

# Perancangan Aplikasi *E-logbook*

Studi Kasus: Fakultas Kedokteran Universitas Indonesia

Andy Wijaya<sup>1</sup>, Johannes Fernandes Andry<sup>2</sup>

<sup>1,2</sup> Program Studi Sistem Informasi, Fakultas Teknologi, Universitas Bunda Mulia, Jakarta Utara, Indonesia  
andywijaya22.aw@gmail.com  
jandry@bundamulia.ac.id

Diterima 29 April 2020

Disetujui 15 Juni 2020

**Abstract**— Berdasarkan penelitian yang telah dilakukan pada Peserta Program Pendidikan Dokter Spesialis (PPDS) FKUI diperoleh masalahnya bahwa sistem yang diterapkan saat ini belum menggunakan teknologi komputerisasi dalam proses kegiatannya sehari-hari. Peserta PPDS adalah calon-calon Dokter Spesialis yang akan menjadi *expert* dan konsultan dalam spesialisasinya masing-masing. PPDS dihadapkan dengan bermacam-macam kasus, melalui pasien gawat darurat, pasien rawat inap, maupun rawat jalan sehingga memiliki kemampuan dalam merawat pasien. Dengan penggunaan aplikasi *E-logbook* ini diharapkan dapat mempermudah dan mengolah proses pencatatan yang ada agar lebih cepat dan mudah dalam mengumpulkan dan mengelola data menjadi informasi yang dapat dimanfaatkan oleh pemangku kepentingan untuk meningkatkan pelayanan. Metode yang digunakan adalah menggunakan SDLC. Hasil dari penelitian yang telah dilakukan pada FKUI, diketahui bahwa sistem pencatatan menggunakan media kertas merupakan alat yang digunakan untuk mencatat setiap data pada rekam medis tersebut mengakibatkan seringnya terjadi kesalahan yang berupa hilang dan rusaknya rekam medis. Oleh karena itu dibangunnya sebuah aplikasi *E-logbook* agar dapat mengurangi *human error* pada proses pembuatan laporan maupun proses pencatatan dan menjaga datanya dari hilangnya dan rusaknya data. Aplikasi *E-logbook* ini dapat terus dikembangkan pada modul lain sesuai kebutuhan *user*.

**Index Terms**—E-logbook, FKUI, PPDS, SDLC

## I. PENDAHULUAN

Fakultas Kedokteran Universitas Indonesia (FKUI) adalah fakultas kedokteran tertua dan salah satu fakultas kedokteran terbaik di Indonesia. FKUI berkomitmen dengan kuat untuk menjamin mutu serta meningkatkan kualitas pendidikan kedokteran dengan cara membangun pendidikan dengan kurikulum internasional yang bersifat kompetensi untuk meningkatkan derajat kesehatan masyarakat [1]. Jenjang Pendidikan dokter spesialis di FKUI yang biasa disebut dengan PPDS adalah calon-calon Dokter Spesialis yang akan menjadi *expert* dan konsultan dalam spesialisasinya masing-masing [2]. PPDS dihadapkan dengan berbagai kasus salah satunya, melalui berbagai kasus seperti rawat jalan, pasien gawat darurat, maupun rawat inap sehingga memiliki

kemampuan dalam menangani pasien. Mendalami ilmu sebagai dasar untuk melatih keterampilan *medic* secara bertahap dan terjadwal melalui berbagai kegiatan seperti kuliah, presentasi kasus, referat, pembacaan jurnal, dan laporan jaga [3].

Salah satu laporan yang dibuat merupakan *logbook* atau biasa disebut rekam medis adalah salah satu pelayanan penunjang layanan kesehatan di rumah sakit yang menjadi dasar penilaian mutu pelayanan *medic* [4]. Dengan adanya pelayanan rekam medis (*logbook*) sebagai sarana yang menyediakan informasi untuk memudahkan pelayanan kepada pasien dan memudahkan untuk pengambilan keputusan berupa perencanaan, pengorganisasian, pelaksanaan, pengawasan, penilaian dan pengendalian oleh pemberi pelayanan klinis dan administrasi pada sarana pelayanan kesehatan [5].

Ada 3 pihak yang bersangkutan dalam proses pencatatan rekam medis (*logbook*), yaitu dokter PPDS (mahasiswa) yang akan mengirim *logbook* kepada dokter supervisor (dokter) kemudian akan diperiksa hasil *logbook* yang dibuat oleh dokter PPDS. Lalu dokter supervisor akan mengirim kembali *logbook* yang berisikan keterangan berupa sesuai atau tidak sesuai diagnosis yang di buat oleh dokter PPDS. Dan SPS yang bertugas memantau kegiatan dan perkembangan PPDS.

Kegiatan rekam medis meliputi, *logbook* patologi organ, *logbook* bimbingan Akademik, *logbook* bimbingan tulisan ilmiah, *logbook* potong beku dan *logbook* grossing. Proses pencatatan *logbook* saat ini masih menggunakan kertas dan buku untuk mencatat. Menurut Clifford dkk dalam penelitiannya menyimpulkan bahwa sistem manual berbasis kertas dapat menyediakan solusi sementara, tetapi pertukaran informasi sangatlah lambat dan rawan kesalahan. Selain itu, seiring bertambahnya jumlah pasien hingga ratusan bahkan ribuan akan menyulitkan dalam penyimpanan dan pengolahan [6]. Lambatnya proses *logbook* konvensional memberikan beberapa efek buruk yang dapat disebabkan olehnya, hilang atau rusaknya *logbook* dan lamanya proses menghitung hasil yang di peroleh dari data *logbook* adalah salah satu dampak buruk dari proses *logbook* konvensional.

