

**PENGEMBANGAN SISTEM INFORMASI MANAJEMEN ARSIP STATIS (SIMARS-PLUS) BERBASIS
WEB DENGAN PENDEKATAN EXTREME PROGRAMMING
PADA DINAS PERPUSTAKAAN DAN KEARSIPAN PROVINSI DKI JAKARTA TAHUN ANGGARAN 2018**

Zulhalim

Teknik Informatika
STMIK Jayakarta

zulhalim@stmik.jayakarta.ac.id

Abstrak

Saat ini Dinas Perpustakaan dan Kearsipan (Dispusip) Provinsi DKI Jakarta telah memiliki Sistem Informasi Manajemen Arsip Statis (SIMARS Plus) berbasis aplikasi desktop yang telah diimplentasikan pada kantor pusat Dispusip provinsi DKI Jakarta. SIMARS-Plus berbasis desktop tidak bisa diimplentasikan pada semua kantor suku dinas Dispusip provinsi DKI Jarkarta mengingat banyaknya infrastruktur pendukung yang harus disiapkan untuk menjalankan SIMARS-Plus berbasis desktop. Penelitian ini bertujuan untuk mengembangkan aplikasi SIMARS-Plus berbasis web yang dapat digunakan pada kelima kantor suku dinas Dispusip provinsi DKI Jakarta. Sedangkan manfaat yang didapatkan dari hasil penelitian ini adalah terciptanya konsolidasi data arsip yang merupakan gabungan data arsip dari kantor pusat dan kelima kantor suku dinas Dispusip provinsi DKI Jakarta. Pendekatan pengembangan aplikasi menggunakan *Extreme Programming* (XP) yang dapat mempercepat proses pengembangan pada proyek dengan kebutuhan fungsional yang fleksibel. Sedangkan untuk framework pemrograman menggunakan *Single Page Application* (SFA) sebagai framework yang ringan dan cocok digunakan pada pengolahan kuantitas metadata arsip yang banyak dan file gambar arsip yang besar.

Kata Kunci: Sistem Informasi Manajemen Arsip Statis, Extreme Programming, Single Page Application, Dinas Perpustakaan dan Kearsipan provinsi DKI Jakarta

I. PENDAHULUAN

A. Latar Belakang

Sesuai dengan amanat Undang-Undang Nomor 43 Tahun 2009 tentang Kearsipan, bahwa dalam menghadapi tantangan globalisasi dan mendukung terwujudnya penyelenggaraan negara dan khususnya pemerintahan yang baik dan bersih, serta peningkatan kualitas pelayanan publik, penyelenggaraan kearsipan di lembaga negara, pemerintahan daerah, lembaga pendidikan, perusahaan, organisasi politik, organisasi kemasyarakatan, dan perseorangan harus dilakukan dalam suatu penyelenggaraan kearsipan nasional yang komprehensif dan terpadu.[1]

Arsip daerah provinsi adalah lembaga kearsipan berbentuk satuan kerja perangkat daerah yang melaksanakan tugas pemerintahan di bidang kearsipan pemerintahan daerah provinsi yang berkedudukan di ibukota provinsi. Lingkup penyelenggaraan kearsipan sesuai dengan Undang-Undang Nomor 43 Tahun 2009 tentang Kearsipan, meliputi keseluruhan penetapan kebijakan, pembinaan kearsipan, pengelolaan arsip dalam suatu sistem kerarsipan nasional yang didukung oleh sumber daya manusia, prasarana dan sarana, serta sumber daya lain dengan ketentuan peraturan perundang-undangan, maka penyelenggaraan

kearsipan provinsi menjadi tanggung jawab pemerintahan daerah provinsi dan dilaksanakan oleh Dinas Perpustakaan dan Kearsipan (Dispusip) Provinsi DKI Jakarta.[1]

Saat ini Dinas Perpustakaan dan Kearsipan Provinsi DKI Jakarta telah memiliki Sistem Informasi Manajemen Arsip Statis (SIMARS Plus) berbasis aplikasi desktop merupakan sarana pendukung pengelolaan arsip yang berbasis teknologi informasi dengan komponen utama adalah komputer. Namun dalam peiaksanaannya penerapan SIMARS-Plus di Dinas Perpustakaan dan Kearsipan Provinsi DKI Jakarta masih jauh dari harapan yang diinginkan oleh para stake holder.

B. Permasalahan

Berdasarkan hasil proses analisis pada penelitian ini maka dapat diidentifikasi masalah yang terjadi pada Dispusip provinsi DKI Jakarta yaitu:

- 1) Aplikasi SIMARS-Plus berbasis desktop diimplementasikan dan digunakan untuk pengolahan data arsip yang hanya tersimpan pada kantor pusat Dispusip DKI Jakarta.
- 2) Pada kantor suku dinas Dispusip provinsi DKI Jakarta pada proses pengolahan data arsip masih menggunakan Microsoft Excell sebagai

penyimpanan metadata arsip dan harddisk eksternal sebagai penyimpanan data arsip.

- 3) Aplikasi SIMARS-Plus berbasis desktop tidak bisa diimplementasikan pada setiap kantor suku dinas Dispusip provinsi DKI Jakarta mengingat banyaknya infrastruktur pendukung yang harus disiapkan untuk menjalankan SIMARS-Plus berbasis desktop.

D. Ruang Lingkup Penelitian

Penelitian ini mempunyai batasan ruang lingkup sebagai berikut:

- 1) Aplikasi yang dihasilkan dari penelitian ini merupakan SIMARS-Plus berbasis web dimana dapat dipakai oleh ke 5 (lima) kantor suku dinas Dispusip provinsi DKI Jakarta.
- 2) Spesifikasi proses dan sistem database aplikasi yang dihasilkan sama dengan spesifikasi yang ada pada aplikasi SIMARS-Plus berbasis Desktop.
- 3) Metode pengembangan aplikasi menggunakan pendekatan *Extreme Programming* yang dapat mempercepat proses pengembangan pada proyek dengan kebutuhan fungsional yang fleksibel.

C. Tujuan dan Manfaat

Adapun tujuan yang harus dicapai pada hasil penelitian ini adalah mengembangkan aplikasi SIMARS-Plus berbasis desktop yang telah digunakan menjadi aplikasi SIMARS-Plus berbasis web yang dapat digunakan pada kelima kantor Suku Dinas Dispusip provinsi DKI Jakarta. Sedangkan manfaat yang didapatkan dari hasil penelitian ini adalah terciptanya konsolidasi data arsip yang merupakan gabungan data arsip dari kantor pusat dan kelima kantor suku dinas Dipusip provinsi DKI Jakarta.

II. LITERATUR DAN METODE

A. Arsip Statis

Arsip adalah rekaman kegiatan atau peristiwa dalam berbagai bentuk dan media sesuai dengan perkembangan teknologi informasi dan komunikasi yang dibuat dan diterima oleh lembaga negara, pemerintahan daerah, lembaga pendidikan, perusahaan, organisasi politik, organisasi kemasyarakatan, dan perseorangan dalam pelaksanaan kehidupan bermasyarakat, berbangsa, dan bernegara. [1]

Dalam undang-undang tersebut juga dijelaskan bahwa menurut fungsinya, arsip dibedakan menjadi dua, yakni arsip dinamis dan arsip statis. Arsip dinamis adalah arsip yang masih digunakan secara

langsung dalam perencanaan, pelaksanaan, dan penyelenggaraan kehidupan kebangsaan pada umumnya atau dalam administrasi negara. Sedangkan arsip statis adalah arsip yang tidak secara langsung dipergunakan. Dengan demikian arsip dinamis merupakan arsip yang masih ada di kantor, baik kantor pemerintah, swasta, maupun organisasi kemasyarakatan karena masih dipakai dalam perencanaan, pelaksanaan dan kegiatan administrasi lainnya.

