

BAB II LANDASAN TEORI

2.1 Tinjauan Pustaka

Jurnal penelitian Annisa Suwanto dan Tri Sandhika Jaya (2019) dari Jurusan Manajemen Informatika, Politeknik Negeri Lampung dengan judul Sistem Pengaduan Peserta Pensiun Berbasis WEB pada PT. Taspen Persero KC Bandar Lampung. Dimana dalam penelitian yang di lakukan penulis mengangkat masalah proses pengaduan saat ini hanya menggunakan kotak sehingga nasabah harus datang langsung ke kantor PT. Taspen untuk mengirimkan surat pengaduan, hal ini mempersulit nasabah dalam melakukan pengaduan untuk mengetahui apakah pengaduannya sudah ditindaklanjuti atau belum oleh pihak PT. Taspen. Hasil dari tugas akhir ini adalah Sistem Pengaduan Peserta Pensiun Berbasis Web pada PT. Taspen Persero KC Bandar Lampung. untuk mempermudah dalam pengaksesan data secara realtime dan terintegrasi dengan database pengaduan sehingga lebih akurat. Metode pengembangan sistem yang digunakan pada aplikasi ini, yaitu metode pengembangan RAD (Rapid Application Development).

Jurnal penelitian Farhan Firdaus dan Fajar Masya (2020) dari Program Studi Sistem Informasi, Universitas Mercu Buana dengan judul Aplikasi Informasi Pengaduan Electronic Data Capture Merchant (Edc) Berbasis Web. Dimana dalam penelitian ini penulis mengangkat masalah mengenai pelayanan pengaduan keluhan dan kerusakan pada EDC yang ada pada merchant, namun untuk saat ini pelaporan kerusakan pada EDC harus melalui beberapa tahapan,

artinya tidak bisa langsung berhubungan dengan tim terkait, sehingga proses perbaikannya memerlukan waktu yang lama, hal ini membuat kurang maksimalnya pelayanan Bank BRI dalam mengumpulkan dan menjaring komplain kerusakan EDC. Dengan membuat sarana pengaduan kerusakan dan keluhan merchant EDC yang berupa web maka Merchant Bank BRI dapat mengadukan permasalahan yang ada pada EDC dan diharapkan dapat penanganan langsung dengan tim terkait tanpa dibatasi tempat dan waktu, sehingga merchant pengguna EDC mempunyai kemudahan untuk melakukan komplain mengenai permasalahan EDC kapanpun dan dimanapun.

Jurnal penelitian Mikrawati, Muhammad Sarjan, Syarli (2020) tentang Sistem Informasi Layanan Pengaduan Nasabah pada PT. Amartha Mikro Fintek Berbasis Web Informasi Layanan Pengaduan. Dimana dalam penelitian penulis mengangkat masalah yang terjadi Selama ini nasabah harus datang ke kantor Amartha Mikro Fintek Unit Tapalang untuk melaporkan masalah yang di hadapi. Untuk masalah pengaduan melalui telepon sering terjadi kesalahan dan kelalaian dalam penanganan. informasi keluhan pelayanan belum dapat di efisienkan secara maksimal, sehingga dapat memperlambat kinerja petugas dalam memberikan pelayanan terhadap nasabah. Tujuan dari penelitian ini adalah untuk merancang Sistem Informasi Layanan Pengaduan Nasabah PT. Amartha Mikro Fintek Berbasis Web agar dapat mempermudah nasabah dalam melakukan pengaduan pada PT. Amartha Mikro Fintek, serta dapat mengefisienkan waktu. Metode penelitian ini menggunakan metode waterfall untuk menganalisis sistem yang sedang berjalan serta sistem yang akan dirancang, dan menggunakan teknologi berbasis web dengan Hypertext

PreProcessor sebagai bahasa pemrograman utama. Hasil dari penelitian ini adalah terciptanya sebuah aplikasi Sistem Informasi Layanan Pengaduan Nasabah PT. Amartha Mikro Fintek Berbasis Web untuk membantu nasabah dalam melakukan pengaduan.