Sistem *logbook* konvensional tidak memiliki system yang dapat memonitor progress dari PPDS. Seiring kemajuan teknologi yang berkembang sangat pesat membuat banyak organisasi dan lembaga pendidikan menggunakan teknologi berbasis komputer dan jaringan untuk membantu pekerjaannya khususnya dalam pengelolaan dan penyampaian sistem informasi karena bersifat efektif dan efisien [7]. Beberapa hal yang harus menjadi perhatian, adalah bahwa dalam mengelola sebuah institusi pelayanan kesehatan, memerlukan adanya manajemen dan penguasaan teknologi yang baik, serta rencana yang matang [8].

Menciptakan sistem *logbook* berbasis *website* memungkinkan para PPDS untuk meningkatkan keamanan *logbook* agar tidak hilang atau rusak dan mempermudah PPDS mengetahui hasil dari data *logbook* tersebut, serta SPS dan Dokter Supervisor dapat dengan mudah melihat perkembangan dari PPDS karena hasil kerjanya akan secara otomatis di perbaharui ketika adanya *logbook* yang di *input*.

Oleh karena itu, berdasarkan pada hambatan yang dialami PPDS [9], prancangan aplikasi *E-Logbook* dirancang dan dibangun dalam berbasis *website* menggunakan metode SDLC *Waterfall*. yang diharapkan dapat memenuhi kebutuhan PPDS dalam pencatatan *logbook* yang terkomputerisasi untuk meningkatkan kinerja, kualitas pelayanan, daya saing dan kualitas PPDS yang dihasilkannya.

## II. PENELITIAN TERDAHULU

### A. *System Development Life Cycle (SDLC)*

*SDLC* merupakan serangkaian proses yang di gunakan untuk memahami bagaimana suatu sistem informasi dapat mendukung kebutuhan bisnis dengan merancang suatu sistem, membangunnya, dan mengirimkannya kepada pengguna. Orang yang berperan dalam SDLC adalah analis sistem, yang menganalisis situasi bisnis, mengidentifikasi peluang untuk perbaikan, dan merancang sistem informasi untuk mengimplementasikannya. Ada dua poin penting untuk dipahami tentang SDLC. Pertama, Anda harus mendapatkan pengertian umum tentang fase dan langkah-langkah di mana proyek SI bergerak dan beberapa teknik yang menghasilkan hasil tertentu. Kedua, penting untuk dipahami bahwa SDLC adalah proses penyempurnaan bertahap suatu proyek SI [10].

*SDLC Waterfall* merupakan serangkaian proses yang melibatkan beberapa tahap pengembangan yang meliputi: *Planing, Analysis, Design, dan Implementasion*. Pada prinsipnya, satu tahap harus diselesaikan terlebih dahulu sebelum lanjut ketahap berikutnya [11].

### B. *E-logbook*

Menurut Kamus Besar Bahasa Indonesia (KBBI) arti kata *logbook* kamus Inggris Indonesia maksud kata definisi pengertian makna dan arti kata *logbook*

merupakan buku pencatatan kejadian [12]. *Log Book* adalah sebuah buku catatan atau dokumen yang dibutuhkan siswa untuk mendokumentasikan secara detail semua aktivitas dalam proses pembelajaran yang berisikan identitas diri, informasi kompetensi, catatan harian, catatan kegiatan pembelajaran, refleksi siswa, lembar penilaian, lembar penelitian, kuesioner, lembar pengawasan guru maupun pembimbing industri jika berupa praktik kerja di bidang industri.

Dalam dunia medis *logbook* dapat di artikan sebagai catatan rekam medis. Rekam medis diartikan sebagai sebuah keterangan yang tertulis maupun yang terdokumentasi tentang identitas, anamneses, diagnosa segala pelayanan, penentuan fisik laboratorium, tindakan medis yang diberikan kepada pasien dan pengobatan baik yang dirawat inap, rawat jalan, maupun yang mendapatkan pelayanan gawat darurat [13].

### C. *Unified Modeling Language (UML)*

UML merupakan metode yang banyak digunakan untuk menggambarkan dan menceritakan desain sistem perangkat lunak. UML menggunakan konsep desain berorientasi objek, tetapi tidak tergantung pada bahasa pemrograman tertentu dan dapat digunakan untuk menjelaskan proses dan persyaratan bisnis secara umum [14].

*Unified Modeling Language (UML)* terbagi menjadi beberapa rangkaian diagram yang saling bersinambungan dan memiliki peran masing-masing dalam menjelaskan fungsi dari sistem yang di rancang. Untuk menjelaskan secara mendetil mengenai proses bisnis dan alur suatu sistem yang sedang dikembangkan atau sistem yang akan dikembangkan.