Arsip statis berasal dari arsip dinamis yang dinyatakan mempunyai nilai guna sekunder. Nilai guna sekunder yakni nilai arsip yang didasarkan pada kegunaan arsip bagi kepentingan lembaga lain dan atau kepentingan umum diluar lembaga pencipta arsip. Nilai guna sekunder meliputi nilai guna kebuktian dan nilai guna informasional. Disamping karena nilai guna sekunder tersebut, suatu arsip dapat menjadi permanen karena nilai guna hakiki yang melekat pada arsip tersebut. Nilai guna hakiki disebabkan oleh keunikan sebuah arsip, misalnya perihal isinya, usia arsip tersebut, dan lain-lain. [2]

Untuk sampai menjadi arsip statis memerlukan rangkaian proses dalam manajemen arsip dinamis. Setiap arsip dinamis akan mengalami fase penjadwalan yang disebut jadwal retensi arsip (JRA). JRA menyatakan umur arsip. Setelah arsip jatuh tempo atau telah habis umurnya sesuai dengan penjadwalan yang telah dibuat, kemudian diadakan penilaian arsip.

Untuk menentukan jadwal retensi, manajer arsip dinamis minta bantuan berbagai bagian, misalnya bagian yang menciptakan arsip dinamis guna mengetahui sumbangan arsip dinamis bagi kegiatan badan korporasi, bagian hukum untuk menentukan periode retensi sesuai dengan ketentuan perundang-undangan, bagian keuangan guna mempertanggungjawabkan arsip dinamis dari segi keuangan. Di Indonesia, JRA biasanya dibuat oleh badan korporasi yang bersangkutan, seringkali dengan bantuan arsiparis dari Arsip Nasional. Hal ini karena tidak semua badan korporasi memiliki arsiparis profesional. [3]

Setelah arsip tiba jatuh temponya, kemudian diadakan penilaian arsip. Penilaian ini untuk menilai apakah suatu arsip mempunyai nilai guna sekunder dan hakiki atau tidak. Apabila tidak, maka arsip tersebut akan dimusnahkan. Untuk itu, diperlukan adanya pengetahuan yang luas dan intuisi yang tinggi dari seorang penilai, sebab apabila suatu arsip telah dinyatakan musnah maka tidak akan ditemukan lagi penggantinya. Dari penilaian ini akan menghasilkan dua hal, yakni arsip simpan permanen atau musnah.

Arsip simpan permanen ini kemudian diserahkan kepada lembaga kearsipan untuk dilestarikan sebagai arsip statis.

Semua arsip statis terlebih dahulu dilakukan pendataan setiap itemnya dan diadakan penataan sebelum dapat dilayankan ke publik. Di dalam penataan arsip terdapat beberapa prinsip yang harus digunakan. Prinsip-prinsip tersebut adalah:[4]

- 1) prinsip asal-usul (*principle of provenance*),
- 2) aturan asli (*original order*),
- 3) kegunaan (*bestemming beginsel*),
- 4) prinsip restorasi,
- 5) prinsip fungsional,
- 6) prinsip organisasi dan
- 7) prinsip masalah.

Arsip-arsip yang telah dideskripsikan tersebut kemudian diadakan penyampulan, penomoran dan pengebokan. Langkah berikutnya adalah penyimpanan arsip kedalam rak arsip statis. Rak arsip statis ini berbeda dengan pentimpanan arsip sebelum menjadi statis. Rak statis tertutup dan biasa disebut rol opak. Maksud dari pentimpanan yang tertutup ini untuk menjaga keamanan arsip. Untuk itu, bentuk media arsip yang berbeda-beda juga akan menentukan cara penyimpanannya. Misalnya, arsip dengan media kertas tekstual, foto, mikrofilm, CD, masing-masing berbeda cara penyimpanan, perawatan dan juga pengolahannya.

B. Sistem Informasi Manajemen Arsip Statis

Pendataan per item arsip ini didasarkan pada standard pendeskripsian internasional yang dikeluarkan oleh ICA (*International Council on Archives* yang disebut dengan ISAD (G) (*International Standard on Archival and Description (General)*). Deskripsi arsip ini dimaksudkan untuk memberikan informasi mengenai [4]:

- a) Asal-usul, isi dan sumber dari beerbagai kumpulan arsip
- b) Struktur pemberkasannya
- c) Hubungannya dengan arsip lain
- d) Cara bagaimana arsip tersebut dapat ditemukan dan digunakan.

Deskripsi tersebut disusun secara bertingkat sehingga disebut *multi level description*, yang seluruhnya memuat 26 ketentuan pendeskripsian. Akan tetapi, dari 26 ketentuan tersebut tidak semuanya harus dipergunakan. Suatu desripsi arsip statis yang lebih sederhana paling tidak memuat tentang lima unsur, yakni:

- 1) bentuk redaksi; yang menyatakan jenis, format naskah yang digunakan dalam merekam informasi

pada sebuah arsip misalnya surat, laporan, nota dinas, dan sebagainya.

- 2) Isi; informasi yang terkandung dalam arsip yang ditulis secara singkat dan padat.
- 3) Tanggal; periode atau kurun waktu terciptanya arsip.
- 4) Tingkat perkembangan; menyatakan identitas tingkat keaslian arsip, misalnya konsep, tembusan, tindasan, asli, copy, salinan, dan sebagainya.
- 5) Bentuk luar; yang menyatakan jumlah volume arsip yang dideskripsi, misalnya lembar, sampul, bundel, dan sebagainya.

Kemudian, daftar arsip tersebut terkumpul dalam sebuah daftar arsip sebagai sarana jalan masuk pencarian arsip.

Perkembangan sistem informasi membuat jalan masuk pencarian arsip tersebut dapat dilakukan secara otomasi. Berikutnya, dalam rangka pelestarian arsip, maka diadakan alih media arsip. Arsip-arsip tekstual, foto dan kartografi dialihmediakan dalam bentuk digital. Layanan arsip kemudian dilakukan melalui media akses sehingga arsip asli dalam wujudnya semula masih tetap terjaga. Hal ini karena tingkat sirkulasi fisik arsip mempengaruhi keawetan sebuah arsip. Dengan layanan melalui media akses, yakni hasil alih media arsip, membuat arsip asli lebih terjaga. Demikian pula sistem penelusuran arsip akan lebih mudah dan cepat.

C. Extreme Programming

Extreme Programming (XP) adalah sebuah pendekatan atau model pengembangan perangkat lunak yang mencoba menyederhanakan berbagai tahapan dalam proses pengembangan tersebut sehingga menjadi lebih adaptif dan fleksibel. XP bukan hanya berfokus pada coding tetapi meliputi seluruh area pengembangan perangkat lunak. XP mengambil pendekatan 'ekstrim' dalam *iterative development*. [5]

XP pertama kali diusulkan oleh Kent Beck dan Ward Cunningham pada bulan Maret 1996, asal mula XP digunakan karena pada saat itu permintaan dari Pelanggan yang sering berubah dengan cepat sehingga mengakibatkan putaran kehidupan metode pengembangan perangkat lunak tradisional menjadi lebih pendek dan tidak selaras dengan metode tradisional karena pada umumnya memerlukan desain yang luas dan itu mengakibatkan perubahan desain yang terjadi dan tentu saja memerlukan biaya yang lebih tinggi. Tujuan XP adalah meminimalisasi biaya yang diperlukan jika ada perubahan dalam pengembangan perangkat lunak. [5]

Aspek dasar XP terdiri dari berbagai teknik atau metode yang diterapkan Beck dan Jeffries pada C3 project, teknik-teknik tersebut antara lain: [6]

- **Whole Team**

Seluruh kontributor dalam proyek yang menggunakan pendekatan XP duduk bersama sebagai suatu tim. Tim ini terdiri beberapa peran, antara lain programmer, pengujian, orang yang mengerti bisnis, analis, manajer, dan lain-lain. Setiap peran yang ada tidak mutlak menjadi peran dari satu orang saja. Tim terbaik dalam XP tidak harus memiliki pakar, hanya kontributor umum dengan keterampilan khusus saja. Semua orang di tim XP memberikan kontribusi dengan cara apapun yang mereka dapat lakukan.