Jurnal penelitian Nur Lutfiyana (2015) tentang Rancang Bangun Sistem Informasi Layanan Pengaduan Nasabah Kartu Kredit Berbasis Web. Dalam hal ini penulis mengangkat masalah yang terjadi saat ini perusahaan perbankan penyedia layanan kartu kredit menerima pengaduan dari nasabah melalui telepon, pengaduan melalui telepon menyebabkan nasabah mengalami kesulitan alam menghubungi *customer service* yang dikarenakan semua *customer service* sedang melayani nasabah lain atau dikarenakan jumlah *customer service* yang terbatas. Selama ini pengaduan dari nasabah oleh petugas *customer service* dimasukkan ke dalam data pengaduan menggunakan Microsoft Excel, hal ini dapat menyebabkan terjadinya kehilangan data, kesulitan pencarian data sampai dengan tidak tertanganinya pengaduan dari nasabah. Tujuan penulisan tugas akhir ini adalah Membangun sebuah sistem informasi berbasis *web* yang memberikan kemudahan kepada nasabah kartu kredit dalam menyampaikan pengaduan. Membangun sebuah sistem informasi berbasis *web* yang mempermudah petugas *customer service* dalam mengontrol atau mencari pengaduan yang belum ditangani. Memperkecil terjadinya kehilangan data, ketidak akuratan data serta mempermudah dalam melakukan rekapitulasi data. Metode penelitian yang digunakan dibagi menjadi dua yaitu, metode pengumpulan data : Observasi dan Studi Pustaka. Sedangkan metode pengembangan sistem menggunakan model waterfall. Hasil penelitian ini

terciptanya sebuah sistem informasi berbasis web dapat memungkinkan nasabah untuk memantau penanganan atas pengaduan yang diajukan. Sistem informasi ini memberikan kemudahan kepada petugas *customer service* dalam mengontrol atau mencari pengaduan nasabah yang belum ditangani. Serta sistem informasi ini dapat menghilangkan kemungkinan terjadinya kehilangan data, ketidakakuratan serta mempermudah dalam rekapitulasi data pengaduan nasabah.

Jurnal penelitian Zuniar Prihatin (2016) tentang Sistem Penanganan Permintaan dan Keluhan Nasabah Kartu Kredit Berbasis Web Study Kasus Bank XYZ. Penelitian ini dilakukan untuk pengembangan sistem informasi penanganan permintaan dan keluhan nasabah kartu kredit pada Bank XYZ. Aplikasi permintaan dan keluhan nasabah berbasis web yang akan dibangun dapat melakukan perekaman data yang diajukan oleh nasabah ke dalam sistem melalui aplikasi secara online, selain itu dapat memberikan informasi terkait tenggat waktu untuk menyelesaikan setiap permintaan dan keluhan yang datang dari nasabah. Aplikasi ini menggunakan metode waterfall yaitu suatu proses pembuatan situs web secara terstruktur dan berurutan di mulai dari penentuan masalah, analisa kebutuhan, perancangan implementasi, integrasi, uji coba sistem, dan penempatan situs web dan pemeliharaan. SDLC (System Development Life Cycle) yang didokumentasikan dengan diagram alir data dan sistem tiket. Hasil penelitian adalah aplikasi ini dapat mengatasi jarak antara nasabah dan pihak kartu kredit serta dapat mengatasi pencarian tiket permohonan yang belum bisa diatasi secara manual sehingga dapat meningkatkan efisiensi waktu dan tenaga.

2.2 Landasan Teori

2.2.1 Pelayanan

Jurnal penelitian Kresna B. D. (2012) Mengemukakan bahwa “Sebagian perusahaan sengaja memanjakan nasabah melalui pelayanan yang diberikan bahkan dewasa ini nasabah sudah dianggap sebagai raja dan nasabah juga dianggap sebagai bagian dari perusahaan yang harus segera dipenuhi kebutuhan serta keinginannya seperti dilayani secara cepat dan akurat”. Maka setiap bank menentukan pelayanan prima yakni sikap atau cara karyawan untuk melayani nasabah secara memuaskan, kepuasan nasabah akan lebih mudah dipenuhi kalau petugas bank lebih mengenal sifat dan karakter dari para nasabah. Pelayanan yang diberikan oleh bank haruslah mencerminkan pendekatan seutuhnya dari seorang karyawan bank kepada nasabah.

