

**RANCANG BANGUN APLIKASI PENYEWAAN
LAPANGAN FUTSAL PADA PELANGI FUTSAL
BERBASIS WEB**

SKRIPSI



Oleh:
LEO NARDO SITIO
141510145

**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS TEKNIK DAN KOMPUTER
UNIVERSITAS PUTERA BATAM
2019**

**RANCANG BANGUN APLIKASI PENYEWAAN
LAPANGAN FUTSAL PADA PELANGI FUTSAL
BERBASIS WEB**

SKRIPSI



Oleh:
LEO NARDO SITIO
141510145

**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS TEKNIK DAN KOMPUTER
UNIVERSITAS PUTERA BATAM
2019**

**RANCANG BANGUN APLIKASI PENYEWAAN
LAPANGAN FUTSAL PADA PELANGI FUTSAL
BERBASIS WEB**

SKRIPSI

Untuk memenuhi salah satu syarat guna memperoleh gelar sarjana



**Oleh:
LEO NARDO SITIO
141510145**

**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS TEKNIK DAN KOMPUTER
UNIVERSITAS PUTERA BATAM
2019**

SURAT PERNYATAAN ORISINALITAS

Yang bertanda tangan di bawah ini saya:

Nama : Leo Nardo Sitio
NPM : 141510145
Fakultas : Teknik dan Komputer
Program Studi : Sistem Informasi

Menyatakan bahwa “**Skripsi**” yang saya buat dengan judul:

Rancang Bangun Aplikasi Penyewaan Lapangan Futsal Pada Pelangi Futsal Berbasis Web

Adalah hasil karya sendiri dan bukan “duplikasi” dari karya orang lain. Sepengetahuan saya, didalam naskah Skripsi ini tidak terdapat karya ilmiah atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip didalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata di dalam naskah Skripsi ini dapat dibuktikan terdapat unsur-unsur PLAGIASI, saya bersedia naskah Skripsi ini digugurkan dan gelar akademik yang saya peroleh dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku.

Demikian pernyataan ini saya buat dengan sebenarnya tanpa ada paksaan dari siapapun.

Batam, 12 Januari 2019

Materai 6000

Leo Nardo Sitio
141510145

**RANCANG BANGUN APLIKASI PENYEWAAN
LAPANGAN FUTSAL PADA PELANGI FUTSAL
BERBASIS WEB**

Oleh:
LEO NARDO SITIO
141510145

SKRIPSI

**Untuk memenuhi salah satu syarat
guna memperoleh gelar sarjana**

**Telah disetujui Pembimbing pada tanggal
seperti tertera di bawah ini**

Batam 27 Januari 2019

Nopriadi, S.Kom., M.Kom

Pembimbing

ABSTRAK

Futsal adalah kegiatan yang berfungsi sebagai media permainan yang saat ini banyak digeluti anak kalangan muda. Lapangan Futsal adalah tempat yang disewakan kepada orang lain dengan menyediakan fasilitas dengan harga lebih murah daripada lapangan lain. Tujuan perancangan sistem informasi ini adalah untuk membantu masyarakat mencari lapangan futsal melalui media internet dengan menggunakan website dan media elektronik lainnya, penyewaan lapangan futsal adalah sebuah jasa yang diperuntukkan untuk masyarakat atau pengguna. Penulis mengembangkan sistem informasi pelayanan jasa penyewaan lapangan futsal berbasis web ini dengan metode Extreme programming, Extreme programming adalah model pengembangan perangkat lunak yang menyederhanakan berbagai tahapan pengembangan sistem menjadi lebih efisien, adaptif dan fleksibel, untuk perancangannya penulis menggunakan UML, bahasa pemrograman PHP dan Mysql sebagai databasenya sementara untuk pengujian sistem menggunakan whitebox testing, output sistem dinamis.

Kata Kunci: Penyewaan , extream programing, PHP, whitebox testing, Uml

ABSTRACT

Futsal is an activity that functions as a media game that is currently in the hands of many young people. Futsal Field is a place that is rented out to other people by providing facilities at lower prices than other fields. The purpose of designing this information system is to help people find futsal fields through internet media using websites and other electronic media, futsal field rentals are a service that is intended for the community or users. The author develops this web-based futsal field service information system with Extreme programming method, Extreme programming is a software development model that simplifies various stages of system development to be more efficient, adaptive and flexible, for the author to use UML, PHP and Mysql programming languages as a temporary database for testing systems using whitebox testing, dynamic system output.

Keywords: *Leasing, extream programing, PHP, whitebox testing, Uml*

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Tuhan Yang Maha Esa, karena berkat rahmat dan hidayah serta karunia-Nya, sehingga penulis dapat menyelesaikan laporan tugas akhir yang merupakan salah satu persyaratan untuk menyelesaikan program studi strata satu (S1) pada program studi Sistem informasi Universitas Putera Batam.

Penulis menyadari bahwa laporan tugas akhir ini masih jauh dari sempurna. Karena itu kritik dan saran akan senantiasa penulis terima dengan senang hati. Dengan segala keterbatasan, penulis menyadari pula bahwa laporan tugas akhir ini takkan terwujud tanpa bantuan, bimbingan dan dorongan dari berbagai pihak. Untuk itu, dengan kerendahan hati, penulis menyampaikan ucapan terimakasih kepada:

1. Rektor Universitas Putera Batam Dr.Nur Elfi Husda.,S.Kom.,M.SI
2. Ketua Program Studi Sistem Informasi Universitas Putera Batam Bapak Muhammad Rasid Ridho,S.Kom., M.SI
3. Dosen Pembimbing skripsi Bapak Nopriadi, S.Kom., M.Kom
4. Dosen dan staff Universitas Putera Batam
5. Kepada Kedua Orang Tua saya yang selalu mendukung dan memberikan pelajaran berharga dalam hidup saya hingga pada tahap menyelesaikan tugas akhir ini.
6. Kepada saudara,Abang, kakak saya yang telah memberikan semangat dan sabar menghadapi tingkah laku saya.

7. Teman-teman saya sistem informasi angkatan 2014 yang telah memberikan dorongan dan motivasi sehingga skripsi ini dapat terselesaikan dengan baik
8. Teman-teman pengurus harian Unit Kegiatan Mahasiswa Kristen (UKMK) Putera Batam

Semoga Tuhan Yang Maha Esa memberikan balasan kebaikan dan selalu mencurahkan hidayah serta taufikNya bagi kita semua.