- **Planning game**

Perencanaan dalam XP mengemukakan dua pertanyaan kunci dalam pengembangan perangkat lunak, yaitu memprediksi apa yang akan dicapai pada waktu tertentu, dan menentukan apa yang harus dilakukan setelah itu. Ada dua langkah kunci dalam perencanaan XP, yang menangani dua pertanyaan tersebut: Release Planning yaitu praktek dimana Pelanggan mengutarakan fitur yang diinginkan ke programmer, dan programmer memperkirakan tingkat kesulitannya. Dengan estimasi biaya di tangan, dan dengan pengetahuan tentang pentingnya fitur yang diinginkan, Pelanggan meletakkan satu rencana untuk proyek tersebut. Rencana rilis awal yang selalu tepat: baik prioritas maupun perkiraan yang benar-benar solid, dan sampai tim mulai bekerja, kita tidak akan tahu seberapa cepat mereka akan pergi. Bahkan rencana rilis pertama cukup akurat untuk pengambilan keputusan, namun, dan tim XP melakukan revisi terhadap rencana rilis secara teratur. 2. Iteration Planning adalah praktek di mana tim diberikan petunjuk atau arahan setiap beberapa minggu sekali. Tim XP membangun perangkat lunak dalam "iterasi" dua minggu, memberikan menjalankan perangkat lunak yang berguna pada setiap akhir iterasi. Selama Iteration Planning, Pelanggan mengutarakan fitur yang diinginkan selama dua minggu ke depan. Para programmer memecahnya ke dalam pekerjaan yang lebih kecil, dan memperkirakan biaya yang diperlukan.

- **Customer Tests**

Sebagai bagian dari presentasi masing-masing fitur yang diinginkan, Pelanggan XP mendefinisikan satu atau lebih tes penerimaan otomatis untuk menunjukkan bahwa fitur tersebut bekerja dengan baik. Tim membangun tes ini dan

menggunakannya untuk membuktikan pada kepada Pelanggan bahwa fitur ini telah diimplementasikan dengan benar. Tes secara otomatis ini penting karena dalam XP hanya diberikan waktu yang singkat sehingga tes manual tidak akan digunakan karena memakan waktu yang lama.

- **Small Release**

Pada setiap Iterasi, tim mengerjakan sebuah unit atau bagian dari perangkat lunak, melakukan tes terhadap unit perangkat lunak yang dibangun, kemudian di akhir iterasi perangkat lunak yang dibangun diberikan kepada Pelanggan. Oleh Pelanggan, perangkat lunak ini bisa dijadikan bahan evaluasi maupun langsung dirilis kepada *end user*. Bisa juga tim XP langsung merilis ke *end user* secara rutin.

- **Simple Design**

Tim XP membangun perangkat lunak dengan desain yang sederhana. Dimulai dengan desain yang sederhana, kemudian melalui pengujian program dan perbaikan desain. Desain yang dibuat harus benar-benar cocok untuk fungsi saat ini dari sistem sehingga tidak ada yang sia-sia dan perangkat lunak siap dikembangkan lagi selanjutnya. Namun, pembuatan desain dalam XP tidak dilakukan hanya sekali. Tahapan desain dalam *Extreme Programming* yang menghasilkan desain yang bagus dianggap sangat penting, sehingga selama proses *development* banyak difokuskan ke tahapan desain.

- **Pair Programming**

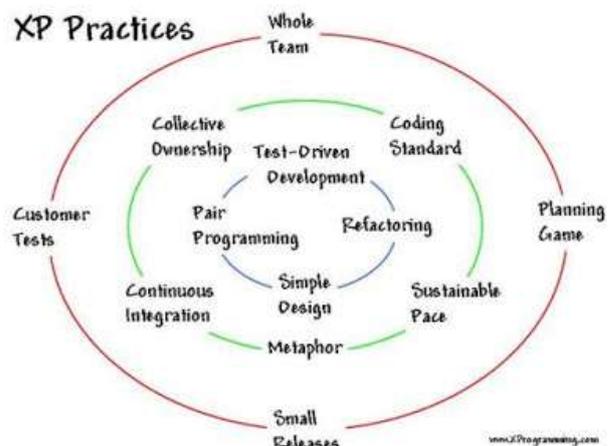
Semua perangkat lunak yang dibangun dengan pendekatan XP dibangun oleh dua orang programmer. Keduanya duduk berdampingan di satu komputer yang sama. Seorang programmer akan membuat code dan programmer yang lainnya akan mengoreksinya. Praktik seperti ini mungkin kelihatan tidak efisien. Namun dari segi hasil dari pair programming, desain akan lebih baik, pengujian lebih baik, dan code yang dihasilkan pun akan lebih baik.

- **Test-Driven Development**

XP begitu terobsesi dengan umpan balik, dan dalam pengembangan perangkat lunak, umpan balik yang baik mensyaratkan pengujian yang baik pula. *Test-Driven Development* bergantung pada pengulangan siklus development yang sangat pendek. Pertama tim XP akan menuliskan *automated test case* yang mendefinisikan perbaikan yang diinginkan atau fungsi baru. Kemudian dari *test case* tersebut dihasilkan jumlah minimal code yang harus dituliskan untuk

lulus tes tersebut. Setelah itu melakukan *refactoring code* baru agar memenuhi standar baru.

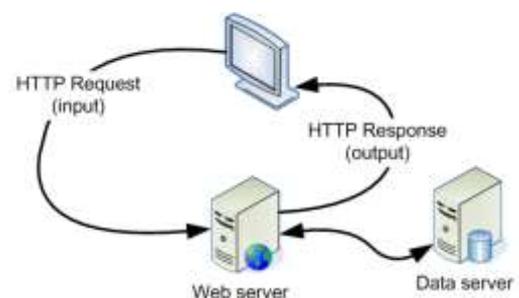
- **Design Improvement**
XP berfokus pada memberikan nilai bisnis dalam setiap perulangan. Agar dapat mencapai tujuan tersebut selama proyek berlangsung, perangkat lunak harus dirancang dengan baik. XP menggunakan proses perbaikan desain secara terus menerus dengan *Refactoring*. Proses *refactoring* berfokus pada penghapusan duplikasi dari code yang telah dibuat. Disamping itu, proses *refactoring* didukung dengan pengujian yang komprehensif untuk memastikan bahwa desain yang dibuat berkembang dan tidak ada yang rusak.
- **Continuous Integration**
Beberapa kali dalam sehari, tim XP akan menggabungkan seluruh salinan pekerjaan tim menjadi satu dalam jaringan utama. Sehingga tim XP harus menjaga tim agar terintegrasi setiap saat.
- **Collective Code Ownership**
Pada proyek XP, setiap pasang programmer dapat meningkatkan code apapun setiap saat. Semua code yang ada dimiliki secara kolektif oleh tim. Manfaatnya setiap code akan mendapat perhatian dari banyak orang, sehingga dapat meningkatkan kualitas code dan mengurangi cacat. Selain itu dapat mengurangi duplikasi code yang sama walaupun dibuat oleh pasangan programmer yang berbeda.
- **Coding Standard**
Setiap anggota tim XP harus mengikuti standar coding yang umum, sehingga semua code dalam sistem seolah-olah tampak dibuat oleh satu orang yang sangat kompeten. Selain itu hal ini sangat mendukung *Collective Code Ownership*.
- **Metaphor**
Tim XP akan membuat suatu deskripsi umum bagaimana program yang mereka kembangkan bekerja dengan benar.
- **Sustainable Pace**
Tim XP akan bekerjasama dalam jangka waktu lama. Mereka bekerja keras dengan kecepatan tertentu tanpa batas waktu. Tim XP akan bekerja lembur pada hari efektif dan memaksimalkan produktivitas setiap minggunya. Hal ini perlu diperhatikan dengan baik, karena akan mengurangi produktivitas atau sebaliknya menghasilkan perangkat lunak yang berkualitas.



Gambar 2.1 Praktik Extreme Programming

D. Aplikasi Berbasis Web

Aplikasi web merupakan sebuah aplikasi yang menggunakan teknologi browser untuk menjalankan aplikasi dan diakses melalui jaringan komputer [7]. Dari pengertian diatas dapat disimpulkan aplikasi web merupakan aplikasi yang diakses menggunakan web browser melalui jaringan internet atau intranet. Aplikasi web juga merupakan suatu perangkat lunak komputer yang dikodekan dalam bahasa pemrograman yang mendukung perangkat lunak berbasis web seperti HTML, JavaScript, CSS, Ruby, Python, Php, Java dan bahasa pemrograman lainnya.