Jurnal penelitian Lewis & Booms Tjiptono dan Chandra, (2005) kualitas layanan adalah ukuran seberapa bagus tingkat layanan yang diberikan mampu sesuai dengan ekspektasi pelanggan. Layanan merupakan perilaku produsen dalam rangka memenuhi kebutuhan dan keinginan konsumen demi tercapainya kepuasan pada konsumen itu sendiri. Kotler juga mengatakan bahwa perilaku tersebut dapat terjadi pada saat, sebelum dan sudah terjadinya transaksi. Pada umumnya layanan yang berkualitas tinggi akan menghasilkan kepuasan yang tinggi serta pembelian ulang yang lebih sering. Dari definisi-definisi tentang kualitas pelayanan tersebut dapat diambil kesimpulan bahwa kualitas pelayanan adalah segala bentuk aktifitas yang dilakukan oleh perusahaan guna memenuhi harapan konsumen. Layanan dalam hal ini diartikan sebagai jasa atau service yang disampaikan oleh pemilik jasa yang berupa kemudahan, kecepatan, hubungan, kemampuan dan

keramahtamahan yang ditujukan melalui sikap dan sifat dalam memberikan pelayanan untuk kepuasan konsumen.

Dimensi kualitas pelayanan, yaitu sebagai berikut Lupiyoadi, (2001):

1. *Tangible* atau bukti langsung

Tangible yaitu kemampuan suatu perusahaan dalam menunjukkan eksistensinya kepada eksternal, penampilan dan kemampuan sarana dan prasarana fisik perusahaan dan keadaan lingkungan sekitarnya adalah bukti nyata pelayanan yang diberikan oleh pemberi jasa yang mengikuti fasilitas fisik (gedung, gudang, dll) perlengkapan dan peralatan yang dapat dipergunakan (teknologi) penampilan pegawainya.

2. *Reliability* atau kehandalan

Reliability yaitu kemampuan perusahaan untuk memberikan pelayanan sesuai yang dijanjikan secara akurat dan terpercaya. Kinerja harus sesuai dengan harapan pelanggan yang berarti ketepatan waktu, pelayanan yang sama untuk semua pelanggan tanpa kesalahan, sikap yang disimpatik dan dengan akurasi yang tinggi.

3. *Responsiveness* atau daya tanggapan

Responsiveness yaitu suatu kemampuan untuk membantu dan memberikan pelayanan yang cepat (*Responsive*) dan tepat pada pelanggan, dengan penyampaian informasi yang jelas. Membiarkan konsumen menunggu tanpa adanya suatu alasan yang jelas menyebabkan persepsi yang negatif dalam kualitas pelayanan.

4. *Assuarance* atau jaminan

Assuarance yaitu pengetahuan, kesopansantunan, dan kemampuan para pegawai perusahaan untuk menumbuhkan rasa percaya para pelanggan kepada perusahaan.

5. *Emphaty* atau empati

Emphaty yaitu memberikan perhatian yang tulus dan bersifat individual atau pribadi yang diberikan kepada pelanggan dengan berupaya memahami keinginan konsumen. Dimana suatu perusahaan diharapkan memiliki pengertian dan pengetahuan tentang pelanggan, memahami kebutuhan pelanggan secara spesifik serta memiliki waktu pengoprasian yang nyaman bagi pelanggan.

2.2.2 Pengembangan

Pengertian pengembangan secara umum adalah suatu usaha untuk meningkatkan kemampuan teknis teoritis, konseptual dan moral karyawan sesuai dengan kebutuhan pekerjaan / jabatan melalui pendidikan dan pelatihan.

Sedangkan pengertian pengembangan secara khusus yaitu terkhusus pada pengembangan aplikasi (perangkat lunak) adalah pengembangan suatu produk perangkat lunak. Istilah “pengembangan perangkat lunak” bisa dipakai untuk menyebut aktivitas pemrograman komputer, yaitu proses menulis dan mengelola kode sumber, namun dalam artian luas istilah ini mencakup semua hal yang terlibat antara penciptaan perangkat lunak yang diinginkan melalui pewujudan akhir perangkat lunak, idealnya dalam proses yang terencana dan terstruktur. Karena itu, pengembangan perangkat lunak bisa mencakup penelitian,

pengembangan baru, purwarupa, modifikasi, pemakaian kembali, rekayasa ulang, pengelolaan, atau aktivitas lain yang menghasilkan produk perangkat lunak.

Perangkat lunak bisa dikembangkan untuk berbagai tujuan, tiga tujuan paling umum adalah memenuhi kebutuhan klien / bisnis tertentu (perangkat lunak kustom), memenuhi persepsi kebutuhan sejumlah pengguna potensial (perangkat lunak komersial dan terbuka), atau memenuhi kebutuhan pribadi (misalnya seorang ilmuwan menulis perangkat lunak untuk mengotomasikan sebuah tugas yang rumit). Pengembangan perangkat lunak tertanam adalah pengembangan perangkat lunak tertanam seperti yang dipakai untuk mengontrol produk konsumen, membutuhkan proses pengembangan yang terintegrasikan dengan pengembangan produk fisik yang dikontrol.