Batam, 27 Januari 2019

Leo Nardo Sitio

DAFTAR ISI

	Halaman
SURAT PERNYATAAN ORISINALITAS	ii
HALAMAN PENGESAHAN	iii
ABSTRAK	iv
ABSTRACT	v
KATA PENGANTAR	vi
DAFTAR ISI	viii
DAFTAR GAMBAR	xiii
DAFTAR TABEL	xv
DAFTAR LAMPIRAN	xvi
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2 Identifikasi Masalah	4
1.3 Batasan Masalah	4
1.4 Rumusan Masalah	5
1.5 Tujuan Penelitian	6
1.6 Mamfaat Penelitian	6
1.6.1 Secara Teoritis	6
1.6.2 Secara Praktis	7
BAB II TINJAUAN PUSTAKA	8
2.1 Tinjauan Teori Umum	8
2.1.1 Sistem	8
2.1.2 Karakteristik Sistem	10
2.1.3 Informasi	12
2.1.4 Sistem Informasi	14
2.1.5 Komponen Sistem Informasi	16

2.1.6	SDLC (<i>System Development Life Cycle</i>).....	18
2.1.7	Sejarah Perkembangan SDLC	18
2.1.8	Tahapan <i>System Development Life Cycle</i> (SDLC).....	21
2.1.9	UML (<i>Unified Modeling Language</i>)	23
2.1.10	Use Case Diagram	25
2.1.11	Class Diagram	28
2.1.12	Activity Diagram	30
2.1.13	Objek Diagram	32
2.1.14	Diagram Sequence.....	33
2.2	Tinjauan Teori Khusus	38
2.2.1.	Jasa (Penyewaan)	38
2.2.2.	Lapangan Futsal	39
2.2.3.	Sistem Informasi Penyewaan Lapangan Futsal (Jasa).....	39
2.2.4.	<i>Website</i>	39
2.2.5.	HTML (<i>Hyper Text Markup Language</i>)	40
2.2.6.	PHP.....	42
2.2.7.	MySQL (<i>My Structure Query Language</i>)	43
2.2.8.	Macromedia Dreamweaver8	44
2.2.9	<i>Extreme Programming</i> (XP)	47
2.2.9.1.	Perencanaan Kegiatan	49
2.2.9.2.	Perancangan	50
2.2.9.3.	Pengkodean.....	52
2.2.9.4.	Pengujian.....	54
2.3	Penelitian Terdahulu.....	55
BAB III METEDOLOGI PENELITIAN.....		59
3.1.	Desain Penelitian	59
3.2.	Objek Penelitian	61
3.3.	Analisa SWOT Program yang berjalan	62
4.1.	Analisa system yang sedang berjalan	64
4.2.	Aliran Sistem Informasi yang Sedang Berjalan	66
4.3.	Permasalahanyang sedang dihadapi	68

4.4.	Usulan Pemecahan Masalah	68
BAB IV ANALISA IMPLEMENTASI DAN PEMBAHASAN		70
4.1	Analisa Sistem Yang Baru	70
4.1.1	Aliran Sistem Informasi Yang Baru	71
4.1.2	Use Case Diagram	73
4.1.3	Activity Diagram	74
4.1.3.1.	<i>Activity Diagram Login Sistem</i>	74
4.1.3.2.	<i>Activity Diagram Member Pada Sistem</i>	75
4.1.3.3.	<i>Activity Diagram Admin Verifikasi Member</i>	76
4.1.3.4.	<i>Activity Diagram Admin Edit Data Lapangan</i>	77
1.1.3.5.	<i>Activity Diagram Admin Tambah Data Admin</i>	78
1.1.3.6.	<i>Activity Diagram Admin Hapus Data Pesananan</i>	79
1.1.3.7.	<i>Activity Diagram Admin Tambah Data Member</i>	80
1.1.3.8.	<i>Activity Diagram Admin Edit Data Member</i>	81
1.1.3.9.	<i>Activity Diagram Admin Hapus Data Member</i>	82
1.1.3.10.	<i>Activity Diagram Member Tambah Data Member</i>	83
1.1.3.11.	<i>Activity Diagram Member Edit Data Profil</i>	84
1.1.3.12.	<i>Activity Diagram Member Menghapus Data Profil</i>	86
1.1.3.13.	<i>Activity Diagram Member Edit Profil</i>	87
4.1.4	Sequence Diagram	87
4.1.4.1.	Sequence Diagram Login Sistem	88
1.1.4.2.	Sequence Diagram <i>Member</i>	89
1.1.4.3.	Sequence Diagram Admin Verifikasi member	90
4.1.4.4.	Sequence Diagram Admin Edit Data Admin	91
4.1.4.5.	Sequence Diagram Admin Tambah Data Admin.....	92
4.1.4.6.	Sequence Diagram Admin Hapus Data <i>Member</i>	93
4.1.4.7.	Sequence Diagram Admin Tambah Data member.....	94
4.1.4.8.	Sequence Diagram Admin Edit Data member	95
4.1.4.9.	Sequence Diagram Admin Hapus Data member	96
4.1.4.10.	Sequence Diagram Member Tambah Data	97
4.1.4.11.	Sequence Diagram Member Edit Data.....	98