#### C.1 *Use Case Diagram*

*Use Case Diagram* biasanya merupakan langkah awal dalam memahami dan menganalisa kebutuhan sistem pada saat perancangan system di usulkan. *Use case diagram* merupakan diagram untuk kebiasaan (behavior) SI yang akan dibuat. *Use case* biasanya digunakan sebagai gambaran fungsi-fungsi yang terdapat pada SI dan mengetahui siapa saja yang mengoperasikan fungsi-fungsi tersebut [15]. Dengan kata lain *Use Case Diagram* biasanya digunakan untuk mengelola dan menjelaskan proses-proses yang dijalankan oleh sebuah sistem dan menggambarkan fungsi-fungsi yang diharapkan dari sebuah sistem. *Use case diagram* berisikan simbol-simbol berupa *actor, relationship* diantaranya dapat dilihat pada Tabel 1, Simbol *Use Case Diagram*.

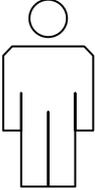
*Extend* adalah situasi dimana *use case* akan dijalankan ketika suatu *use case* memenuhi syarat tertentu.

#### C.2 *Activity Diagram*

*Diagram activity* menunjukkan aktivitas sistem dalam bentuk kumpulan aksi-aksi, bagaimana masing-

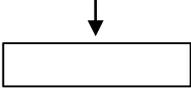
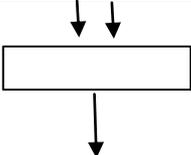
masing aksi tersebut dimulai, keputusan yang mungkin terjadi hingga berakhirnya aksi. *Activity diagram* juga dapat menggambarkan proses lebih dari satu aksi dalam waktu bersamaan [16]. Berikut simbol-simbol yang di gunakan dalam *activity diagram*, lihat Tabel 2.

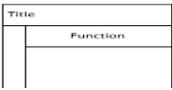
Tabel 1. Simbol *Use Case Diagram* [15].

Simbol	Keterangan
 Actor	Actor adalah sebuah gambaran dari orang yang akan menjalankan sistem dan mengaktifkan fungsi-fungsi dari sistem yang akan di jalankan. Untuk mengidentifikasi aktor, adanya ketentuan untuk pembagian tenaga kerja dan proses-proses yang dijalankan berkaitan dengan proses yang akan dijalankan oleh sistem. Baik Orang ataupun sistem dapat memiliki beberapa peran.
	Use Case menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktor. penulisan dalam menggunakan use case adalah dengan menggunakan kata kerja untuk menggambarkan fungsi dari use case tersebut
	Asosiasi sebagai penghubung aktor dan use case, yang berupa garis tanpa panah berperan sebagai hubungan antara aktor dengan system yang berinteraksi secara langsung.
	Asosiasi dengan panah terbuka merupakan penghubung yang menggabungkan aktor dengan use case untuk mengindikasikan aktor melakukan interaksi satu arah terhadap sistem.
	<i>Include</i> , merupakan fitur-fitur atau fungsi-fungsi yang berjalan ketika aktor menjalankan use case. Sehingga ketika aktor menjalankan use case tersebut akan menjalankan use case yang di tunjuk dengan tanda include secara otomatis

D. *Entity Relationship Diagram (ERD)*

Tabel 2. Simbol *Activity Diagram* [15].

Simbol	Keterangan
	<i>Start Point</i> , merupakan titik awal suatu system berjalan atau mulai.
	<i>End Point</i> , berfungsi sebagai tanda telah berakhirnya aktivitas yang terjadi dalam system.
	<i>Activities</i> , menggambarkan suatu proses/kegiatan bisnis yang sedang di jalankan.
	<i>Fork</i> atau percabangan, menunjukkan kegiatan yang dilakukan secara bersamaan untuk menggabungkan dua kegiatan paralel menjadi satu kegiatan
	<i>Join</i> atau rake, menunjukan adanya 2 kegiatan atau lebih yang akan bergabung menjadi satu kegiatan

	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, true atau false
	<i>Swimlane</i> , pembagian activity diagram untuk menunjukkan siapa melakukan apa

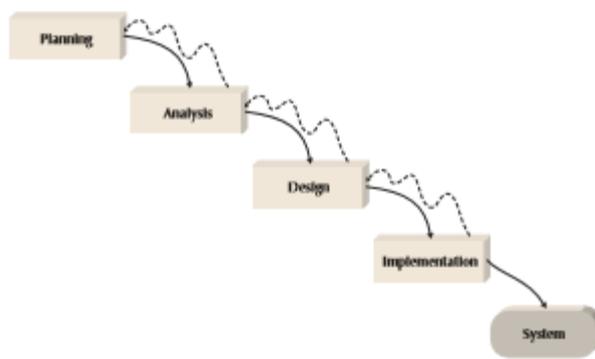
Merupakan sekumpulan basis data yang digunakan untuk menggambarkan model *Entity Relationship* yang berisi komponen-komponen. Himpunan Entitas dan Himpunan Relasi yang masing-masing dilengkapi dengan atribut-atribut yang mempersentasikan seluruh fakta dari dunia nyata yang kita tinjau [17].

E. *Wireframe*

*Wireframe* adalah kerangka dasar atau blueprint dari suatu halaman aplikasi yang akan dibangun oleh pengembang aplikasi. Secara garis besar di dalam wireframe ini kita menempatkan elemen-elemen penting dari halaman aplikasi tersebut pada posisinya masing-masing seperti *banner*, *body content*, *menu link*, kolom, *footer* maupun fitur-fitur lainnya yang ada dalam aplikasi nantinya. Secara visual tampilan dari wireframe ini hanya terdiri dari kotak dan garis yang menandakan posisi dari masing-masing elemen dari layout halaman aplikasi [18].