Perlunya pengawasan kualitas yang lebih baik pada proses pengembangan perangkat lunak menciptakan disiplin teknik perangkat lunak, yang bertujuan menerapkan pendekatan sistematis yang tercantum dalam paradigma teknik hingga proses pengembangan perangkat lunak.

2.2.3 Aplikasi

Aplikasi berasal dari kata *application* yaitu bentuk benda dari kata kerja *to apply* yang dalam bahasa Indonesia berarti pengolah. Secara istilah, aplikasi komputer adalah suatu subkelas perangkat lunak komputer yang menggunakan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pemakai.

Adapun pengertian aplikasi menurut para ahli adalah :

1. Menurut Ali Zaki dan Smitdev *Community*

Aplikasi adalah komponen yang berguna melakukan pengolahan data maupun kegiatan-kegiatan seperti pembuatan dokumen atau pengolahan data. Dan aplikasi adalah bagian PC yang berinteraksi langsung dengan user. Aplikasi berjalan di atas sistem operasi, sehingga aplikasi dapat diaktifkan dan perlu melakukan instalasi sistem operasi terlebih dahulu.

2. Menurut Hendrayudi

Aplikasi adalah kumpulan perintah program yang dibuat untuk melakukan pekerjaan-pekerjaan tertentu.

3. Menurut Hengky W. Pramana

Aplikasi adalah suatu unit perangkat lunak yang dibuat untuk melayani kebutuhan akan beberapa aktivitas seperti sistem perniagaan, game pelayanan masyarakat, periklanan, atau semua proses yang hampir dilakukan manusia.

Aplikasi adalah program siap pakai yang dapat digunakan untuk menjalankan perintah-perintah dari pengguna aplikasi tersebut dengan tujuan mendapatkan hasil yang lebih akurat sesuai dengan tujuan pembuatan aplikasi tersebut, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputansi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan.

2.2.4 Pengertian *Website*

Ada banyak sumber yang mendefinisikan tentang *website*, tetapi dari definisi tersebut menjelaskan memiliki inti yang sama. Berikut beberapa definisi *website* dari sumber yang berbeda :

Pada awalnya aplikasi web dibangun dengan hanya menggunakan bahasa yang disebut HTML (HyperText Markup Language). Pada perkembangan berikutnya, sejumlah skrip dan objek dikembangkan untuk memperluas kemampuan HTML seperti PHP dan ASP pada skrip dan Applet pada objek. Aplikasi Web dapat dibagi menjadi dua jenis yaitu aplikasi web statis dan dinamis. Web statis dibentuk dengan menggunakan HTML. Kekurangan aplikasi seperti ini terletak pada keharusan untuk memelihara program secara terus menerus untuk mengikuti setiap perkembangan yang terjadi. Kelemahan ini diatasi oleh model aplikasi web dinamis. Pada aplikasi web dinamis, perubahan informasi dalam halaman web dilakukan tanpa perubahan program tetapi melalui perubahan data. Sebagai implementasi, aplikasi web dapat dikoneksikan ke basis data sehingga perubahan informasi dapat dilakukan oleh operator dan tidak menjadi tanggung jawab dari webmaster. Arsitektur aplikasi web meliputi klien, web server, middleware dan basis data. Klien berinteraksi dengan web server. Secara internal, web server berkomunikasi dengan middleware dan middleware yang berkomunikasi dengan basis data. Contoh middleware adalah PHP dan ASP. Pada mekanisme aplikasi web dinamis, terjadi tambahan proses yaitu server menerjemahkan kode PHP menjadi kode HTML. Kode PHP yang diterjemahkan oleh mesin PHP yang akan diterima oleh klien. (Abdul Kadir,2009).

Menurut Achmad dkk (2017), “*Web* adalah salah satu aplikasi yang berisikan dokumen dokumen multimedia (teks, gambar, suara, animasi, video) di dalamnya yang menggunakan protocol HTTP (hypertext transfer protocol) dan untuk mengaksesnya menggunakan perangkat lunak yang disebut browser”.

Menurut Dillon, Schonhaler, dan Vossen (2017 : 1), sejak awal 1990, *world wide web* atau website merevolusi kehidupan pribadi maupun profesional. Web menjadi situs yang terus berkembang dan sebagai perpustakaan informasi yang ada di mana-mana yang dapat diakses melalui mesin pencari dan portal. Web menjadi tempat penyimpanan media yang memfasilitasi hosting dan berbagi sumber daya yang sering kali gratis dan sebagai pendukung layanan *do-it-yourself*. Web juga menjadi platform perdagangan tempat orang dan perusahaan semakin menjalankan bisnisnya.