4.1.4.12.	Sequence Diagram Member Edit Profil	99
4.1.5	Class Diagram	99
4.2	Disain Rinci	101
4.2.1	Rancangan Layar Masukan	101
4.2.1.1	Halaman Login	101
4.2.1.2	Halaman Layar Utama website	102
4.2.1.3	Halaman Menu Home	103
4.2.1.4	Halaman Menu <i>Login</i> Pengguna	104
4.2.1.5	Halaman Menu Ketentuan Pemakaian	104
4.2.1.6	Halaman Administrator setelah Login	105
4.2.1.7	Halaman Admin Menu Home	106
4.2.1.8	Halaman Admin menu Pesanan	106
4.2.1.9	Halaman Admin menu Registrasi Registrasi Baru	107
4.2.1.10	Halaman Admin Menu Data <i>Member</i>	108
4.2.1.11	Halaman Admin Menu Data <i>Admin</i>	109
4.2.1.12	Halaman Admin Lihat Member	110
4.2.1.13	Halaman Admin Cetak Laporan.....	110
4.2.1.14	Halaman Utama Member Setelah Login.....	111
4.2.1.15	Halaman Member Menu Home.....	112
4.2.1.16	Halaman Member Menu Data Pemesanan.....	113
4.2.1.17	Halaman Member Menu Data Edit Member.....	113
4.2.1.18	Halaman Member Menu Profile.....	114
4.3	Rancangan File	114
4.4	Rencana Implementasi	117
4.4.1	Jadwalimplementation	117
4.4.2	Perkiraanbiayaimplementasi	118
4.5	Perbandingan Sistem	118
4.6	Analisis Produktifitas	119
4.6.1	SegiEfisiensi.....	120
4.6.2	SegiEfektifitas	120

BAB V SIMPULAN DAN SARAN	122
5.1 Simpulan.....	122
5.2 Saran.....	123
DAFTAR PUSTAKA	124
LAMPIRAN.....	LI
LAMPIRAN	LII
LAMPIRAN	LIII

DAFTAR GAMBAR

Gambar 3. 1 Extream Programing.....	60
Gambar 3. 2 Pelangi Futsal	62
Gambar 4. 1 Aliran Sistem Informasi penyewaan lapangan futsal	72
Gambar 4. 2 Use Case Diagram Penyewaan lapangan futsal	73
Gambar 4. 3 Activity Diagram Login Sistem	75
Gambar 4. 4 Activity Diagram Member	76
Gambar 4. 5 Activity Diagram Admin verifikasi member.....	77
Gambar 4. 6 Activity Diagram Admin Edit data Lapngan.....	78
Gambar 4. 7 Activity Diagram Admin Tambah data Admin	79
Gambar 4. 8 Activity Diagram admin hapus Pesanana Lapangan	80
Gambar 4. 9 Activity Diagram admin tambah data member	81
Gambar 4. 10 Activity Diagram Admin Edit data member	82
Gambar 4. 11 Activity Diagram Admin Hapus Data member	83
Gambar 4. 12 Activity Diagram member Tambah Data Member.....	84
Gambar 4. 13 Activity Diagram member Edit Profil	85
Gambar 4. 14 Activity Diagram member hapus data Profil.....	86
Gambar 4. 15 Activity Diagram member edit data profil	87
Gambar 4. 16 Sequence Diagram Login User.....	88
Gambar 4. 17 Sequence Diagram Member.	89
Gambar 4. 18 Sequence Diagram Verifikasi akun member	90
Gambar 4. 19 Sequence Diagram Admin Edit Data Admin	91
Gambar 4. 20 Sequence Diagram Admin Tambah Data Admin.....	92
Gambar 4. 21 Sequence Diagram Admin Hapus Data member.....	93
Gambar 4. 22 Sequence Diagram Admin Tambah Data member	94
Gambar 4. 23 Sequence Diagram Admin Edit Data member	95
Gambar 4. 24 Sequence Diagram Admin Hapus Data member.....	96
Gambar 4. 25 Sequence Diagram member Tambah Data	97
Gambar 4. 26 Sequence Diagram member Edit Data	98
Gambar 4. 27 Sequence Diagram member Edit Profil.....	99
Gambar 4. 28 Class Diagram Lapanagn Futsal.....	100
Gambar 4. 29 Tampilan login pada halaman utama website	102

Gambar 4. 30 Halaman Utama Website	102
Gambar 4. 31 Halaman Menu Home	103
Gambar 4. 32 Halaman Menu Registrasi.....	104
Gambar 4. 33 Halaman Menu Ketentuan Pemakaian Lapangan	105
Gambar 4. 34 Halaman Utama Admin	105
Gambar 4. 35 Halaman Admin Menu Home.....	106
Gambar 4. 36 Halaman Admin Registrasi Pemesanan.....	107
Gambar 4. 37 Halaman Admin menu Registrasi Registrasi Baru	107
Gambar 4. 38 Halaman Admin Menu Data Member	108
Gambar 4. 39 Halaman Admin Menu Data Admin.....	109
Gambar 4. 40 Halaman Admin Menu member Lihat member.....	110
Gambar 4. 41 Halaman Admin Menu Laporan	111
Gambar 4. 42 Halaman utama member setelah login.....	112
Gambar 4. 43 Halaman member Menu Home.....	112
Gambar 4. 44 Halaman member Menu Data	113
Gambar 4. 45 Halaman member Menu Data edit member.....	114
Gambar 4. 46 Halaman member Menu Profile.....	114

DAFTAR TABEL

Tabel 2. 1 Use Case Diagram	26
Tabel 2. 2 Class Diagram.....	29
Tabel 2. 3 Aktiviti Diagram.....	31
Tabel 2. 4 Objek Diagram	32
Tabel 2. 5 Sequence Diagram.....	33
Tabel 2. 6 Penelitian Terdahulu.....	55
Tabel 4. 1 <i>Table structure for table admin</i>	114
Tabel 4. 2 <i>Table structure for table about</i>	115
Tabel 4. 3 <i>Table structure for table harga</i>	115
Tabel 4. 4 <i>Table structure for table hari</i>	115
Tabel 4. 5 <i>Table structure for table jadwal</i>	115
Tabel 4. 6 <i>Table structure for table pelanggan</i>	115
Tabel 4. 7 <i>Table structure for table pemesanan</i>	116
Tabel 4. 8 <i>Table structure for table profil</i>	116
Tabel 4. 9 Jadwal Implementasi	117
Tabel 4. 10 Biaya Implementasi	118
Tabel 4. 11 Perbandingan sistem.....	118

DAFTAR LAMPIRAN

LAMPIRAN I	TAMPILAN LAYAR PROGRAM	L-I
LAMPIRAN I	HASIL SCREENSHOT DAN CODING PROGRAM	L-I
LAMPIRAN II	DAFTAR RIWAYAT HIDUP	L-II
LAMPIRAN III	SURAT PENELITIAN.....	L-III