Gambar 2.2 Alur data aplikasi web

Keunggulan dari aplikasi berbasis web adalah:

- 1) Kita dapat menjalankan aplikasi berbasis web dimanapun kapanpun tanpa harus melakukan penginstalan.
- 2) Terkait dengan isu lisensi (hak cipta), kita tidak memerlukan lisensi ketika menggunakan *web-based application*, sebab lisensi telah menjadi tanggung jawab dari web penyedia aplikasi.
- 3) Dapat dijalankan di sistem operasi manapun. Tidak peduli apakah kita menggunakan linux,

windows, aplikasi berbasis web dapat dijalankan asalkan kita memiliki browser dan akses internet.

- 4) Dapat diakses lewat banyak media seperti: computer dan smartphone.
- 5) Tidak perlu spesifikasi komputer yang tinggi untuk menggunakan aplikasi berbasis web ini, sebab di beberapa kasus, sebagian besar proses dilakukan di web server penyedia aplikasi berbasis web ini.

Sedangkan kekurangan dari aplikasi web adalah:

- 1) Dibutuhkan koneksi intranet dan internet yang handal dan stabil, hal ini bertujuan agar pada saat aplikasi dijalankan akan berjalan dengan baik dan lancar.
- 2) Dibutuhkan sistem keamanan yang baik dikarenakan aplikasi dijalankan secara terpusat, sehingga apabila server di pusat terganggu maka sistem aplikasi tidak bisa berjalan.

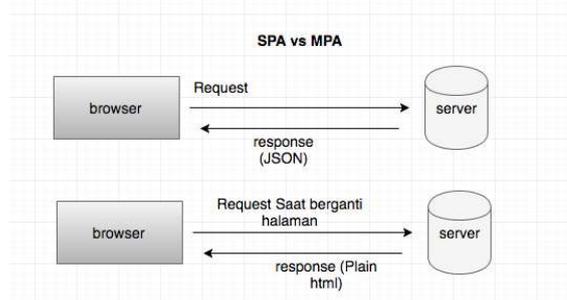
E. Single Page Application

Single Page Application (SPA) adalah salah satu jenis aplikasi website dimana hanya ada 1 halaman yang meng-handle semua aktivitas yang terjadi dalam aplikasi tersebut. User tidak akan berpindah halaman browser ketika melakukan request seperti mengisi formulir, klik link, atau aksi lain yang membutuhkan data dari server. Semua aksi-aksi tersebut akan di-handle secara *Asynchronous oleh Javascript* (seperti AJAX). Perpindahan 1 halaman ke halaman lain di handle dengan sistem *routing*, tanpa loading browser sehingga prosesnya lebih cepat dan lebih baik UX-nya. [8]

SPA sangat bergantung pada utilitas browser karena menggunakan javascript sepenuhnya. Sementara untuk berkomunikasi terhadap backend tetap menggunakan http request terhadap suatu API/Web services. Teknologi SPA berkembang seiring dengan kebutuhan industri pada aplikasi website yang dapat bekerja lebih cepat dan hemat resource sehingga akan menghemat pengeluaran untuk server. Saat ini, website SPA umumnya digunakan pada aplikasi dashboard atau aplikasi sejenis yang mana pada satu halaman membutuhkan banyak informasi dan data dari proses bisnis, transaksi, perhitungan, dan sebagainya.

Yang membedakan SPA dari *Multi Page Application* (MPA), adalah bagaimana SPA melakukan request dan resource. SPA lebih hemat dibanding dengan *Multi Page Application*. Contohnya, pada saat berpindah ke halaman baru, SPA hanya me-request data-data yang diperlukannya saja berupa JSON (*Javascript Object Notation*), tanpa me-request

ulang static files seperti file css, html, dan js nya karena itu sudah di load saat pertama kali.



Gambar 2.3 SPA vs MPA

Berbeda dengan MPA yang setiap perpindahan halaman akan merequest ulang seluruh elemen baik itu static files dan data baru untuk halaman tersebut. Contoh website yang menggunakan SPA antara lain gmail, facebook, github, techinasia dan lain sebagainya. Untuk MPA contohnya adalah wikipedia, kaskus, bukalapak dan lain-lain.

Mekanisme yang berjalan pada SPA layaknya kita menggunakan browser secara native dimana secara penuh memanfaatkan perpindahan halaman, data binding, form, dan lain sebagainya dengan menggunakan javascript. Tidak ada waktu menunggu saat terjadi perpindahan halaman. Hal seperti ini memberikan UX (*User experience*) yang luar biasa kepada user.

Keunggulan penggunaan SPA pada aplikasi web adalah:

- 1) Lebih cepat dibandingkan dengan multi page application. Kebanyakan resource seperti HTML, CSS dan JS diload pada saat pertama kali aplikasi dibuka dan akan berfungsi terus sampai aplikasi ditutup. Perpindahan halaman tidak perlu melakukan request keseluruhan elemen halaman, tetapi hanya data tertentu saja yang belum di request saat inisiasi pertama kali. Ada konsep "*virtual DOM*" dimana perubahan pada DOM hanya terjadi pada elemen yang berubah. Sedangkan pada MPA, DOM akan diperbarui secara menyeluruh karena perpindahan halaman harus melakukan request kepada server dan server memberikan respon berupa html.
- 2) Tidak perlu mensetup "server". SPA hanya memiliki satu buah index.html, beberapa css dan beberapa javascript. Sehingga kita cukup drag index.html tersebut untuk menjalankan aplikasi SPA. Namun tentunya ini sangat dasar. Kita tetap membutuhkan server jika ingin mengoptimasi performa dari SPA itu sendiri.

- 3) Proses debug sangat mudah cukup dengan menggunakan browser (*console*). Jika terjadi error cukup refresh pada browser maka akan tampil errornya di console tidak perlu melalui proses compile. Pada SPA kompilasi akan dilakukan diakhir saat seluruh aplikasi telah selesai dikerjakan (*build*) dengan menggunakan webpack dan babel (untuk transpile) untuk menghasilkan *index.html*, file-file css serta file-file javascript yang telah digabungkan menjadi satu.
- 4) Satu kode untuk berbagai macam platform. Dengan javascript kita dapat membuat website, desktop, android, iOS, Windows phone. Terdapat beberapa framework untuk membangun mobile apps dengan javascript antara lain react native dan ionic (berbasis angular).
- 5) Proses Caching lebih efektif karena memanfaatkan local storage pada browser. Sebuah aplikasi dapat mengirim hanya 1 request, lalu responnya di simpan dalam local storage sehingga dapat digunakan kembali bahkan dalam keadaan offline sekalipun.

Sedangkan kekurangan penggunaan SPA pada aplikasi web adalah:

- 1) Tidak bagus dalam hal *Search Engine Optimization* (SEO). Optimasi SEO pada SPA sangat sulit dilakukan karena sejatinya hanya ada 1 (satu) halaman yang bisa diindex oleh mesin pencari, serumit dan sebanyak apapun data dari aplikasi tersebut. Route pada SPA tidak bisa dijadikan sebagai URL yang bisa diindex pada mesin pencari. Saat ini, mulai dikembangkan tools SPA untuk keperluan SEO. Metodenya adalah mem-build SPA secara *server-side* sehingga setiap route yang dihasilkan akan menjadi URL. Contoh tools ini adalah *Angular Universal* dan *Server-side Rendering* pada VueJS.
- 2) Berat saat di-load/buka pertama kali. SPA merequest kebanyakan resourcenya saat pertama kali aplikasi dibuka. Hal ini pastinya menyebabkan loading yang sangat lama.
- 3) Sangat bergantung pada Javascript. Semua aksi dalam SPA menggunakan javascript, sehingga ketika javascript di browser dimatikan oleh user aplikasi tidak akan berjalan sama sekali.
- 4) Kurang aman dibanding dengan Website biasa. SPA rawan terhadap serangan *Cross-site Scripting* (XSS), yang memungkinkan hacker *inject script client-side* (Javascript) pada aplikasi.
- 5) Masalah kompatibilitas browser. Karena SPA mengandalkan javascript dan browser, ketidakcocokan antaran browser satu dan lainnya

lebih beresiko terjadi. Ini bisa disebabkan karena user menggunakan browser lama, perbedaan versi javascript pada browser yang berbeda, dan sebagainya.