Menurut Sebok, Vermat, dan tim (2018 : 70) adalah kumpulan halaman yang saling terhubung yang di dalamnya terdapat beberapa item seperti dokumen dan gambar yang tersimpan di dalam *web server*. *Web app* adalah sebuah aplikasi yang berada dalam *web server* yang bisa user akses melalui *browser*. *Web app* biasanya menampilkan data user dan informasi dari server.

Beberapa tipe website menurut Sebok, Vermat, dan tim (2018 : 75), yaitu :

1. *Search Engines*

Search engine adalah perangkat lunak yang menemukan situs web, halaman web, gambar, video, berita, peta, dan informasi lain yang berkaitan dengan topik tertentu.

2. *News, Weather, Sports, and Other Mass Media*

Situs website ini berisi materi yang layak diberitakan termasuk cerita dan artikel yang berkaitan dengan kejadian terkini, kehidupan, uang, politik, cuaca, dan olahraga.

3. *Educational*

Website *educational* menawarkan jalan yang menarik dan menantang untuk pengajaran dan pembelajaran formal dan informal. Instruktur sering menggunakan web untuk menyempurnakan pengajaran di kelas dengan menerbitkan materi, nilai, dan informasi kelas terkait lainnya.

4. *Business, Governmental, and Organizational*

Merupakan website yang berisi konten yang meningkatkan kesadaran merek, memberikan latar belakang perusahaan, dan mempromosikan produk atau layanan. Hampir setiap perusahaan memiliki situs web bisnis.

5. *Banking and Finance*

Online banking dan *online trading* memungkinkan *user* untuk mengakses catatan keuangan mereka dari mana saja selama memiliki koneksi internet. Dengan menggunakan *online banking*, *user* dapat mengakses akun, membayar tagihan, mentransfer dana, dan mengelola aktivitas keuangan lainnya. Dengan *online trading*, *user* dapat berinvestasi di saham atau pasar uang tanpa menggunakan *broker*.

6. *Travel and Tourism*

Travel and tourism website memungkinkan *user* untuk mencari opsi perjalanan dan membuat pengaturan perjalanan. *User* dapat membaca *review* perjalanan, mencari dan membandingkan harga penerbangan, memesan maskapai penerbangan, kamar, atau mobil sewaan.

7. *E-Commerce*

E-Commerce merupakan transaksi bisnis yang terjadi melalui jaringan elektronik. Beberapa orang menggunakan istilah *M-Commerce* atau

mobile commerce untuk mengidentifikasi *E-Commerce* yang terjadi menggunakan perangkat *mobile*. Penggunaan *E-Commerce* yang sering dijumpai yaitu belanja dan lelang, keuangan, perjalanan, hiburan, dan kesehatan.

2.2.5 *Rapid Application Development (RAD)*

Dalam pengembangan aplikasi web pengaduan nasabah di PT. BPR Utomo Manunggal Sejahtera Lampung digunakan metode *rapid application development* sebagai acuan dalam tahap – tahap pengerjaan -nya. Berikut gambaran *rapid application development (RAD)*.

2.2.5.1 Pengertian *Rapid Application Development (RAD)*

Pada penelitian ini model pengembangan sistem yang akan digunakan yaitu menggunakan metode *Rapid Application Development (RAD)*. *Rapid Application Development (RAD)* adalah sebuah model proses perkembangan perangkat lunak (*Software*) *sekuensial linier* yang menekankan pada siklus perkembangan yang cepat dengan menggunakan pendekatan konstruksi berbasis komponen. Sehingga apabila kebutuhan dipahami dengan baik, model ini akan membentuk sistem fungsional yang utuh serta dapat diselesaikan dalam waktu kira-kira 60 – 90 hari (Pressman, 2002).



Gambar 2.1 Tahapan RAD
Sumber : Kendall, (2011)

Model ini dikembangkan oleh James Martin dengan tahapan sebagai berikut (Kendall & Kendall, 2010):

1. Fase Menentukan Syarat-Syarat (*Requirement Planning Phase*)

Fase ini merupakan fase pertemuan antara penganalisis dan pengguna untuk mengidentifikasi tujuan dari sistem yang akan dibangun serta mengidentifikasi syarat-syarat informasi yang akan timbul untuk mencapai tujuan tersebut serta menganalisa semua sistem yang dibutuhkan oleh pengguna.