III. METODOLOGI

Metode yang digunakan untuk pengembangan sistem adalah dengan metode SDLC, merupakan metodologi pengembangan sistem, peran dan keterampilan yang diperlukan untuk analisis sistem, karakteristik dasar sistem berorientasi objek. Dalam mengerjakan proyek, fase dan langkah SDLC berlangsung dalam jalur logis dari awal hingga akhir. Dalam proyek lain, tim proyek bergerak melalui langkah-langkah berturut-turut, secara bertahap, berulang, atau dalam pola lain. SDLC memiliki 4 fase dasar yang meliputi, *Planing*, *Analysis*, *Design*, *Implementation* yang umum untuk semua proyek pengembangan sistem informasi. Pada Gambar 1, SDLC model menurut David Tegarden, dapat di lihat fase-fase yang di lalui oleh tahap SDLC [10].



Gambar 1. SDLC [10]

### A. *Planning*

Fase perancangan merupakan proses dasar untuk memahami mengapa sistem informasi harus dibangun. Biasanya pada tahap ini dilakukan wawancara untuk memahami sistem apa yang akan dibangun. *Planning* memiliki 2 langkah yang meliputi:

1. Selama proses inisiasi proyek, permintaan sistem menyajikan ringkasan singkat dari kebutuhan bisnis, dan itu menjelaskan bagaimana sistem yang mendukung kebutuhan akan menciptakan nilai bisnis. Departemen ini bekerja bersama dengan orang atau departemen yang menghasilkan permintaan (disebut sponsor proyek) untuk melakukan analisis kelayakan.
2. Setelah proyek disetujui, ia memasuki manajemen proyek. Selama manajemen proyek, manajer proyek membuat rencana kerja, menjadi staf proyek, dan menerapkan teknik untuk membantu tim proyek mengendalikan dan mengarahkan proyek melalui seluruh SDLC. Penyampaian untuk manajemen proyek adalah rencana proyek, yang menggambarkan bagaimana tim proyek akan mengembangkan sistem.

### B. *Analysis*

Setelah melewati fase *planning*, ditentukannya atau dimulai Analisa untuk menjawab pertanyaan tentang siapa yang akan menggunakan sistem, apa yang akan dilakukan sistem, dan di mana serta kapan akan digunakan. Selama fase ini, tim proyek menyelidiki setiap sistem saat ini, mengidentifikasi peluang untuk perbaikan, dan mengembangkan konsep untuk sistem baru. Pada fase ini memiliki 3 tahap yaitu:

1. Strategi analisis dikembangkan untuk memandu upaya tim proyek. Strategi seperti itu biasanya mencakup analisis sistem saat ini (disebut sistem apa adanya) dan masalahnya dan kemudian cara merancang sistem baru (disebut sistem yang akan datang).

2. Langkah selanjutnya adalah mengumpulkan persyaratan (biasanya melalui wawancara atau tanya jawab). Analisis informasi ini mengarah pada pengembangan konsep untuk sistem baru. Konsep sistem ini kemudian digunakan sebagai dasar untuk mengembangkan seperangkat bisnis model analisis, yang menggambarkan bagaimana bisnis akan beroperasi jika sistem baru dikembangkan.
3. Kemudian Analisis, konsep sistem, dan model digabungkan menjadi sebuah dokumen yang disebut proposal sistem, yang disajikan kepada sponsor proyek dan pembuat keputusan utama lainnya (misalnya, anggota komite persetujuan) yang memutuskan apakah proyek harus terus bergerak maju.

### C. *Design*

Setelah selesai tahap analisis mulailah memutuskan bagaimana sistem akan beroperasi, dalam hal perangkat keras, perangkat lunak, dan infrastruktur jaringan; antarmuka pengguna, formulir, dan laporan; dan program, database, dan file tertentu yang akan dibutuhkan. Pada fase desain terbagi menjadi 4 tahap yaitu:

1. Strategi desain pertama kali dikembangkan. Ini mengklarifikasi apakah sistem akan dikembangkan oleh pemrogram perusahaan sendiri, apakah sistem akan diserahkan ke perusahaan lain (biasanya perusahaan konsultan), atau apakah perusahaan akan membeli paket perangkat lunak yang ada.
2. Pada tahap ini mengarah pada pengembangan desain arsitektur dasar untuk sistem, yang menggambarkan perangkat keras, perangkat lunak, dan infrastruktur jaringan yang akan digunakan. Dalam kebanyakan kasus, sistem akan menambah atau mengubah infrastruktur yang sudah ada di organisasi. Desain antarmuka menentukan cara pengguna akan bergerak melalui sistem dan formulir serta laporan yang akan digunakan sistem.
3. Pada tahap ini database dan spesifikasi file dikembangkan. Lalu mendefinisikan dengan tepat data apa yang akan disimpan dan di mana mereka akan disimpan.
4. Kemudian tim analisis mengembangkan desain program, yang mendefinisikan program yang perlu ditulis dan apa yang akan dilakukan oleh masing-masing program.