BAB I

PENDAHULUAN

1.1. Latar Belakang

Pemerintah Indonesia telah menjadikan internet sebagai salah satu bagian penting dalam berbagai bidang. Internet saat ini telah dimanfaatkan sebagai media pembelajaran, bisnis, kemiliteran, dan berbagai bidang lain. Dengan memanfaatkan internet, Indonesia telah melakukan satu langkah maju untuk semakin berkembang dan maju pada masa yang akan datang. Kehadiran Internet yang walaupun masih merupakan industri baru yang dalam Fase pertumbuhan, yang masih terus berubah serta penuh ketidakpastian, telah memperkokoh keyakinan akan pentingnya peranan teknologi dalam pencapaian tujuan finansial perusahaan melalui modifikasi dan efisiensi proses bisnis yaitu dengan memanfaatkan *E-Commerce*. Kemampuan Internet untuk menjangkau pelanggan baru dan penghematan biaya yang cukup signifikan untuk distribusi dan pelayanan pelanggan merupakan keuntungan yang bisa didapat perusahaan dengan memindahkan roda nilai *e-commerce* ke media Internet

Pelangi Futsal adalah sebuah usaha yang memberikan jasa penyewaan lapangan futsal yang beralamatkan di Perumahan Kavling Sei Roja Blok L No. 7 Batam dan Pelangi Futsal berdiri pada tanggal 23 september 2015. Pelayanan penyewaan lapangan futsal merupakan sebuah bisnis yang berkaitan dengan olahraga yang banyak diminati dari setiap kalangan. Berolahraga futsal tentunya dapat menyegarkan tubuh, tidak heran jika saat ini persaingan antar jasa penyewaan lapangan futsal semakin banyak, khususnya dikota batam. Pelangi futsal memiliki 4 lapangan dengan ukuran serta tipe yang sama dan diisi oleh rumput-rumput yang terbuat dari plastik dan juga menyewakan rompi serta kaos kaki futsal untuk pelanggan. Penyewaan rompi dan kaos kaki futsal ini adalah bentuk solusi jika para pelanggan lupa membawa atau sama sekali tidak punya rompi ataupun kaos kaki futsal.

Pelangi futsal memiliki kendala dalam jasa penyewaan futsal, ini faktor pelayanan mempengaruhi kelancaran operasional dalam informasinya. Dengan pelayanan sistem informasi yang baik dan juga sistem yang baik dapat mengoptimalkan keuntungan pada jasa penyewaan futsal tersebut. Sistem pelayanan dan informasi dalam penyewaan lapangan futtsal saat ini masih dilakukan secara manual, seperti penyewa masih harus menuju ke tempat futsal untuk melihat lapangan futsal yang ingin disewa. Tidak adanya sistem informasi yang terkomputerisasi dalam penyewaan futsal ini menyebabkan sulitnya penyewa mendapatkan informasi dengan cepat. Dan seringkali para pelanggan harus berdebat diakibatkan tidak validnya data penyewaan yang ada pada admin . Banyak kegiatan yang dilakukan pihak pelangi futsal untuk menambah daya tarik palanggan

dengan cara membuat kegiatan berupa turnamen ataupun diskon dihari-hari yang ditentukan. Yang menjadi kendalanya adalah minimnya alat yang digunakan pihak pelangi futsal untuk mempromosikannya kepada pelanggan dan peminat olahraga futsal, hal-hal yang sering terjadi juga diakibatkan tidak adanya informasi terbaru mengenai jadwal lapangan. Penyewa kadang tidak jadi membooking lapangan futsal dikarenakan tidak adanya informasi yang tepat saat lapangan sudah penuh terpakai dan masih sulitnya pendataan keuangan karena masih menggunakan buku besar.

Dari uraian di atas maka penulis mengajukan untuk membuat suatu sistem informasi jasa penyewaan lapangan futsal berbasis web dengan menggunakan bahasa pemrograman *PHP* dan *MySQL* sebagai database server. Dengan harapan sistem ini mampu untuk meningkatkan pelayanan pada Pelangi Futsal dan mempermudah dalam pembuatan laporan. Dengan adanya perubahan sistem yang didasari pada perkembangan teknologi informasi ini diharapkan sistem yang akan dibangun dapat memberikan pengaruh kepada pemilik usaha sehingga mempermudah dalam pengaksesan oleh pengguna dalam melakukan pemesanan lapangan atau penyewaan yang lebih detail dan akurat akan menghasilkan data yang mudah diakses oleh pengguna dan lebih menguntungkan kepada pihak pemilik lapangan itu sendiri atau pemilik usaha. Karena alasan tersebut di atas, maka peneliti mengambil judul “RANCANG BANGUN APLIKASI PENYEWAAN LAPANGAN FUTSAL PADA PELANGI FUTSAL BERBASIS WEB”.

1.2 Identifikasi Masalah

Berdasarkan latar belakang yang telah dikemukakan di atas, identifikasi masalah yang dapat disimpulkan adalah sebagai berikut:

1. Kurangnya media informasi bagi pengguna lapangan untuk mendapatkan lapangan futsal secara *online* yang mudah dilakukan penyewaan secara cepat.
2. Dengan kemajuan pembangunan sarana futsal saat ini para pengembang maupun pemilik usaha lapangan futsal mengalami kesulitan mempromosikan lapangan kepada pengguna atau pihak yang membutuhkan.
3. Dengan jumlah penyedia lapangan futsal yang cukup tinggi karena banyak turnamen sesaat menyebabkan sulitnya mengetahui ketersediaan lapangan futsal.
4. Banyaknya pengguna atau peminat khususnya di bidang futsal, sehingga membutuhkan suatu media yang mempermudah mendapatkan informasi secara lebih cepat dan tepat.
5. Terbuangnya waktu yang lama untuk pemesanan lapangan futsal pada saat waktu yang mendesak atau pada saat dibutuhkan

1.3 Batasan Masalah

Adapun identifikasi masalah yang sudah diambil oleh penulis, maka agar permasalahan tidak menyebar luas, maka peneliti membuat batasan masalah sebagai berikut:

1. Penyewaan lapangan futsal berbasis web
2. Sistem hanya bias dikelola oleh admin pelangi futsal sebagai administrator
3. Menu-menu yang ada dibuat hanya sistem penyewaan lapangan futsal
4. Software yang digunakan *Adobe Dreamweaver CS* dan *MySQL*

1.4 Rumusan Masalah

Berdasarkan latar belakang, identifikasi masalah dan pembatasan masalah di atas, penulis merumuskan masalah yang akan di bahas pada skripsi ini antara lain sebagai berikut:

1. Bagaimana membangun sistem penyewaan lapangan futsal berbasis web pada pelangi futsal?
2. Bagaimana membangun sistem informasi untuk menampilkan informasi terbaru pada pelangi futsal?
3. Bagaimana membangun database pada pelangi futsal?