Jadi, bisa disimpulkan bahwa SPA tidak cocok untuk website-website yang membutuhkan SEO yang bagus, seperti blog. SPA cocok untuk aplikasi website seperti dashboard, situs media sosial, dan sejenisnya.

III. METODE

A. Metode Penelitian

Penelitian ini merupakan penelitian deskriptif dengan pendekatan kualitatif. Penelitian deskriptif mendeskripsikan gambaran atau lukisan secara sistematis, faktual dan akurat mengenai fakta-fakta, sifat-sifat hubungan antara fenomena yang diselidiki. Metode kualitatif yaitu dimana diusahakan untuk mencari gambaran dan penjelasan mengenai permasalahan yang dibahas [9].

Sumber data primer, merupakan data yang diperoleh dari pengguna aplikasi SIMARS-Plus versi desktop pada Bidang Pelayanan Kearsipan dan bidang Teknologi Informasi Dispusip provinsi DKI Jakarta dengan memberikan sejumlah pertanyaan sebagai instrumen penelitian. Data Sekunder, yaitu data yang diperoleh untuk melengkapi data primer berupa dokumen-dokumen atau laporan yang dapat mendukung pembahasan dalam kaitannya dengan penelitian ini.

Pengumpulan data adalah cara-cara untuk memperoleh data-data yang lengkap, objektif dan dapat dipertanggung jawabkan kebenarannya sesuai dengan permasalahan penelitian yang berkaitan dengan persepsi masyarakat tentang ketersediaannya sistem informasi dengan kebutuhan informasi di Dispusip. Adapun teknik pengumpulan data yang digunakan dalam penelitian ini adalah:

- 1) Observasi, observasi merupakan suatu proses yang kompleks, suatu proses yang tersusun dari pelbagai proses biologis dan psikologis[9].
- 2) Wawancara, adalah pertemuan dua orang untuk bertukar informasi dan ide melalui tanya jawab, wawancara langsung atau bertatap muka terhadap informan agar menjawab pertanyaan pertanyaan lisan maupun tulisan yang berkaitan dengan objek penelitian[9].
- 3) Dokumentasi, yaitu cara mengetahui sesuatu dengan melihat kode sumber aplikasi, catatan-catatan, arsip-arsip, dokumen-dokumen yang berhubungan dengan informasi yang dibutuhkan dalam penelitian.

Instrumen penelitian adalah alat yang digunakan dalam melaksanakan penelitian yang sesuai dengan metode yang digunakan. Adapun instrumen yang digunakan peneliti dalam mengumpulkan data antara lain:

- 1) Pedoman Wawancara, adalah proses tanya jawab dalam penelitian yang berlangsung secara lisan di mana wawancara tersebut dilakukan terhadap responden yang dianggap mampu memberikan data tentang apa yang sedang diteliti dengan menggunakan alat recorder smartphone.
- 2) Kamera SmartPhone, alat yang digunakan untuk mendokumentasi-kan data penelitian berbentuk gambar.
- 3) Recorder SmartPhone, digunakan untuk merekam saat wawancara berlangsung.

Data yang telah terkumpul akan mempunyai arti setelah diolah dan dianalisis dengan menggunakan beberapa teknik deskriptif kualitatif dalam bentuk naratif yaitu hasil data observasi, wawancara, dan dokumentasi tersebut dianalisis data kualitatif deskriptif adalah sebagai berikut [9].

- 1) Reduksi Data yaitu data yang diperoleh dari lapangan yang banyak dan kompleks maka perlu dilakukan analisis data melalui reduksi data. mereduksi data dengan cara merangkum, memilih hal-hal pokok, memfokuskan hal-hal yang penting dan membuang hal-hal yang dianggap kurang penting. Dengan demikian data yang direduksi dapat memberi gambaran yang jelas bagi penulis untuk mendapat data selanjutnya.
- 2) Penyajian Data yaitu data yang sudah direduksi disajikan dalam bentuk uraian singkat berupa teks yang bersifat naratif. Melalui penyajian data tersebut maka data akan mudah dipahami sehingga memudahkan rencana kerja selanjutnya.
- 3) Penarikan Kesimpulan. Data yang sudah disajikan dianalisis secara kritis berdasarkan fakta-fakta yang diperoleh di lapangan. penarikan kesimpulan dikemukakan dalam bentuk naratif sebagai jawaban dari rumusan masalah yang dirumuskan sejak awal.

Berdasarkan tahapan yang sudah dijelaskan diatas, analisis data perlu dipahami komponennya agar dapat memahami makna dari penelitian yang diteliti. Tahapan analisis data yang diperoleh penulis dari lapangan melalui observasi, wawancara, dan dokumentasi yang nantinya dapat ditarik kesimpulan dari penelitian tersebut. Setelah melakukan tahap-tahap diatas dapat diketahui bagaimana

pengembangan SIMARS-Plus versi web pada Dispusip provinsi DKI Jakarta .

B. Metode Pengembangan

Metode pengembangan aplikasi ini menggunakan metode *Extreme Programming* (XP). XP menggunakan pendekatan berorientasi objek sebagai paradigma pengembangan yang disyaratkan dan mencakup seperangkat aturan dan praktik yang terjadi dalam konteks empat kegiatan kerangka kerja berikut ini. [5]

1) Perencanaan (*Planning*)

Kegiatan perencanaan dimulai dengan mendengarkan kebutuhan pelanggan dengan kegiatan pengumpulan kebutuhan yang memungkinkan anggota teknis tim XP untuk memahami konteks bisnis perangkat lunak dan untuk mendapatkan perkiraan yang luas untuk luaran yang dibutuhkan dan fitur serta fungsionalitas utama. Mendengarkan mengarah pada penciptaan serangkaian "*stories*" yang menggambarkan luaran, fitur, dan fungsionalitas yang diperlukan untuk perangkat lunak yang akan dibangun. Setiap *stories* ditulis oleh pelanggan dan ditempatkan pada kartu indeks (*index card*). Pelanggan memberikan nilai (yaitu prioritas) ke *stories* berdasarkan nilai bisnis keseluruhan dari fitur atau fungsi. Anggota tim XP kemudian menilai setiap *stories* dan menetapkan biaya (*cost* yang diukur dalam minggu pengembangan — untuk *stories* tersebut. Jika *stories* diperkirakan membutuhkan lebih dari tiga minggu pengembangan, pelanggan diminta untuk membagi *stories* menjadi *stories* yang lebih kecil dan penugasan nilai dan biaya terjadi lagi. Penting untuk dicatat bahwa *stories* baru dapat ditulis kapan saja.

2) Desain (*Design*)

Desain XP dengan ketat mengikuti prinsip KIS (*Keep It Simple*). Desain yang sederhana selalu lebih disukai daripada representasi yang lebih kompleks. Selain itu, desain tersebut memberikan panduan implementasi untuk sebuah *stories* seperti yang ditulis — tidak kurang, tidak lebih. Desain fungsi tambahan (karena pengembang menganggapnya akan diperlukan nanti) tidak disarankan. XP mendorong penggunaan kartu CRC sebagai

mekanisme yang efektif untuk berpikir tentang perangkat lunak dalam konteks orientasi objek. Kartu CRC (*class-responsibility-collaborator*) mengidentifikasi dan mengatur kelas berorientasi objek yang relevan dengan software increment saat ini. Kartu CRC adalah satu-satunya work product desain yang diproduksi sebagai bagian dari proses XP. Jika masalah desain yang sulit ditemui sebagai bagian dari desain sebuah stories, XP merekomendasikan pembuatan prototipe operasional dari bagian desain tersebut. Disebut *spike solution*, prototipe desain diimplementasikan dan dievaluasi. Tujuannya adalah untuk menurunkan risiko ketika implementasi yang sebenarnya dimulai dan untuk memvalidasi perkiraan awal untuk stories yang berisi masalah desain.