#### D. Implementation

Merupakan tahap di mana sistem sebenarnya dibangun mengikuti rancangan design yang telah di buat. Ini adalah fase yang biasanya mendapat perhatian paling besar, karena bagi kebanyakan sistem itu adalah bagian tunggal yang paling lama dan paling mahal dari proses pengembangan. Pada fase ini memiliki 3 tahap, yaitu:

1. Langkah pertama dalam implementasi adalah konstruksi sistem. Sistem ini dibangun dan diuji untuk memastikan kinerja seperti yang dirancang. Pengujian adalah salah satu langkah paling penting dalam fase implementasi, karena biaya untuk memperbaiki bug bias sangat besar. Sebagian besar organisasi memberikan banuak waktu dan perhatian untuk menguji daripada menulis program.
2. Pada tahap ini Sistem di-*install*. Instalasi merupakan proses dimana sistem lama dimatikan dan yang baru dijalankan. Salah satu aspek kenversi yang paling penting adalah pengembangan rencana pelatihan untuk mengajari pengguna cara menggunakan sistem baru dan membantu mengelola perubahan yang disebabkan oleh sistem baru.
3. Lalu pada tahap ini Tim analis membuat rencana dukungan untuk sistem. Rencana ini biasanya mencakup tinjauan formal atau informal paska-implementasi serta cara sistematis untuk mengidentifikasi perubahan besar dan kecil yang diperlukan untuk sistem.

#### E. System

merupakan tahap di mana sistem sudah siap dipakai dan digunakan.

### IV. HASIL DAN PEMBAHASAN

Berikut merupakan tahapan system informasi *E-logbook* yang akan di bangun menggunakan SDLC *Waterfall*:

#### A. Planning

Pada tahap ini, terdapat permintaan dari PPDS FKUI untuk memnbuat sebuah sistem yang dapat membantu proses pencatatan *logbook*. Oleh karena itu dilakukannya studi kasus yang berupa wawancara kepada salah satu PPDS dan SPS untuk mengidentifikasi kebutuhan sistem yang akan dibangun. Setelah itu dibuatnya rencana kerja yang menggambarkan prose kerja yang akan dipakai. Dari hasil wawancara tersebut dapat di simpulkan bahwa sistem yang di butuhkan merupakan *E-logbook*.

#### B. Analysis

Setelah tahap *planning*, diketahui bahwa sistem *logbook* yang sekarang di pakai masih menggunakan kertas dan buku. Sehingga sistem masih melalui proses yang lama untuk mengetahui progress dari setiap PPDS. Untuk proses biasanya PPDS mendianogsis pasien yang kemudian akan di tuliskan ke dalam kertas, setelah itu PPDS akan menaru *Logbook* tersebut di ruang supervisor untuk di validasi hasil dari Analisa PPDS terhadap pasien. Setelah di periksa, supervisor akan memberikan penilaian apakah *logbook* tersebut sudah sesuai atau tidak sesuai dan memberikan tanda tangan. Dari sistem tersebut terdapat kendala yaitu PPDS harus meluangkan waktu untuk memberikan *logbook* tersebut kepada supervisor dan supervisor harus meluangkan waktu untuk datang untuk memvalidasi *logbook* tersebut sehingga membutuhkan waktu dan tenaga. Dan di sisi lain terdapat kendala ketika PPDS harus melaporkan progressnya ke SPS untuk mengetahui apakah memenuhi standar kelulusan, PPDS harus merekap *logbook* dan menghitung secara manual hasil *logbook* yang konvensional sehingga sistem yang sudah berjalan dianggap kurang efektif dan efisien.

Oleh karena itu pada tahap ini, dilakukannya analisa kebutuhan sistem dari *planning* yang mencakup beberapa *logbook* yang akan di rancang yang meliputi *logbook* dianogsis pasien, *logbook* bimbingan Akademik, *logbook* bimbingan tulisan ilmiah, *logbook* potong beku dan *logbook* *grossing*. Yang dimana para PPDS dapat mengimput laporan yang dapat di lihat dan dapat memberikan keterangan sesuai atau tidak sesuai oleh supervisor yang bersangkutan. Lalu SPS dapat melihat perkembangan PPDS.

Semua itu di rancang kedalam aplikasi berbasis web yang menggunakan *frame work* Laravel. Dengan menggunakan sistem penyimpanan basis data berupa My SQL. Sehingga memungkinkan supervisor mengecek hasil laporan yang di buat PPDS dimana saja yang terhubung dengan internet. Sehingga PPDS tidak harus menunggu kedatangan dokter supervisor untuk pengecekan *logbooknya*.

#### C. Design

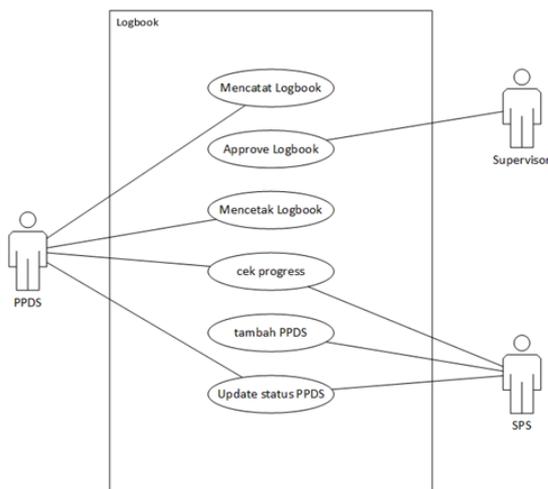
Pada tahap ini dibuatnya desain berdasarkan hasil Analisa. Dapat di simpulkan bahwa desain aplikasi memiliki 3 langkah yang meliputi:

##### C.1 Unified Modeling Language (UML)

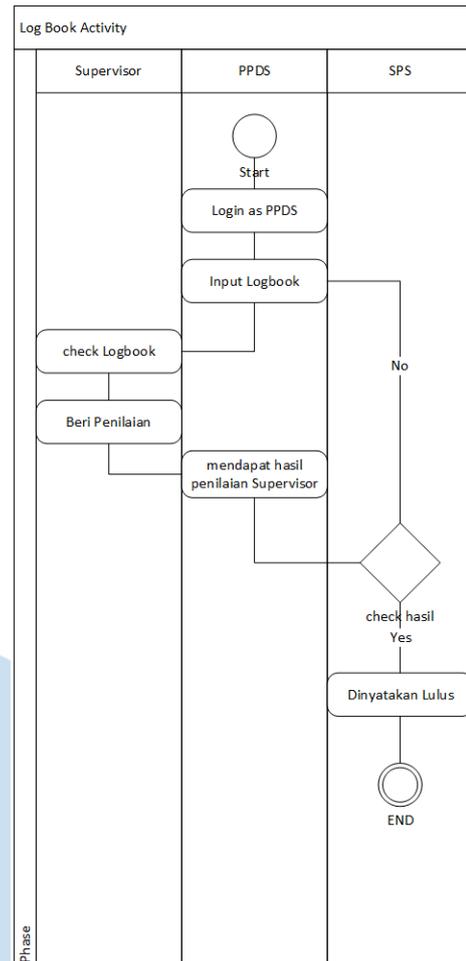
Pada tahap ini mulai merancang arsitektur yang terkait dengan menggunakan UML, lalu membuat diagram structural dan perilaku yang menggambarkan deskripsi kelas domain masalah dan interaksinya. Berikut merupakan *use case diagram* dan *activity diagram*.

C.2 Use Case Diagram

Pada Gambar 2, *Uses Case Diagram* ini menggambarkan beberapa fungsi dari setiap aktor pada aplikasi. Ada terdapat 3 aktor yang terlibat pada aplikasi *E-logbook*, yaitu PPDS, Supervisor, dan SPS. Dimana PPDS dapat mengimput kan data kedalam *logbook*, melihat progress, mencetak *logbook*, dan meng-*update* status, supervisor dapat *Approve logbook* yang telah di tulis dan SPS dapat mencetak progress, tambah PPDS dan dapat *Update* status PPDS.



Gambar 2. Use Case Diagram

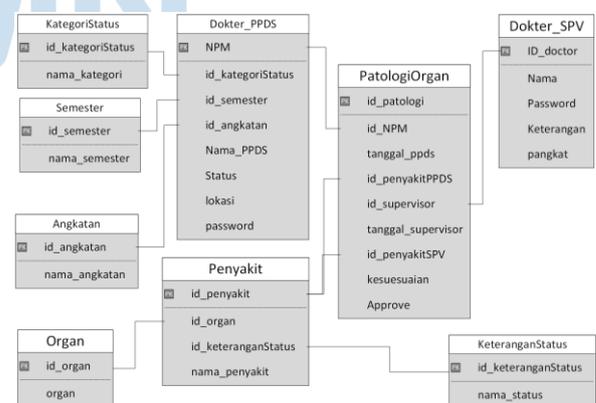


Gambar 3. Activity Diagram

C.3 Activity Diagram

Pada Gambar 3, *Activity Diagram* menjelaskan proses dan aktivitas *actor* yang di dalam aplikasi *E-logbook*, aplikasi di mulai ketika PPDS melakukan login, dan menginput *logbook*, lalu *logbook* yang di input akan otomatis terkirim ke Supervisor untuk dilakukannya pengecekan. Supervisor melakukan pengecekan yang akan memberikan nilai berupa “Sesuai”, “Kurang Sesuai”, “Tidak Sesuai”, setelah supervisor memberikan penilaian maka aplikasi akan secara otomatis menambah nilai PPDS yang bersangkutan yang nantinya akan di akumulasi sebagai syarat kelulusan mereka jika nilai tidak sesuai standar kelulusan maka PPDS harus mengimput *logbook* hingga nilai mencukupi standar kelulusan. SPS hanya dapat memantau kegiatan yang di lakukan oleh PPDS.

C.4 Entity Relationship Diagram (ERD)

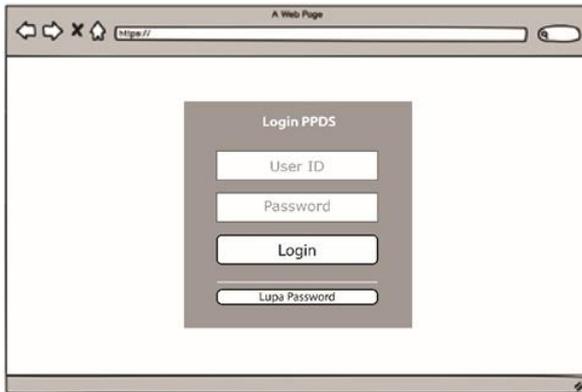


Gambar 4. ERD diagram

Dengan data yang belum di olah dari beberapa sumber dan *logbook* yang bersifat manual maka dapat di rancang dan di gambarkan *design database*-nya menggunakan notasi *crowfoot* pada susunan data yang telah di normalisasi lalu di gambarkannya dalam

bentuk *Entity Relationship Diagram* (ERD) yang dapat di lihat pada Gambar 4.