1.5 Tujuan Penelitian

Beranjak dari perumusan dan ruang lingkup permasalahan diatas maka akan penulis uraikan tujuan dari skripsi ini. Adapaun tujuan adalah sebagai berikut:

1. Untuk membangun sistem penyewaan lapangan futsal berbasis web
2. Untuk membangun sistem informasi yang menampilkan informasi terbaru pada pelangi futsal
3. Untuk membangun database pada pelangi futsal.

1.6 Manfaat Penelitian

Manfaat penelitian ini penulis kelompokkan menjadi 2 poin, yakni secara teoritis dan secara praktis. Yang di uraikan sebagai berikut:

1.6.1 Secara Teoritis

Secara teoritis, hasil penelitian ini diharapkan dapat menambah ilmu pengetahuan khususnya mengenai Rancang Bangun Aplikasi Penyewaan Lapangan Futsal, serta menjadi acuan untuk penelitian selanjutnya dan perbaikan dan pengembangan dari sistem yang dibangun terutama berkaitan dengan Rancang Bangun Aplikasi Penyewaan Lapangan Futsal.

1.6.2 Secara Praktis

Manfaat Praktis dari penelitian ini adalah sebagai berikut:

1. Bagi pihak perusahaan :

Dengan adanya pembuatan program ini dapat meningkatkan kualitas dan mempermudah dalam pengeolahan data pelanggan dan sistem penyewaan lapangan dengan cepat, tepat, dan akurat. Disamping itu juga dapat meningkatkan pelayanan pada pelanggan.

2. Bagi Penulis :

- a. Penelitian ini sebagai sarana untuk menambah pengetahuan dan wawasan dalam penerapan teori-teori yang sudah diperoleh dibangku kuliah.
- b. Sebagai klarifikasi untuk dapat gelar sarjana.

3. Bagi peneliti selanjutnya :

Penelitian ini dapat digunakan sebagai bahan bagi penelitian selanjutnya yang berkaitan dengan pembuatan program khususnya penelitian mengenai administrasi jasa penyewaan lapangan futsal terkhususnya sistem informasi dan database penyewaan lapangan futsal.

BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Teori Umum

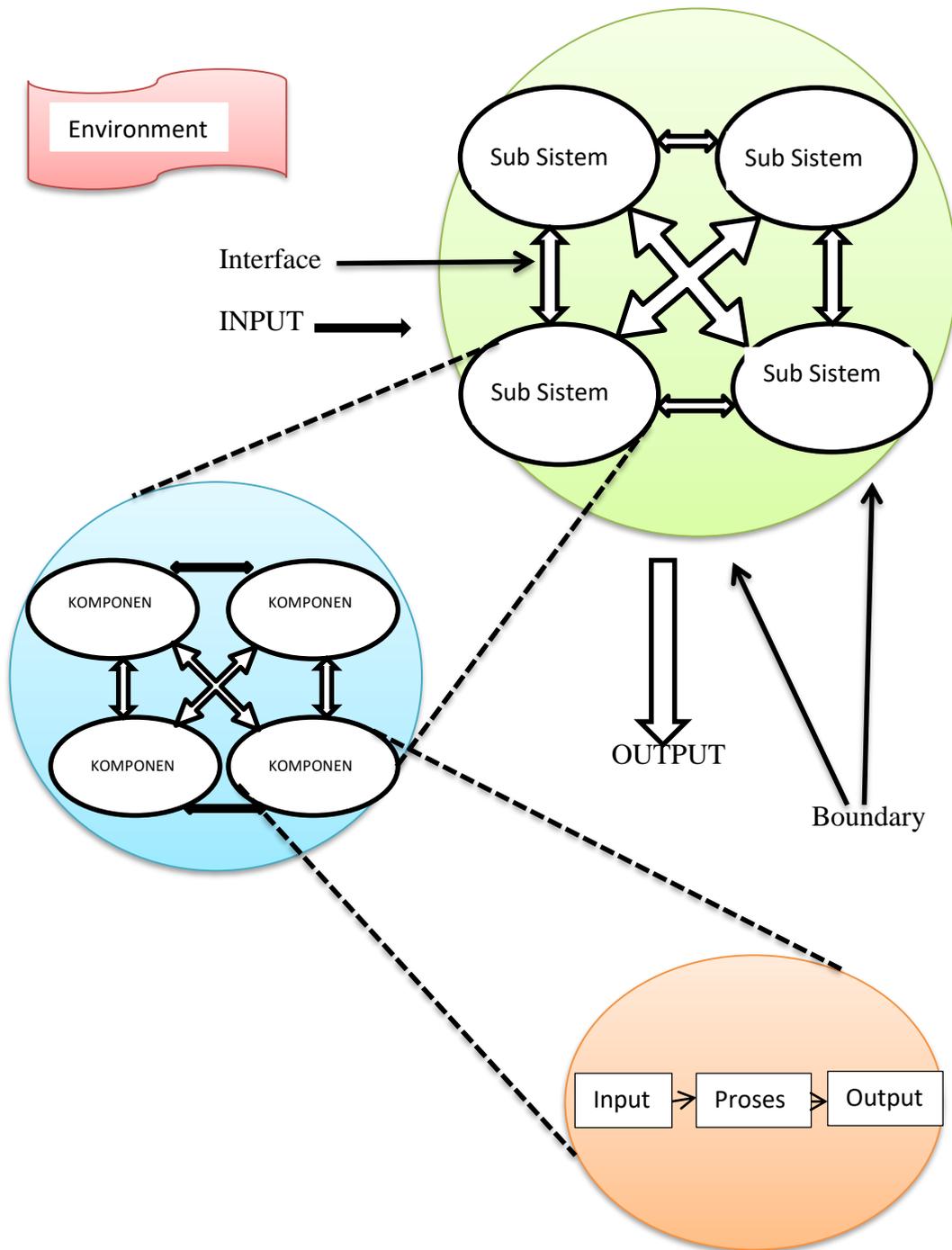
Dalam penulisan skripsi ini, penulis menggunakan teori umum dan teori khusus yang menjadi patokan dalam penulisan skripsi ini dan sebagai bahan acuan.