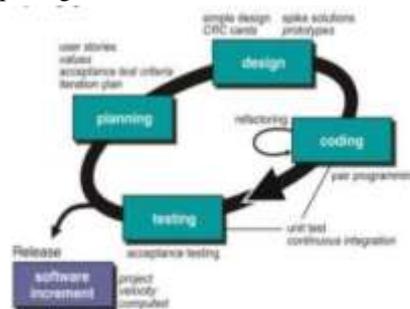
3) **Pengkodean (Coding)**

Setelah stories dikembangkan dan pekerjaan desain awal dilakukan, tim tidak pindah ke kode, melainkan mengembangkan serangkaian pengujian unit yang akan melatih setiap stories yang akan dimasukkan dalam rilis saat ini (*software increment*). Sekali pengujian unit telah dibuat, pengembang lebih mampu fokus pada apa yang harus diterapkan untuk lulus pengujian. Tidak ada hal ekstra yang ditambahkan (KIS). Setelah kode selesai, kode dapat segera diuji, dengan demikian memberikan umpan balik instan kepada pengembang. Konsep kunci selama aktivitas pengkodean (dan salah satu aspek XP yang paling banyak dibicarakan) adalah pemrograman berpasangan (*pair programming*). XP merekomendasikan agar dua orang bekerja bersama di satu komputer untuk membuat kode sebuah stories. Ini memberikan mekanisme untuk pemecahan masalah waktu-nyata (dua kepala seringkali lebih baik dari satu) dan jaminan kualitas waktu nyata (kode ditinjau saat dibuat). Itu juga membuat pengembang fokus pada masalah yang dihadapi. Dalam praktiknya, setiap orang mengambil peran yang sedikit berbeda. Sebagai contoh, satu orang mungkin berpikir tentang detail pengkodean dari bagian tertentu dari desain sementara yang lain memastikan bahwa standar pengkodean (bagian

yang diperlukan dari XP) sedang diikuti atau bahwa kode untuk stories akan memenuhi pengujian unit yang telah dikembangkan untuk memvalidasi kode terhadap stories. Ketika pemrogram berpasangan menyelesaikan pekerjaan mereka, kode yang mereka kembangkan terintegrasi dengan pekerjaan orang lain. Dalam beberapa kasus ini dilakukan setiap hari oleh tim integrasi. Dalam kasus lain, programmer berpasangan memiliki tanggung jawab integrasi. Strategi "integrasi berkesinambungan" ini membantu menghindari masalah kompatibilitas dan antarmuka dan menyediakan lingkungan "*smoke testing*" yang membantu mengungkap kesalahan lebih awal.

4) **Pengujian (Testing)**

Pembuatan pengujian unit sebelum pengkodean adalah elemen kunci dari pendekatan XP. Pengujian unit yang dibuat harus diimplementasikan menggunakan kerangka kerja yang memungkinkan mereka untuk diotomatisasi. Ini mendorong strategi pengujian regresi setiap kali kode diubah. Ketika pengujian unit individual diorganisasikan ke dalam "*universal testing suite*", pengujian integrasi dan validasi sistem dapat terjadi setiap hari. Ini memberikan tim XP indikasi kemajuan yang berkesinambungan dan juga dapat menaikkan bendera peringatan lebih awal jika semuanya serba salah. *Acceptance test*, juga disebut *Customer test*, ditentukan oleh pelanggan dan fokus pada keseluruhan fitur sistem dan fungsionalitas yang dapat dilihat dan ditinjau oleh pelanggan. *Acceptance test* berasal dari *user stories* yang telah diterapkan sebagai bagian dari rilis perangkat lunak.



Gambar 3.1. Tahapan Extreme Programming

IV. HASIL DAN PEMBAHASAN

A. Perencanaan

Pada proses perencanaan dibagi menjadi 3 (tiga) tahapan perencanaan sesuai dengan metode pengembangan XP. Pembagian tahapan berdasarkan kebutuhan fungsional yang fleksible yang dibutuhkan pengguna. Fleksibilitas kebutuhan fungsional yang dapat diimplementasikan pada kantor pusat maupun pada kantor Sudin Dispusip.

Tahapan pertama dengan Aplikasi Release#1 direncanakan pada kantor pusat Dispusip dengan target aplikasi dapat dibuka melalui komputer, notebook dan smartphone dengan *web browser* Google Chrome, Mozilla FireFox dan Opera pada versi terakhir. Berikut ini kegiatan yang direncanakan pada aplikasi release#1.

Perencanaan Release #1				
Kegiatan	Bulan pada tahun 2018			
	9	10	11	12
1. Perencanaan				
2. Desain				
3. Pengkodean				
4. Pengujian				

Tahapan kedua direncanakan mengadakan implementasi aplikasi pada setiap kantor Sudin Dispusip meliputi: Jakarta Pusat, Jakarta Timur, Jakarta Selatan, Jakarta Barat dan Jakarta Utara. Sedangkan untuk kantor di Kepulauan Seribu tidak termasuk pada proyek tahun ini. Berikut ini kegiatan yang direncanakan pada aplikasi release#2.

Perencanaan Release #2				
Kegiatan	Bulan pada tahun 2019			
	1	2	3	4
1. Perencanaan				
2. Desain				
3. Pengkodean				
4. Pengujian				

Tahapan ketiga direncanakan mengadakan integrasi data antara kantor pusat dan kantor Sudin Dispusip. Jika sebelumnya release aplikasi masih menggunakan *database development* sebagai database sementara untuk pelatihan pengguna. Dengan adanya integrasi data pada server database terpusat dan operasional Dispusip maka integritas dan konsolidasi data arsip pada lingkungan kantor pusat dan Sudin Dispusip dapat diwujudkan. Berikut ini kegiatan yang direncanakan pada aplikasi release#3.

Perencanaan Release #3				
Kegiatan	Bulan pada tahun 2019			
	5	6	7	8
1. Perencanaan				
2. Desain				
3. Pengkodean				
4. Pengujian				

B. Desain

Pada proses desain mengimplementasikan metode XP dengan menggunakan kartu CRC sebagai mekanisme yang efektif untuk berpikir tentang perangkat lunak dalam konteks orientasi objek. Kartu CRC dibuat berdasarkan *user story* yang dikumpulkan pada wawancara dengan pengguna. Kartu CRC dibuat sebagai kebutuhan fungsional fungsional yang diimplementasikan pada spesifikasi modul yang akan dibuat.

Desain proses masih sama seperti aplikasi SIMARS-Plus versi Desktop. Begitu juga dengan desain database yang masih dipertahankan sama seperti desain database. Persamaan proses dan database dipertahankan karena nantinya aplikasi SIMARS-Plus versi desktop dan versi web dapat terintegrasi.

Fitur-fitur aplikasi SIMARS-Web dibuat berdasarkan kartu CRC yang telah dibuat beserta spesifikasi kebutuhan fungsional aplikasi. Kartu CRC dapat ditambah dan diubah pada setiap release aplikasi untuk menjamin fleksibilitas kebutuhan aplikasi yang dapat berubah. Di bawah ini adalah hasil akhir kartu CRC yang dibuat saat proses perencanaan aplikasi release#3.

Menu Utama	
-Dapat dibuka melalui alamat URL www.simars-plus.jakarta.go.id	SYSCOM SYSCFG INTERNET
-Memuat 6 (enam) link URL suku dinas (sudin) ditambah dengan Dispusip.	

Login	
-dibuat untuk setiap dinas dan suku dinas dispusip	SYSCOM SYSRGT SYSUSR
-terdapat input user dan password	
-menggunakan kode khusus (acak) yang diinput setelah password	
-password menggunakan enkripsi base64 yang disimpan di database.	