C.5 Wireframe



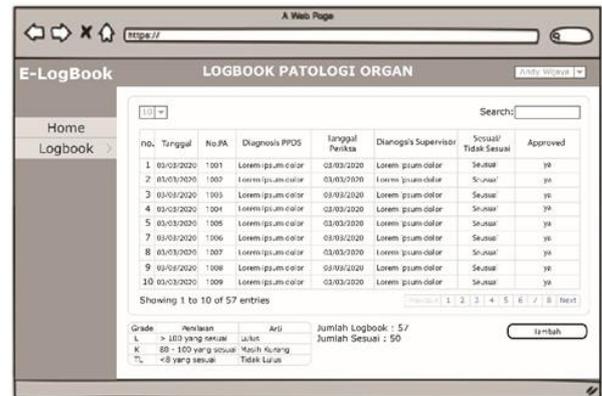
Gambar 5. Form login

Pada Gambar 5 menunjukkan *form login* yang berfungsi untuk *login* untuk dokter PPDS. Berfungsi untuk membatasi hak akses bagi untuk orang orang yang dapat mengakses *website* ini.



Gambar 6. Form home

Setelah selesai melakukan *login*, *User* akan memasuki halaman *home* yang seperti pada Gambar 6, menunjukkan *form home* yang berfungsi sebagai tampilan pertama setelah berhasil *login* yang terdapat beberapa menu yang dapat di pilih sesuai kebutuhan untuk melanjutkan proses yang selanjutnya.



Gambar 7. Form logbook patologi organ

Pada Gambar 7 menunjukkan *form logbook* patologi organ menampilkan data analisis dari dokter PPDS yang di catat dalam *logbook* tersebut dan menampilkan hasil penilaian dan koreksi dari dokter supervisor. Kemudian *form* ini dapat melihat perkembangan yang terlampir keterangan pada bawah tabel. Dan juga ada tombol “tambah yang akan menghubungkan ke form Tambah *Logbook Pantologi Organ*”. Disana PPDS dapat memasukan nomor, tanggal, nomor pasien, dianogsis PPDS yang akan dikirimkan kepada dokter Supervisor yang kemudian akan di periksa dan diberikan penilaian. Setelah dokter supervisor memberikan penilaian table *logbook* pantologi organ akan secara otomatis meng-*update* tabel.

C.6 Implementasi

Ini adalah tahap dimana aplikasi *E-logbook* akan dibangun sesuai dengan analisis dan desain yang telah di buat pada tahap sebelumnya. Pada saat telah berhasil dibangun sesuai dengan tahapan-tahapan sebelumnya. Implementasi dan sudah diuji oleh pengembang maka *system prototype* telah dapat diuji coba oleh *user*. Setelah *user* menguji coba sistem dapat disimpulkan bahwa sistem telah memenuhi kebutuhan dan semua fitur berjalan dengan baik dan tidak ada keluhan. Selain itu tidak ada fitur yang harus kembangkan. Dari pendapat *user*, dapat disimpulkan bahwa sistem dapat diterapkan ke tahapan selanjutnya yaitu implementasi dari *system prototype* menjadi sistem final yang akan digunakan dalam perusahaan.

C.7 Sistem

Pada tahap ini aplikasi yang dibangun sudah selesai dan telah selesai dilakukannya uji coba dari pihak *user*. setelah selesai dilakukannya uji coba peneliti hanya perlu memperbaiki kekurangan dari aplikasi *E-logbook* dan menambah fitur-fitur baru untuk keperluan yang mendatang. Supervisor di aplikasi tersebut dapat mengetahui *update* dari setiap *user* dengan mudah.dan dengan begitu dapat

mengambil keputusan sebelum membuat permintaan ke divisi lain serta dapat meningkatkan kinerja dari PPDS.

## V. SIMPULAN

Berdasarkan hasil perancangan yang di buat oleh penulis dapat di simpulkan bahwa aplikasi *E-logbook* ini diharapkan dapat membantu banyak PPDS FKUI untuk mengatasi beberapa masalah yang terjadi pada saat proses mendianogsis pasien dengan sistem *E-logbook* yang semuanya terkomputerisasi, sehingga kegiatan pemantauan akademik PPDS dan pengumpulan data laporan dari PPDS semuanya dapat dilakukan dengan lebih mudah dan lebih efektif karena hampir semua kegiatan di-input ke dalam computer.

Perancangan sistem *E-logbook* pada PPDS FKUI dapat menggunakan metode SDLC *waterfall* dan sangat membantu pihak terkait dalam menyelesaikan proyek ini dengan cepat dan juga dapat beradaptasi dengan persyaratan tambahan di masa depan. Berdasarkan pengalaman ini, penulis percaya bahwa membuat aplikasi *E-logbook* ini dapat terus di kembangkan dan modul lain dapat ditambahkan sesuai dengan kebutuhan aplikasi *E-logbook* nanti. Selain itu aplikasi dapat mendukung terjadinya proses pencatatan *logbook* agar lebih cepat dan lebih mudah dalam pengumpulan dan pengolahan data menjadi informasi yang dapat dimanfaatkan oleh pemangku kepentingan untuk meningkatkan pelayanan kesehatan.

