakan peralatan output yang digunakan yaitu layar, plotter, printer dan sebagainya.

2. UML (*Unified Modeling Language*)

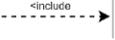
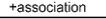
Unified Modeling Language merupakan salah satu metode pemodelan visual yang digunakan dalam perancangan dan pembuatan sebuah *software* yang berorientasikan pada objek. UML merupakan sebuah standar penulisan atau semacam *blueprint* dimana didalamnya termasuk sebuah bisnis proses, penulisan kelas-kelas dalam sebuah bahasa yang spesifik [23]. Jenis-jenis UML antara lain:

a. *Use Case Diagram*

Use case digunakan untuk menggambarkan fungsi dasar dari sebuah sistem informasi. *Use case* mendeskripsikan cara sistem bisnis

berinteraksi dengan lingkungannya [25]. Berikut adalah elemen-elemen dari use case diagram seperti yang terlihat pada Tabel 2.3.

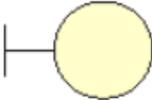
Tabel 2. 3 Simbol *Use Case* Diagram

No	Simbol	Fungsi
1.		<i>Actor/role</i> adalah orang atau sistem lain yang berinteraksi dengan sistem saat ini.
2.		Memasukkan satu <i>use case</i> dalam <i>use case</i> lainnya. Perilaku (<i>behavior</i>) yang harus terpenuhi agar sebuah <i>event</i> dapat terjadi, di mana kondisi ini sebuah <i>use case</i> adalah bagian dari <i>use case</i> lainnya. Tanda panah mengarah dari <i>base use case</i> (pusat) menuju ke <i>use case</i> yang di- <i>include</i> .
3		Menggambarkan navigasi antar <i>class</i> , berupa banyak objek lain yang berhubungan dengan satu objek, dan apakah suatu <i>class</i> menjadi bagian dari <i>class</i> lainnya.
4.		Berisi nama dari sistem yang diletakkan di dalam atau di bagian atas <i>boundary</i> . Mewakili ruang lingkup sistem. <i>Actor</i> berada di luar ruang lingkup sistem.
5.		<i>Use case</i> adalah bagian utama dari fungsionalitas sistem. Bisa <i>extend</i> (memperluas) <i>use case</i> lainnya. Ditempatkan di dalam <i>system boundary</i> (batasan sistem). Dilabeli dengan kata kerja – frase kata benda.

b. *Sequence* Diagram

Sequence diagram menunjukkan pesan yang lewat di antara objek untuk use case tertentu dari waktu ke waktu. *Sequence* diagram mengilustrasikan objek-objek yang berpartisipasi di dalam suatu use case [25]. Berikut adalah elemen-elemen dari *sequence* diagram seperti yang terlihat pada Tabel 2.4.

Tabel 2. 4 Simbol *Sequence Diagram*

No.	Nama	Simbol	Keterangan
1.	<i>Actor</i>		Menggambarkan orang yang berinteraksi dengan sistem
2.	<i>Entity Class</i>		Menggambarkan hubungan yang akan dilakukan
3.	<i>Boundary Class</i>		Menggambarkan sebuah gambaran dari <i>form</i>
4.	<i>Control Class</i>		Menggambarkan hubungan antara <i>boundary</i> dengan tabel
5.	<i>Lifeline</i>		Menggambarkan tempat mulai dan berakhirnya <i>message</i>
6.	Line Message		Menggambarkan pengiriman <i>message</i> antar <i>class</i>

c. Class Diagram

Class diagram adalah model statis yang menunjukkan kelas dan hubungan di antara kelas yang tetap konstan dalam sistem dari waktu ke waktu. *Class diagram* menggambarkan kelas, yang meliputi perilaku dan keadaan, dengan hubungan antar kelas [25]. Adapun elemen-elemen dari *class diagram* dapat dilihat pada Tabel 2.5.

Tabel 2. 5 Simbol *Class Diagram*

No.	Nama	Simbol	Keterangan
1.	<i>Generalization</i>		Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus)
2.	<i>Nary Association</i>		Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
3.	<i>Class</i>		Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
4.	<i>Realization</i>		Operasi yang dilakukan oleh suatu objek
5.	<i>Dependency</i>		Relasi antarkelas dengan makna kebergantungan antarkelas
6.	<i>Association</i>		Diagram yang menghubungkan antara objek satu dengan objek lainnya.

2.2.5 Rekayasa Web

Rekayasa *Web* adalah subdisiplin dari rekayasa perangkat lunak yang membantu menyediakan metodologi untuk merancang, mengembangkan, memelihara dan melibatkan aplikasi *web*. Rekayasa *Web* merupakan sebuah aplikasi yang menggunakan pendekatan sistematis, disiplin, dan terukur untuk pengembangan, operasi, dan pemeliharaan aplikasi berbasis web. Rekayasa *Web* memiliki tujuan yaitu secara efektif mendukung kesuksesan dalam mengatur kompleksitas dan pengembangan keanekaragaman aplikasi *web* [26].