Dashboard	
-terdapat jumlah input arsip yang dihitung per hari, minggu, bulan dan tahun berdasarkan tanggal input arsip. -adanya grafik jumlah arsip yang dihitung per tanggal input. -adanya daftar pengguna yang paling banyak input arsip.	ARSDAT INDEX

Daftar Arsip	
-adanya link untuk menambah, mengedit dan menghapus metadata index arsip. -daftar arsip memuat data nomor urut, nomor dan tanggal arsip, pemilik arsip, jenis, judul dan status. -pengguna dapat melihat / download dokumen arsip dalam bentuk PDF -adanya link untuk mengupload dan reupload gambar arsip dalam bentuk PDF.	ARSDAT MSTPMK MSTMSH MSTJNS

Index Arsip	
-adanya kolom nomor urut, nomor arsip, nomor inventaris, tanggal arsip, tanggal scan, tanggal index, tanggal verifikasi, pemilik, masalah, jumlah lembar arsip dan lokasi penyimpanan arsip. -jumlah lembar arsip dibedakan atas ukuran kertas yaitu: A7, A6, A5, A4, F4, A3, A2, A1 dan A0 serta jumlah keseluruhan lembar. -lokasi simpan terdiri dari gedung, lantai, ruangan, deret, lemari, baris dan kolom.	ARSDAT MSTPMK MSTMSH MSTJNS LOC SYSUSR

Upload Arsip	
-upload arsip jika index metadata arsip sudah lengkap. -upload arsip dalam format PDF. -dapat mengupload maksimal 2GB pada satu file PDF.	ARSDAT IMAGE

Download Arsip	
-dapat melihat data indeks metadata arsip. -mendownload arsip khusus untuk user yang terpilih (admin). -format file dalam PDF yang langsung dapat dibuka melalui browser tanpa	ARSDAT SYSRGT IMAGE

menginstall adobe acrobat reader.	
-----------------------------------	--

Parameter Laporan	
-berdasarkan pemilik, masalah, jenis dan tahun arsip serta tanggal input arsip. -parameter bisa bersifat wajib dan opsional. -laporan didownload dengan format Microsoft Excel (*.xls).	ARSDAT XLS

User	
-menambah, mengedit dan menghapus akun pengguna. -adanya hak admin, inputer, verifikator. -adanya password default untuk pertamakali login dan dapat mengubah password saat login pertama kali. -dapat mereset password jika pengguna lupa passwordnya.	SYSUSR SYSRGT

Lokasi Simpan	
-menambah, mengedit dan menghapus master lokasi simpan arsip. -lokasi simpan secara hirarki terdiri dari gedung, lantai, ruangan, deret, lemari, baris dan kolom.	LOGGDG LOCLNT LOCRNG LOCLRG LOCLMR LOCKLM LOCBRS

Pemilik	
-menambah, mengedit dan menghapus master pemilik arsip. -adanya input kode, nama dan alamat pemilik arsip.	MSTPMK

C. Pengkodean

Tahapan pengkodean adalah proses implementasi kartu CRC menjadi koding dalam bahasa pemrograman. Proses ataran kartu CRC, pengkodean dan pengujian dilakukan secara simultan untuk menjaga fleksibilitas perubahan kebutuhan fungsional pengguna. Pengkodean mengikuti pendekatan XP, dimana dalam mengembangkan kode sumber menggunakan metode *Pair Programming* dengan 2 (dua) orang programmer secara bersama-sama mengembangkan kode sumber.

PHP versi 5.6 menjadi bahasa pemrograman dan SPA sebagai *framework* pemrogramannya. Pemilihan bahasa PHP sebagai *server-side scripting* berdasarkan permintaan bagian teknologi informasi Dispusip.

Pemilihan SPA sebagai *framework* pemrograman didasarkan bahwa kuantitas metadata dan gambar arsip yang besar membutuhkan *framework* yang ringan dan dapat dibuka dengan cepat pada sisi client. Berikut ini hasil tampilan terakhir dari tahapan pengkodean pada aplikasi release#3.



Gambar 4.1. Tampilan Menu Utama



Gambar 4.2. Tampilan Login



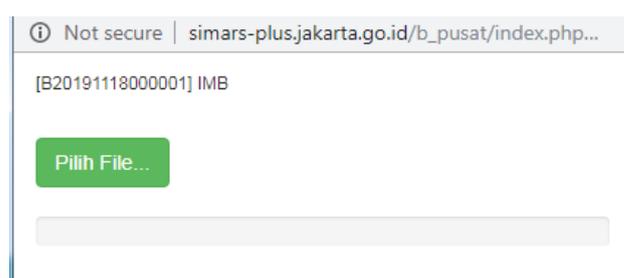
Gambar 4.3. Tampilan Dashboard



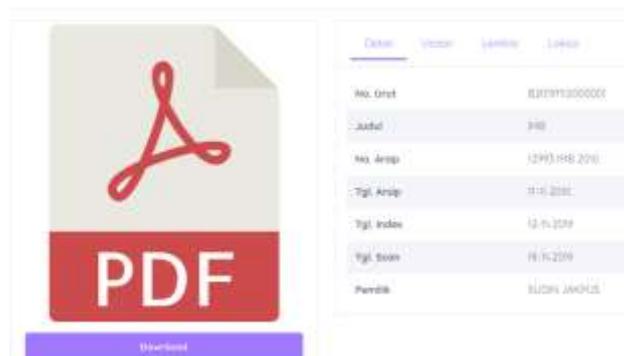
Gambar 4.4. Tampilan Daftar Arsip



Gambar 4.5. Tampilan Index Arsip



Gambar 4.6. Tampilan Upload Arsip



Gambar 4.7. Tampilan Download Arsip



Gambar 4.8. Tampilan Parameter Laporan



Gambar 4.9. Tampilan Master User



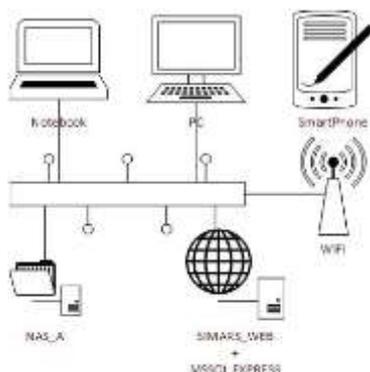
Gambar 4.10. Tampilan Master Lokasi Simpan



Gambar 4.11. Tampilan Master Pemilik Arsip

D. Pengujian

Pada proses pengujian dibagi menjadi 3 (tiga) tahapan pengujian sesuai metode pengembangan XP. Hasil pengujian berdasarkan *User Acceptance Test* (UAT) yang dilakukan pengguna terhadap masing-masing release aplikasi. Pengujian pertama mencakup hasil Release#1 yang dilaksanakan dikantor pusat Dispusip. Pengujian dilakukan dengan mengkonfigurasi jaringan komputer di bawah ini.



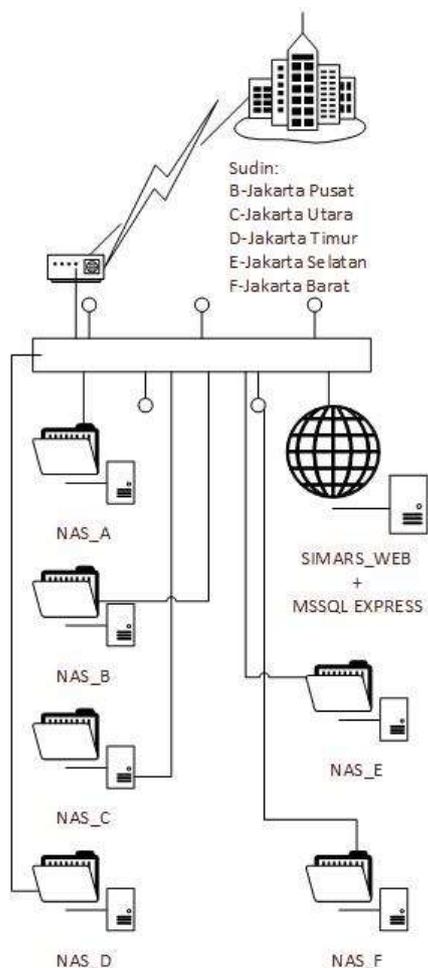
Gambar 4.12. Topologi Jaringan Release #1

Seperti digambarkan di atas bahwa aplikasi sudah terinstall pada server *simars_web* dimana menggunakan Apache web server dan sistem database Microsoft SQL Server versi Express Edition. *Network Attached Storage* (NAS) digunakan untuk menyimpan image arsip dengan struktur direktori yang sudah ditentukan. Pengujian dilakukan pada setiap jenis client seperti komputer, notebook dan smartphone. Browser yang digunakan adalah Google Chrome, Mozilla Firefox dan Opera dengan versi yang paling terakhir. Di bawah ini adalah hasil pengujian pada aplikasi release#1.