A theory is asset of systematically interrelated concepts, definition, and proposition that are advanced to explain and predict phenomena (fact).

2.1.1 Sistem

Sistem (*system*) adalah kumpulan dari sub-sub sistem, elemen-elemen, prosedur-prosedur, yang saling berintegrasi untuk mencapai tujuan tertentu, seperti informasi, target atau *goal*. Karakter suatu sistem terdiri dari : komponen (*Components*), Batas Sistem (*Boundary*), Lingkungan luar Sistem (*Environments*), Penghubung (*Interface*), *input*, *process* dan *output*, Sasaran (*Objectives*), Tujuan (*Goal*).

Menurut buku *Conceptual, Structure and Development*, “Sistem dapat bersifat abstrak atau *fisis*. Sistem yang abstrak adalah susunan yang teratur dari gagasan-gagasan atau konsep-konsep yang saling tergantung” (Ali Hapzi & Tonny, 2010)



Gambar 2. 1 Karakteristik Sistem

Sistem menurut (Tyoso, 2016) merupakan kumpulan dari komponen-komponen yang membentuk satu kesatuan. Sebuah organisasi dan sistem informasi adalah sistem fisik dan sosial yang ditata sedemikian rupa untuk mencapai tujuan tertentu.

2.1.2 Karakteristik Sistem

Sebuah sistem memiliki karakteristik atau sifat-sifat tertentu, yang mencirikan bahwa hal tersebut dapat dikatakan sebagai suatu sistem. Suatu sistem memiliki karakteristik atau sifat-sifat tertentu, yaitu (Muslihudin, 2016):

1. Batas Sistem

Batas Sistem (*boundary*) merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batas suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

2. Lingkungan luar (*environment*)

Lingkungan luar (*environment*) dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat bersifat merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi dari sistem dan dengan demikian harus tetap dijaga dan dipelihara. Sedang lingkungan luar yang merugikan harus ditahan dan dikendalikan, kalau tidak maka akan mengganggu kelangsungan hidup sistem.

3. Penghubung sistem (*interface*)

Penghubung (*interface*) merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Keluaran (*output*) untuk subsistem yang lainnya dengan melalui penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan.

4. Masukan Sistem (*input*)

Masukan (*input*) adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran. Sebagai contoh di dalam komputer, program adalah *maintenance input* yang digunakan untuk mengoperasikan komputernya dan data adalah *signal input* untuk diolah menjadi informasi.

5. Keluaran Sistem (*output*)

Keluaran (*output*) adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keuaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada supra sistem. Missal nya untuk sistem komputer, panas yang dihasilkan adalah keluaran yang tidak berguna dan merupakan hasil sisa pembuangan, sedang informasi adalah keluaran yang dibutuhkan.

6. Pengolahan Sistem

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukan menjadi keluaran. Suatu sistem produksi akan mengolah masukan berupa bahan baku dan bahan-bahan yang lain menjadi keluaran berupa barang jadi. Sistem akuntansi akan mengolah data-data transaksi menjadi laporan-laporan yang dibutuhkan oleh manajemen.

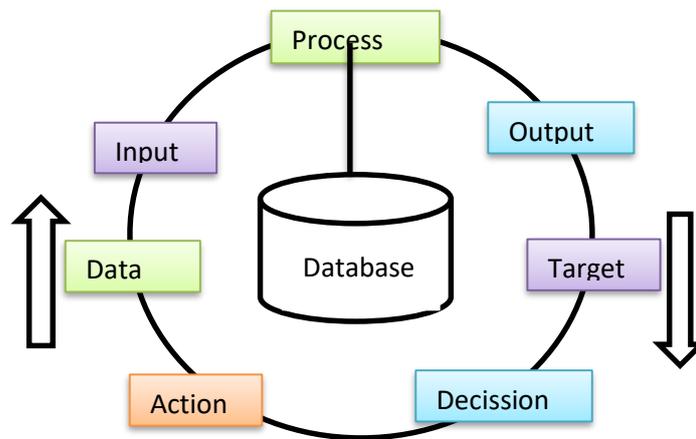
7. Sasaran Sistem (*goal*)

Suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*). Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya.

2.1.3 Informasi

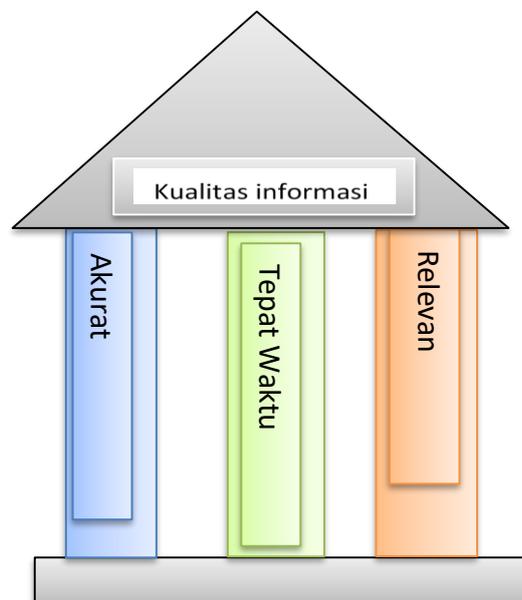
Informasi (*information*) adalah data yang telah diolah menjadi suatu bentuk yang penting bagi penerima dan mempunyai nilai yang nyata atau dapat dirasakan manfaatnya dalam keputusan-keputusan yang akan datang. *Output* informasi dari komputer digunakan oleh para Manager, non Manager, serta orang-orang dan organisasi-organisasi dalam lingkungan perusahaan. (Ali Hapzi & Tonny, 2010)

Menurut (Muslihudin, 2016) informasi adalah data yang diolah menjadi bentuk yang lebih berguna untuk membuat keputusan.