2. Fase Perancangan (*User Design Phase*)

Fase ini merupakan fase dalam bentuk workshop desain RAD antara penganalisis dan pemrogram untuk merancang sebuah sistem yang akan dibangun. Penganalisis dan pemrogram saling bekerja sama dalam membangun sistem dan menunjukkan representasinya dalam bentuk visual desain dan pola kerja kepada pengguna sistem. Pada fase ini juga pengguna merespon prototipe yang telah dirancang. Penganalisis dan pemrogram dapat memperbaiki serta menganalisis

modul modul yang dirancang berdasarkan dari respon pengguna sistem.

3. Fase Konstruksi (*Construction Phase*)

Fase Konstruksi merupakan fase eksekusi dalam bentuk pembuatan script program dan merupakan kelanjutan dari fase kedua. Pada fase ini juga menunjukkan platform, hardware, dan software yang digunakan. Setiap desain yang dibuat pada fase sebelumnya, akan ditingkatkan dengan menggunakan perangkat RAD. Setelah fungsi baru tersedia, fungsi baru tersebut ditunjukkan kepada pengguna untuk mendapatkan interaksi dan revisi, selanjutnya penganalisis akan melakukan perubahandalam setiap desain aplikasi berdasakan instruksi dari pengguna.

4. Fase Pelaksanaan (*Cutover Phase*)

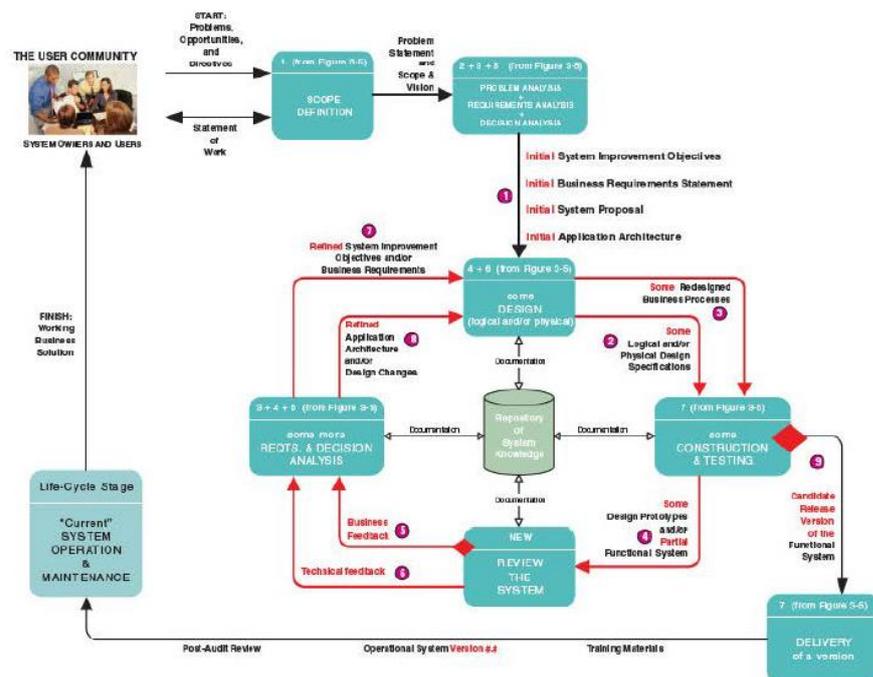
Fase ini penganalisis berkerja dengan para pengguna secara intensif selama workshop berlangsung, dan merancang beberapa aspek dan nonteknis yang dibutuhkan. Segera setelah aspek-aspek ini disetujui dan sistem-sistem dibangun dan disaring, sistem-sistem baru atau bagian dari sistem diujicoba dan kemudian diperkenalkan kepada organisasi.

Menurut Whitten & Bentley (2007:98) *Rapid Application Development* (RAD) adalah sebuah strategi pengembangan sistem yang menekankan kecepatan dalam pengembangan melalui keterlibatan pengguna dalam pembangunan secara cepat, iteratif, dan *incremental* dari suatu serangkaian *prototype* dari suatu sistem yang dapat berkembang menjadi suatu sistem akhir atau versi tertentu.