A. PHP

Hypertext Preprocessor atau yang biasa disingkat menjadi PHP merupakan pemrograman interpreter yaitu proses penerjemahan baris kode sumber menjadi kode mesin yang dimengerti komputer secara langsung pada saat baris kode dijalankan. PHP disebut sebagai pemrograman *Server Side Programming*, hal ini dikarenakan seluruh prosesnya dijalankan pada server tidak dijalankan pada *client*. PHP merupakan suatu bahasa dengan hak cipta terbuka atau yang juga dikenal dengan istilah *Open Source*, yaitu pengguna dapat mengembangkan kode fungsi PHP dengan kebutuhannya [27].

B. HTML

HTML adalah kepanjangan dari *Hypertext Markup Language*. Perlu diketahui bahwa HTML bukan termasuk bahasa pemrograman, melainkan HTML adalah *markup* standar yang digunakan dalam menampilkan halaman *website*. *Markup language* adalah dokumen HTML yang memiliki banyak tanda tertentu untuk menentukan tampilan teks dan tingkat kepentingan dari teks tersebut dalam dokumen. Pembacaan suatu dokumen dapat langsung pada ke topik yang diinginkan sesuai dengan link-nya melalui *hypertext* pada dokumen HTML [28].

C. Web Server

Web server adalah sebuah *software* yang memberikan layanan berbasis data dengan menggunakan protokol HTTP atau HTTPS dari *client* menggunakan aplikasi *web browser* untuk *request* data dan *server* akan mengirim data dalam bentuk halaman *web* dan pada umumnya berbentuk dokumen HTML. Halaman *web* yang diminta bisa terdiri dari berkas teks, video, gambar, file dan banyak lagi.

Salah satu *web server* yang bersifat *open source* ialah *apache* yang digunakan untuk melayani dan melakukan pengaturan fasilitas *web*, pada umumnya memiliki fungsi untuk memperoleh berkas berisi permintaan

(*request*) dari *client* melalui *web browser*, kemudian *apache* akan memproses data tersebut dengan menghasilkan keluaran (*output*) yang diinginkan oleh *client*. Output didapatkan berdasarkan data yang tersimpan dalam *database website* tersebut. Sesuai hasil survei yang dilakukan oleh Netcraft, bulan Januari 2005 saja jumlahnya tidak kurang dari 68% pangsa *web server* yang berjalan di Internet. Ini berarti jika semua *web server* selain *Apache* digabung, masih belum bisa mengalahkan jumlah *Apache* [29].

D. Web Browser

Web Browser adalah aplikasi perangkat lunak yang digunakan untuk mengambil dan menyajikan sumber informasi web. *Web Browser* digunakan untuk menjelajahi situs *web* lewat layanan HTTP. Untuk mengakses layanan WWW (*World Wide Web*) dari sebuah komputer digunakan program *web* klien yang disebut dengan *web browser / browser*. Sumber informasi web diidentifikasi dengan *Uniform Resource Identifier* (URL) yang dapat terdiri dari halaman web, video, gambar, ataupun konten lainnya. Jenis-jenis *web browser* yang populer antara lain: Google Chrome, Mozilla Firefox, *Safari Browser*, *Opera Browser*, *Internet Explorer*, dan masih banyak lagi [30].

E. Framework Laravel

Framework merupakan sebuah kerangka kerja pemrograman yang dikembangkan untuk memudahkan perancangan sebuah *website*. *Framework* memiliki komponen dan variabel yang dibutuhkan para *website developer* dalam pembacaan kode, perencanaan, pengujian dan pemeliharaan sebuah *website*. Sebuah *framework* biasanya mencakup *library* kode, *Application Programming Interfaces* (API), model perangkat lunak, dan elemen lainnya [31].

Laravel merupakan salah satu *framework* berbasis *Hypertext Preprocessor* (PHP) yang telah menggunakan konsep *Model View Controller* (MVC). Laravel dirilis pada 5 Juni 2011 dibawah lisensi MIT License dengan GitHub sebagai tempat untuk berbagi kode. MVC merupakan suatu pendekatan yang memisahkan antara tampilan, logika, dan *database*. *Model* untuk membantu mengelola data ke *database*, *view* untuk mengatur tampilan dari sistem, dan *controller* untuk logika-logika setiap fungsi pada sistem. Laravel stabil dirilis pada 13 Juli 2016 dengan versi 5.2.36 dengan bahasa pemrograman PHP dan sistem operasi *cross platform* [31].

2.2.6 Basis Data (*Database*)

Basis data atau *database* adalah sekumpulan data yang terhubung dan tersimpan secara sistematis pada suatu media, tidak memerlukan kerangkapan data dan tidak perlu mengatap satu sama lain. Basis data memiliki tujuan yaitu supaya dapat memudahkan pencarian data yang dapat memudahkan pencarian data dalam proses menemukan kembali atau memperoleh data kembali secara mudah dan cepat [32]. Basis data memungkinkan tempat penyimpanan data yang besar dan dapat digunakan secara bersamaan oleh banyak departemen dan pengguna. *Database* mewakili entitas, atribut, dan hubungan logis antara entitas. Basis data terdiri dari kumpulan data yang terorganisir, relasi antar data, dan objektifnya. Objektif utama adalah kecepatan dan kemudahan berinteraksi dengan data yang dikelola atau diolah. Selain itu terdapat pengertian bahwa basis data adalah sekumpulan data persisten yang digunakan oleh aplikasi sistem dari perusahaan.