Gambar 2. 2 Siklus Informasi

Kualitas dari suatu sistem informasi (*quality of information*) tergantung dari tiga hal, yaitu informasi harus akurat (*accurate*), tepat waktunya (*timelines*) dan relevan (*relevance*). John Burch dan Gary Grudnitski menggambarkan kualitas dari informasi dengan bentuk bangunan yang ditunjang oleh tiga buah pilar.



Gambar 2. 3 Kualitas suatu Informasi

1. Akurat, berarti informasi harus bebas dari kesalahan-kesalahan dan tidak bisa menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi kemungkinan banyak terjadi gangguan (*noise*) yang dapat merubah atau merusak informasi tersebut
2. Tepat waktu, berarti informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi. Karena informasi merupakan landasan di dalam pengambilan keputusan.
3. Relevan, berarti informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi informasi untuk tiap-tiap orang satu dengan yang lainnya berbeda. Misalnya informasi mengenai sebab-musabab kerusakan mesin produksi kepada akuntan perusahaan adalah kurang relevan dan akan lebih relevan bila di tujukan kepada ahli teknik perusahaan.

2.1.4 Sistem Informasi

Sistem informasi (*information system*) merupakan suatu kumpulan dari komponen-komponen dalam suatu perusahaan atau organisasi yang berhubungan dengan proses penciptaan dan pengaliran sistem informasi. Dalam hal ini, TI merupakan salah satu komponen dalam perusahaan. Komponen-komponen yang lainnya adalah prosedur, struktur organisasi, sumberdaya manusia, produk, pelanggan, rekanan dan sebagainya. (Ali Hapzi & Tonny, 2010).

Menurut (Muslihudin, 2016) sistem informasi adalah suatu alat untuk menyajikan informasi, sedemikian rupa, sehingga bermamfaat bagi penerimanya.

Sistem informasi adalah cara yang terorganisir untuk mengumpulkan, memasukkan, dan memroses data dan menyimpannya, mengelola, mengontrol dan melaporkannya sehingga dapat mendukung perusahaan atau organisasi untuk mencapai tujuan. (Tantra, 2012)

2.1.5 Aliran Sistem Informasi

Aliran Sistem Informasi adalah suatu bagan yang menggambarkan arus logika dari data yang akan diproses dari awal suatu program sampai akhir program. Bagan alir terdiri dari simbol-simbol yang akan dikerjakan. Aliran sistem informasi merupakan bagan alir yang menunjukkan arus dari program dan formulir termasuk tembusan – tembusannya. Secara umum aliran sistem merupakan gambaran yang didapat dari rangkaian flowchart yang dibangun dalam mendefinisikan sistem, dimana system yang dibangun terdiri dari beberapa komponen yang membentuk suatu aliran system informasi, dalam membangun aliran sistem informasi harus memperhatikan simbol-simbol yang berlaku atau menjadi ketetapan dalam membangun sistem atau rancangan aliran yang dibangun dalam suatu sistem, dalam membangun aliran sistem informasi harus memahami arti atau pengertian dari simbol yang berlaku. Dari definisi dan penjelasan diatas dapatlah diambil kesimpulan, suatu sistem terdiri dari elemen yang bisa berbentuk individu atau bagian-bagian yang terpisah, kemudian berinteraksi satu sama lain untuk mencapai tujuan.

2.1.6 Komponen Sistem Informasi

(Muslihudin, 2016) John Burch dan Gary Grudnitski mengemukakan bahwa sistem informasi terdiri dari komponen-komponen yang disebutnya dengan istilah blok bangunan (*building block*), yaitu blok masukan (*input block*), blok model (*model block*), blok keluaran (*output block*), blok teknologi (*technology block*), blok basis data (*database block*) dan blok kendali (*controls block*).

1. Blok masukan (*input block*)

Input mewakili data yang masuk ke dalam sistem informasi. Input disini termasuk metode-metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

2. Blok model (*model block*)

Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok keluaran (*output block*)

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen semua pemakai sistem.

4. Blok teknologi (*technology block*)

Teknologi merupakan “kotak alat” (*tool box*) dalam sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari 3 bagian

utama, yaitu teknisi (*human atau brainware*), perangkat lunak (*software*) dan perangkat keras (*hardware*). Teknisi dapat berupa orang-orang yang mengetahui teknologi dan membuatnya dapat beroperasi.

5. Blok basis data (*database block*)

Basis data (*database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data perlu diorganisasikan sedemikian rupa, supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi dengan menggunakan perangkat lunak paket yang disebut DBMS (*Database Management System*)

6. Blok kendali (*controls block*)

Banyak hal yang dapat merusak sistem informasi, seperti misalnya bencana alam, api, temperature, air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, kesalahan-kesalahan, ketidak efisienan, sabotase dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

2.1.7 SDLC (*System Development Life Cycle*)

System Development Life Cycle disingkat dengan SDLC. SDLC merupakan siklus pengembangan sistem. (Telkom, 2009). Pengembangan sistem teknik (*engineering system development*). SDLC berfungsi untuk menggambarkan tahapan-tahapan yang secara garis besar terbagi dalam empat kegiatan utama, yaitu *initiation*, *analysis*, *design* dan *implementation*. Setiap kegiatan SDLC (*System Development Life Cycle*) dapat dijelaskan melalui tujuan (*purpose*) dan hasil kegiatannya (*deliverable*). SDLC didefinisikan oleh Departemen Kehakiman AS sebagai sebuah proses pengembangan *software* yang digunakan oleh *analyst system* untuk mengembangkan sebuah sistem informasi. SDLC mencakup kebutuhan (*requirement*), validasi, pelatihan, kepemilikan (*user ownership*) sebuah sistem informasi yang diperoleh melalui investigasi, analisis, desain, implementasi, dan perawatan *software*. *Software* yang dikembangkan berdasarkan SDLC akan menghasilkan sistem dengan kualitas yang tinggi, memenuhi harapan penggunanya, tepat dalam waktu dan biaya, bekerja dengan efektif dan efisien dalam infrastruktur teknologi informasi yang ada atau yang direncanakan, serta murah dalam perawatan dan pengembangan lebih lanjut.

2.1.8 Sejarah Perkembangan SDLC

(Telkom, 2009) Sejarah perkembangan *System Development Life Cycle* (SDLC) diawali pada pertengahan tahun 60-an dimana terjadi kegagalan yang sangat besar dalam penerapan aplikasi EDP (*Electronic Data Processing*) untuk

sistem-sistem besar, sebagian besar disebabkan tidak adanya pengembangan sistem.