Berdasarkan perancangan dan kesimpulan yang telah dibuat oleh penulis dapat disimpulkan bahwa dari perancangan ini diperlukannya sebuah aplikasi website *E-logbook* ini diharapkan dapat dikembangkan dan disempurnakan lagi sesuai kebutuhan yang akan bertambah seiring berjalannya waktu.

## DAFTAR PUSTAKA

- [1] "Fakultas Kedokteran - Universitas Indonesia," [Online]. Available: <https://fk.ui.ac.id/program-pendidikan-dokter-2.html>.
- [2] F. U. G. Mada, "Orientasi Program Pendidikan Spesialis," [Online]. Available: [fk.ugm.ac.id](http://fk.ugm.ac.id) › Uncategorized.
- [3] FKUI, "Program Spesialis - FKUI," [Online]. Available: <https://fk.ui.ac.id/spesialis.html>.
- [4] N. Nuraini, "Analisis Sistem Penyelenggaraan Rekam Medis di Instalasi Rekam Medis RS ' X ' Tangerang Periode April-Mei 2015," *J. Adm. Rumah Sakit*, vol. 1, no. 2, pp. 147–158, 2015.
- [5] R. Dewi, "Perancangan Sistem Informasi Rekam Medis Hasil Deteksi Kaki Datar Berbasis Web," *J. Sist. Komput.*, vol. 8, no. 1, pp. 7–13, 2018, doi: 10.14710/jsk.v8i1.143.
- [6] N. Achmad, R. Agung, M. S. Moch, "Implementasi Framework Laravel Pada Sistem Informasi Pemesanan Penggunaan Lapangan Futsal Berbasis Web Di Zona6 Futsal Semarang", Prosiding SNST ke-8 Tahun 2017.
- [7] H. Dafitri and M. Elsera, "Rancang Bangun Sistem Informasi Akademik Berbasis Web ( Studi Kasus: Sma Swasta Harapan I Medan )," *J. Sist. Inf.*, vol. 1, no. 2, pp. 23–32, 2017.
- [8] F. Hakam, "Analisis Sistem Dan Teknologi Informasi Sebagai Acauan Dalam Perancangan Rencana Strategis Sistem Informasi Dan Teknologi Informasi (Renstra Si/Ti) Di Rumah Sakit Islam YOGYAKARTA Pdhi," *Sriwij. J. Inf. Syst.*, vol. 9, no. 1, 2017.
- [9] H. Prasetya, Sholih, and F. Artwodini Muktadiroh, "Perancangan dan Pembuatan Sistem Informasi Manajemen Pustaka pada Program Pendidikan Dokter Spesialis (PPDS) Anestesiologi dan Reanimasi RSUD Dr. Soetomo," *J. Tek. Pomits*, pp. 1–6, 2013.
- [10] A. Dennis, H. B. Wixom, and D. Tegarden, *Systems Analysis Design with UML Version 2.5: An Object-Oriented Approach*. 2015.
- [11] M. W. Earley, J. S. Sargent, and J. M. Caloggero, *Tenth Edition*. 2005.
- [12] "Kamus KBBI," [Online]. Available: <http://www.kamuskbbi.id/inggris/indonesia.php?mod=view&logbook&id=19226-kamus-inggris-indonesia.html>.
- [13] N. F. Dewi, "Analisis Sistem Pelayanan Rekam Medis Rawat Inap Di Rsup Dr. Kariadi Semarang Tahun 2016," *J. Vokasi Indones.*, vol. 5, no. 2, pp. 29–37, 2017, doi: 10.7454/jvi.v5i2.74.
- [14] J. Wilson and G. Morrisroe, *Systems analysis and design*. 2005.
- [15] K. Kawano, Y. Umemura, and Y. Kano, "Pemodelan Uml Sistem Informasi Monitoring Penjualan Dan Stok Barang (Studi Kasus: Distro Zhezha Pontianak)," *Crop Sci.*, vol. 23, no. 2, pp. 201–205, 1983, doi: 10.2135/cropsci1983.0011183x002300020002x.
- [16] Suendri, "Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database Oracle (Studi Kasus: UIN Sumatera Utara Medan)," *J. Ilmu Komput. dan Inform.*, vol. 3, no. 1, pp. 1–9, 2018.
- [17] A. Sukmaindrayana and R. Sidik, "Aplikasi Grosir Pada Toko RSIDIK Bungursari Tasimalaya," *J. Manaj. Inform.*, vol. 4, no. 2, pp. 1–30, 2017, doi: 10.1017/CBO9781107415324.004.
- [18] D. R. Anggitama, H. Tolle, and H. M. Az-zahra, "Evaluasi Dan Perancangan User Interface Untuk Meningkatkan User Experience Menggunakan Metode Human Centered Design Dan Heuristic Evaluation Pada Aplikasi EzyPay," *J. Pengemb. Teknol. Inf. dan Ilmu Komput. Univ. Brawijaya*, vol. 2, no. 12, pp. 6152–6159, 2018.