Ide dasar *Rapid Application Development* (RAD) adalah sebagai berikut:

1. Untuk lebih mengefektifkan pengguna dalam keterlibatan kegiatan menganalisa, mendesain dan membangun.
2. Untuk mengorganisasikan pengembangan sistem agar lebih fokus, dan lebih menerlibatkan *system owner, users, analysts, designer, dan builder*.
3. Untuk mempercepat kegiatan analisa dan perancangan kebutuhan sistem melalui pendekatan pembangunan iteratif.
4. Untuk mengurangi waktu yang dibutuhkan pengguna untuk melihat proses kerja sistem yang dikembangkan.

Berikut adalah proses siklus dari *Rapid Application Development* (RAD).



Gambar 2.2 *Rapid Application Development*.(Whitten & Bentley, 2007, p99)

Whitten & Bentley menjelaskan gambaran proses *Rapid Application Development* (RAD) diatas sebagai berikut :

1. Untuk menekan waktu yang dibutuhkan dalam pengembangan sistem dan aplikasi maka tahap analisa masalah, analisa kebutuhan sistem, dan analisa keputusan perlu dipercepat dan dikonsolidasi. Penggunaan RAD yang menggunakan pendekatan *iterative* dalam proses *initial analysis* dapat diselesaikan dalam waktu beberapa minggu.
2. Proses desain *physical* dan *logical* biasanya disingkat dan dipercepat secara signifikan. Dalam setiap *iteration cycle* hanya beberapa spesifikasi desain yang diperhatikan. Ketika beberapa model sistem dapat diambil, mereka dapat dipilih dan ditekankan untuk keberlangsungan dari *rapid development*.
Megasumsikan kesalahan dapat ditemukan dan diatasi pada proses iteratif berikutnya.
3. Jarang sekali namun terdapat beberapa kemungkinan proses bisnis perlu didesain ulang untuk merepresentasikan keterlibatan aplikasi dalam sistem.
4. Dalam setiap *iteration cycle*, beberapa desain *prototype* dan beberapa fungsi dari bagian sistem dibangun dan di uji coba. Nantinya, aplikasi yang sudah selesai akan menjadi hasil dari *final iteration process*.
5. Setelah setiap *prototype* dan fungsional sistem dikembangkan dan di uji coba, pengguna diberikan kesempatan untuk mencoba menggunakan. Harapan dari pengguna terhadap *prototype* akan menjadi kebutuhan sistem baru dan memberikan tanggapan terhadap gambaran proses bisnis.

6. Setelah setiap *prototype* dan fungsional sistem dikembangkan dan di uji coba, *system analysts* dan *designers* akan *me-review* untuk mendapatkan tanggapan teknis dan arah pengembangan untuk proses berikutnya.
7. Berdasarkan tanggapan yang telah didapatkan, *system analysts* akan mengidentifikasi tujuan atau arah pembenaran sistem yang akan dilakukan atau kebutuhan sistem.
8. Berdasarkan tanggapan yang didapatkan, *designers* akan mengidentifikasi tujuan atau arah perubahan desain dan pembenaran arsitektur aplikasi.
9. Nantinya sistem akan dapat dinilai siap untuk diimplementasikan.

2.2.5.2 Keuntungan dan Kerugian *Rapid Application Development*

Menurut Whitten & Bentley (2007:100) *Rapid application development* (RAD) memberikan beberapa keuntungan dalam pemanfaatannya dalam membangun *prototype website* namun juga memiliki kekurangan yang dapat menyebabkan kerugian. Berikut adalah penjabaran lebih lanjut mengenai keuntungan dan kerugian *rapid application development* dalam pemanfaatannya.

Tabel 2.1 Keuntungan dan Kerugian *Rapid Application Development*

(Sumber : Whitten & Bentley, 2007, p100)

Keuntungan	Kerugian
<ol style="list-style-type: none"> 1. Sangat berguna untuk proyek dimana kebutuhan sistem yang akan dikembangkan belum sepenuhnya tepat atau tidak menentu. 2. Mendorong antusias <i>end-user</i> untuk berpartisipasi dalam proyek tersebut. 3. Proyek memiliki visibilitas dan dukungan yang tinggi yang dikarenakan keterlibatan pengguna. 4. Solusi berbasis <i>software</i> lebih cepat dibandingkan solusi berbasis bisnis model. 5. Kesalahan dan kelalaian dapat diketahui lebih cepat pada <i>prototype</i> ini dibandingkan pada <i>system model</i>. 	<ol style="list-style-type: none"> 1. Beberapa isu yang mengatakan RAD lebih membutuhkan banyak tenaga, dukungan, dan pengembangan yang menyebabkan meningkatnya biaya yang diperlukan. 2. RAD dapat menyelesaikan masalah yang keliru jika masalah tersebut diabaikan dan disingkat. 3. <i>Prototype</i> berbasis RAD dapat memungkinkan berkurangnya semangat dari <i>analysts</i>. 4. Terkadang para <i>stakeholder</i> melihat <i>prototype</i> sebagai pembuangan tenaga dan waktu. 5. Menekankan pada kecepatan penyelesaian proyek dapat berdampak buruk bagi kualitas produk.