Pengujian Release #1			
No	Halaman	Hasil	Keterangan
1	Menu Utama	Gagal	Sudin belum bisa
2	Login	Berhasil	
3	Dashboard	Berhasil	
4	Daftar Arsip	Berhasil	
5	Index Arsip	Berhasil	
6	Upload Arsip	Gagal	File 2GB Error
7	Download Arsip	Berhasil	
8	Parameter Laporan	Gagal	Laporan konsolidasi
9	User	Berhasil	
10	Lokasi Simpan	Berhasil	
11	Pemilik	Berhasil	

Pengujian kedua mencakup implementasi aplikasi pada setiap Sudin Dispusip yaitu: Jakarta Pusat, Jakarta Timur, Jakarta Selatan, Jakarta Barat dan Jakarta Utara. Sedangkan untuk Sudin Kepulauan Seribu tidak diimplementasikan pada tahun anggaran ini. Setiap Sudin memiliki NAS masing-masing yang digunakan untuk menyimpan data image arsip. Jaringan internet menggunakan jaringan Diskominfo dengan cadangan koneksi LinkNet atau Indihome untuk setiap Sudin Dispusip.

Sistem database masih menggunakan server database development dengan struktur database yang sama dengan sever operasional kantor pusat Dispusip provinsi DKI Jakarta. Hal tersebut demi keamanan database operasional karena setting keamanan firewall untuk link www.simars-plus.jakarta.go.id dikonfigurasi oleh Diskominfo provinsi DKI Jakarta. Pengujian release#2 dilakukan dengan mengkonfigurasi jaringan komputer di bawah ini.



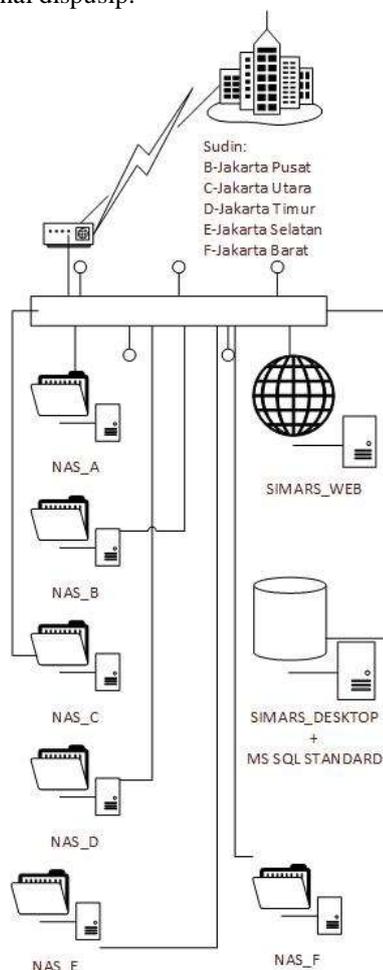
Gambar 4.13. Topologi Jaringan Release #2

Seperti terlihat pada topologi jaringan di atas bahwa sistem database masih menggunakan Microsoft SQL Server versi Express Edition yang terinstall pada server-web. Hal tersebut dilakukan karena faktor keamanan sistem database simars-desktop sebagai database arsip yang terpusat dan operasional Dispupis. Selain itu faktor integritas data pada server-web sebagai sistem database development masih pada tahap pelatihan input data. Di bawah ini adalah hasil pengujian pada aplikasi release#2.

Pengujian Release #2			
No	Halaman	Hasil	Keterangan
1	Menu Utama	Berhasil	Sudin OK
2	Login	Berhasil	
3	Dashboard	Berhasil	
4	Daftar Arsip	Berhasil	
5	Index Arsip	Berhasil	
6	Upload	Gagal	File 2GB Error

	Arsip		
7	Download Arsip	Berhasil	
8	Parameter Laporan	Gagal	Laporan konsolidasi
9	User	Berhasil	
10	Lokasi Simpan	Berhasil	
11	Pemilik	Berhasil	

Pengujian ketiga mencakup penggunaan database terpusat dan operasional Dispupis, artinya database pada simars-web tidak digunakan lagi. Server simars-web hanya terinstall Apache Webserver dengan *mapping drive* pada setiap Sudin Dispupis sesuai dengan ketentuan. Data-data arsip yang diinput di setiap Sudin dilakukan migrasi data ke sistem database simars-desktop sebagai database terpusat dan operasional dispupis.



Gambar 4.14. Topologi Jaringan Release #3

Seperti terlihat pada hasil pengujian sebelumnya, bahwa adanya masalah upload image arsip yang melebihi 2 GB, maka metode upload image dirubah dari menggunakan metode standar multi-part upload menjadi metode *chunk* yang membagi file menjadi maksimal 200MB per *chunk* sehingga file dapat diupload simultan per bagian file yang diupload. Dengan menggunakan sistem database terpusat maka konsolidasi data arsip antara kantor pusat Dispusip dan 5 (lima) kantor Sudin dapat terlaksana. Di bawah ini adalah hasil pengujian pada aplikasi release#3.

Pengujian Release #3			
No	Halaman	Hasil	Keterangan
1	Menu Utama	Berhasil	
2	Login	Berhasil	
3	Dashboard	Berhasil	
4	Daftar Arsip	Berhasil	
5	Index Arsip	Berhasil	
6	Upload Arsip	Berhasil	File 2GB OK
7	Download Arsip	Berhasil	
8	Parameter Laporan	Berhasil	Konsolidasi OK
9	User	Berhasil	
10	Lokasi Simpan	Berhasil	
11	Pemilik	Berhasil	

V. KESIMPULAN

A. Simpulan

Berdasarkan hasil penelitian, desain serta pengujian yang dilakukan dapat disimpulkan bahwa:

- 1) Pendekatan XP dapat mempercepat proses pengembangan aplikasi, khususnya pada proyek yang tidak dapat memprediksi semua kebutuhan aplikasi pada tahapan perencanaan.
- 2) Pengujian release aplikasi dilakukan secara simultan memberikan gambaran yang utuh kepada pengguna tentang apasaja kebutuhan fungsional dan kebutuhan nonfungsional dari pengembangan aplikasi dengan pendekatan XP.
- 3) Fleksibilitas pengembangan fitur-fitur aplikasi dapat dilakukan bertahap seiring tersedianya kebutuhan infrastruktur yang ditetapkan oleh spesifikasi pengembangan aplikasi.

B. Saran

Berdasarkan hasil penelitian, desain serta pengujian yang dilakukan dapat diberikan saran untuk pengembangan fitur selanjutnya yaitu:

- 1) Dokumentasi tiap tahapan pengembangan aplikasi dalam pendekatan XP harus secara simultan dilakukan pada setiap release aplikasi untuk mencegah terjadinya kekurangan dalam materi dokumentasi pengembangan aplikasi.
- 2) Dibutuhkan tim pengembang yang solid dan berkomitmen kepada penyelesaian proyek mengingat pada proyek pemerintah daerah memerlukan penyelesaian pada akhir tahun dan dilanjutkan pada masa garansi proyek.
- 3) Pengguna aplikasi harus diinformasikan dahulu tentang perencanaan pengembangan aplikasi setiap releasenya untuk mencegah kesalahpahaman pengguna bahwa aplikasi sudah sempurna saat release pertama.

REFERENSI

- [1] Undang-Undang Nomor 43 Tahun 2009 tentang Kearsipan, Arsip Nasional Republik Indonesia (ANRI)
- [2] Pengantar Kearsipan, (2002), Arsip Nasional Republik Indonesia (ANRI)
- [3] Basuki, S., 2003. Manajemen Arsip Dinamis, GRAMEDIA, Jakarta
- [4] Modul Penataan Arsip Statis, (2007), Arsip Nasional Republik Indonesia (ANRI)
- [5] Pressman, Roger S. (2010). Software Engineering. Singapore : The McGraw-Hill Companies
- [6] Glenford, J. M., Sandler, C., & Badgett, T. (2012). The Art of Software Testing. New Jersey: John Wiley & Sons, Inc.
- [7] Remick, Jarel. (2011). What Is a Web App? Here's Our Definition, <http://web.appstorm.net>, diakses pada tanggal 2 Desember 2019
- [8] Skólski, Paweł (2017), Single-page application vs. multiple-page application, <https://neoteric.eu/blog/single-page-application-vs-multiple-page-application>, diakses pada tanggal 2 Desember 2019
- [9] Sugiyono. (2014). Metodologi Penelitian Kualitatif, Kuantitatif dan R&D. Bandung:Alfabeta.