A. DBMS (*Database Management System*)

DBMS (*Data Base Management System*) adalah koleksi terpadu dari aplikasi program (*system software*) yang dapat digunakan dalam melakukan definisi membuat melakukan akses dan memelihara *database*. DBMS menyediakan berbagai fasilitas yaitu [33]:

1. DDL (*Data Definition Language*) merupakan perintah-perintah yang biasa digunakan pada *database* untuk mendefinisikan skema dan subskema *database*. Perintah yang termasuk di dalamnya yaitu:
 - a. *Create* : Digunakan untuk membuat, termasuk diantaranya membuat *database* dan tabel baru.
 - b. *Alter* : Digunakan untuk mengubah struktur tabel yang telah dibuat.
 - c. *Drop* : Digunakan untuk menghapus *database* dan tabel.
2. DML (*Data Manipulation Language*) merupakan merupakan perintah-perintah yang memungkinkan pengguna melakukan akses dan manipulasi data sebagaimana yang telah diorganisasikan sebelumnya dalam model data yang tepat, (*Data Manipulation Language*) digunakan untuk memanipulasi *database* yang telah didefinisikan dengan DDL. Perintah yang termasuk DML:

- a. *Insert* : Digunakan untuk menyisipkan atau memasukan dalam tabel.
- b. *Select* : Digunakan untuk mengambil data atau menampilkan data dari suatu tabel atau beberapa tabel.
- c. *Update* : Digunakan untuk memperbaharui data lama menjadi data terbaru.
- d. *Delete* : Digunakan untuk menghapus data dari tabel.

B. SQL (*Structured Query Language*)

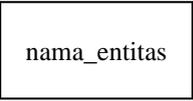
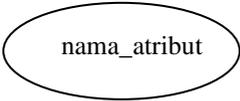
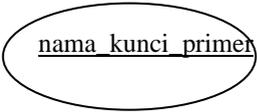
SQL (*Structured Query Language*) adalah bahasa yang digunakan untuk mengelola data pada RDBMS, SQL awalnya dikembangkan berdasarkan teori aljabar relasional dan kalkulus. MySQL merupakan basis data yang dikembangkan dari bahasa SQL (*structure query language*). MySQL merupakan sebuah *database* yang cukup populer di dunia, saat ini penggunaannya sudah mencapai sekitar 6 juta pengguna. MySQL adalah RDBMS (*Relational Database Management System*) server yang berarti MySQL yaitu program yang dapat memungkinkan pengguna untuk dapat membuat, mengelola, serta menggunakan data pada suatu model relasional.

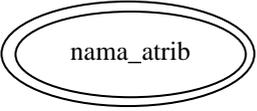
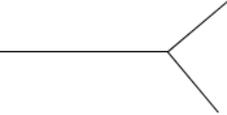
Kelebihan menggunakan MySQL adalah didukung oleh berbagai bahasa dalam memberikan pesan *error* seperti Belanda, Inggris, Jerman, Italia, Spanyol, dan Portugis, MySQL juga tidak membutuhkan RAM yang besar, memiliki struktur tabel yang cukup fleksibel. MySQL juga cepat, handal, dan mudah dalam penggunaannya sehingga tidak memerlukan seseorang yang ahli untuk menggunakannya [34].

C. ERD (*Entity Relationship Diagram*)

ERD (*Entity Relationship Diagram*) merupakan tools yang digunakan untuk memodelkan struktur data dengan menggambarkan entitas dan hubungan antar entitas (*relationship*) secara abstrak (konseptual). ERD berfungsi untuk memodelkan struktur data dan hubungan antar data, untuk menggambarkannya digunakan beberapa notasi dan symbol [35]. Adapun simbol-simbol ERD dapat dilihat pada Tabel 2.1.

Tabel 2. 6 Simbol-simbol ERD [16]

No	Simbol	Keterangan
1.	Entitas / <i>entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.
2.	Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.
3.	Atribut kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom; asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).

4.	Atribut multi nilai / <i>multivalue</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih.
5.	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja
6.	Asosiasi / <i>Association</i> 	Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan <i>one to many</i> yang menghubungkan entitas A dan entitas B.

2.2.7 Notifikasi Email

Notifikasi *email* adalah pemberitahuan atau pesan otomatis yang dikirimkan melalui *email* kepada pengguna atau penerima tertentu untuk memberi tahu mereka tentang suatu peristiwa atau kejadian yang relevan. Notifikasi *email* sering digunakan dalam berbagai aplikasi dan platform untuk memberikan informasi penting, pengingat, atau konfirmasi kepada pengguna [36].