Tabel 2.1 Keuntungan dan Kerugian *Rapid Application Development* (lanjutan)

(Sumber : Whitten & Bentley, 2007, p100)

Keuntungan	Kerugian
<p>6. Uji coba adalah suatu keharusan dari suatu produk yang mendasari dari pendekatan <i>prototype</i>.</p> <p>7. Pendekatan secara iterative lebih wajar karena perubahan adalah factor yang diharapkan dalam proses pengembangan.</p>	

2.2.6 Framework

Framework diterjemahkan sebagai ‘kerangka kerja’. Secara garis besar *framework* adalah menyediakan kerangka kerja untuk membangun suatu sistem sehingga pembangun sistem tidak harus merancang sistem dari awal. Dalam konsep *web framework*, *framework* menyediakan berbagai fungsi, sintaksis, *library*, *extension*, dan *template* siap pakai untuk mempercepat pekerjaan membangun *website*. Banyak jenis kerangka kerja ditemukan termasuk: Django, Yii, Zend, dan juga Laravel. Aplikasi situs web, harus mengikuti aturan kerangka kerja, *framework* (dalam hal ini PHP *framework*), tidak wajib memikirkan kode perintah dasar / fungsi aplikasi *website*.

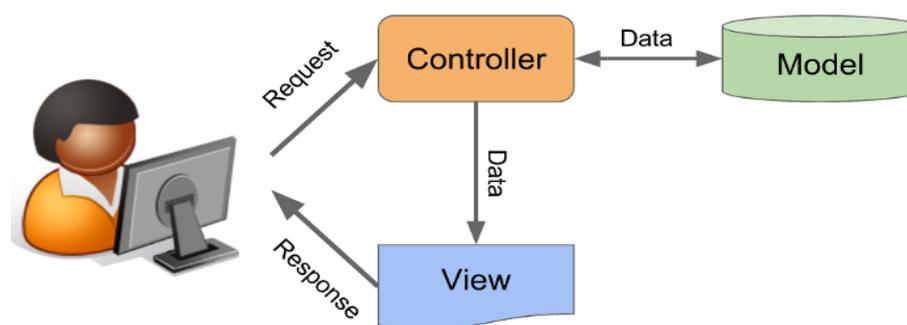
2.2.6.1 Sejarah Laravel

Laravel sendiri diciptakan oleh Taylor Otwell, dia memulai proyek laravel pada tahun 2011 dibulan April. Proyek ini didasari karena Otwell tidak menemukan *framework* yang mempunyai kelebihan update sejalan dengan versi PHP,

menurutnya mengembangkan *framework* yang telah ada tidak mudah dengan keterbatasan sumber daya. Oleh karena itu Otwell membuat *framework* laravel dengan memberikan ketentuan yaitu PHP versi 5.4 ke atas.

2.2.6.2 Laravel

Laravel merupakan *framework* PHP dengan ciri open source. Dengan deesain *Model View Controller* (MVC) yang dipakai agar membangun aplikasi *website*. *Framework* ini pertama sekali diciptakan oleh Taylor Otwell pada tanggal 22 Febuari 2021. Selain iitu ada juga yang beranggapan bahwa, Laravel ialah pengembangan *website* berlandaskan MVP yang ditulis dalam PHP yang dirancang untuk meningkatkan kualitas *software* dengan mengurangi cost pengembangan awal, biaya perawatan, seta untuk mengoptimalkan pengalaman bekerja dengan applikasi dengan menyediakan sintaks ekspresif, jelas dan efisien. Pembuatan aplikasi dapat dilihat pada struktur dan model dari *framework* Laravel adalah *model*, *view*, dan *controller* (Yudhanto & Prasetyo, 2018),



Gambar 2.3 Model *framework* Laravel

Dalam konsep MVC (*Model-View-Controller*) untuk model berisi *coding* untuk pengelolaan basis data tetapi tidak berkoneksi dengan *view* yang merupakan file dalam script pada HTML, sedangkan *controller* adalah penghubung *view* dan mode.