Sesudah terjadinya kegagalan tersebut pada akhir tahun 60-an dan awal 70-an, kesadaran akan pentingnya metodologi pengembangan sistem mulai tumbuh. Sejak itulah berbagai proposal metodologi mulai dibuat dan penerapan mulai terlihat. Para desainer dari hampir semua bidang metodologi pengembangan sistem informasi mempunyai pandangan yang sama, yaitu: mereka telah mengetahui bahwa proses pengembangan sistem informasi, baik yang berdasarkan komputer atau tidak, menyerupai dengan proses pengembangan sistem *engineering*. Hubungan dengan konstruksi dan operasi berbagai jenis gedung, mesin, peralatan kimia yang merupakan contoh perkembangan sistem informasi *engineering*, kita dapat meringkas tahap-tahap proses secara umum perkembangan tersebut adalah perencanaan (*planning*), analisis (*analysis*), desain (*design*), pelaksanaan (*implementation*) dan perawatan (*maintenance*).

Dalam tahap perencanaan, kita mengumpulkan informasi tentang permasalahan serta persyaratannya. Kemudian kita menentukan kriteria dan pembatasan pemecahan, serta memberikan alternatif jalan keluarnya. Dalam tahap analisis, kita menguji alternatif pemecahan berdasarkan kriteria dan batasan-batasan. Analisis merupakan pusat dari semua proses perkembangan. Tahap berikutnya yaitu desain, dapat dikatakan sebagai hasil dari sistem baru. Tahap desain juga dapat dikatakan sebagai pemecahan yang optimum atas sejumlah kebutuhan penting dari suatu set pada keadaan khusus atau sebagai kegiatan kreativitas yang meliputi pembuatan barang baru dan berguna yang belum pernah

ada sebelumnya. Sistem yang tersusun dibentuk dan dioperasikan. Perawatan dilakukan pada tiap sistem operasional. Istilah daur/siklus hidup (*life cycle*) pada suatu sistem digunakan untuk menjelaskan tahap-tahap perkembangan sistem, serta langkah-langkah dalam proses perkembangannya. Untuk mengetahui proses sistem informasi dan proses sistem engineering, kita harus membandingkan daur/siklus hidup kedua sistem tersebut. Dengan mengetahui daur/siklus hidup sistem informasi tahun 1960 sampai dengan tahun 1983, kita akan mengetahui perbedaannya.

Daur hidup sistem informasi sangat dekat dengan daur hidup yang terjadi dalam sistem *engineering*; perencanaan, analisis, desain, pelaksanaan, dan perawatan. Proses perkembangan sistem informasi merupakan proses *engineering*. Meskipun selama hampir dua puluh tahun putaran sistem informasi, yang kurang lebih berisi langkah-langkah yang sama, namun pemberian nama dan dukungan pada langkah-langkah tersebut belum cukup untuk mengembangkan sistem informasi yang baik. Kekurangan tersebut adalah bahwa pada tiap perkembangan sistem *engineering* terdapat beberapa peralatan dan metodologi yang digunakan secara paralel dengan daur/siklus hidup sistem tersebut. Kegagalan dalam menentukan tuntutan dan peran serta pemakai dalam perkembangan sistem juga penyebab lain dari kegagalan sistem informasi, demikian juga masalah sulitnya memperoleh komputer dari produsen, staf yang tidak memenuhi syarat, batas waktu yang tidak realistis dan manajemen yang tidak memadai.

Kesalahan. Kesalahan interpretasi mengenai tahap-tahap perkembangan sistem di atas adalah linier. Seolah olah semua fase dan tahap terlihat berderet secara

berurutan. Tetapi sebenarnya tidak demikian. Semua tahap pada proses perkembangan sistem tersebut mempunyai sifat dasar yang iteratif yaitu pekerjaan pada suatu tahap sering harus diulang-ulang, dan apa pun yang dikerjakan pada suatu tahap mungkin perlu dikoreksi secara keseluruhan.

Meskipun terdapat beberapa variasi diantara masing-masing tahap, metode sistem klasik ternyata tidak cukup untuk menghasilkan sistem informasi yang baik, kemudian sebagai tambahan pada penamaan tahap-tahap dari suatu daur/siklus hidup sistem, kita harus mempunyai beberapa peralatan dan teknik baku untuk mengembangkan sistem tersebut.

2.1.9 Tahapan *System Development Life Cycle* (SDLC)

SDLC (*System Development Life Cycle*) meliputi tahapan secara global adalah sebagai berikut: (Shalahuddin.M, 2013)

1. Inisiasi (*initiation*), tahap ini biasanya ditandai dengan pembuatan proposal proyek perangkat lunak.
2. Pengembangan konsep sistem (*System concept development*), mendefinisikan lingkup konsep termasuk dokumen lingkup sistem, analisis manfaat biaya, manajemen rencana, dan pembelajaran kemudahan sistem.
3. Perencanaan (*Planning*), mengembangkan rencana manajemen proyek dan dokumen perencanaan lainnya. Menyediakan dasar untuk mendapatkan sumber daya (*resource*) yang dibutuhkan untuk memperoleh solusi.

4. Analisa kebutuhan (*Requirements analysis*), menganalisis kebutuhan pemakai sistem perangkat lunak (*user*) dan mengembangkan kebutuhan *user*. Membuat kebutuhan fungsional.
5. Desain (*design*), mentransformasikan kebutuhan detail menjadi kebutuhan yang sudah lengkap, dokumen desain sistem focus pada bagaimana dapat memenuhi fungsi-fungsi yang dibutuhkan.
6. Pengembangan (*Development*), mengkonversi desain sistem informasi yang lengkap termasuk bagaimana memperoleh dan melakukan instalasi lingkungan sistem yang dibutuhkan; membuat basis data dan mempersiapkan prosedur kasus pengujian; mempersiapkan berkas atau file pengujian, pengodean, pengompilasian, memperbaiki dan membersihkan program; peninjauan pengujian.
7. Integrasi dan pengujian (*integration and test*), mendemonstrasikan sistem perangkat lunak bahwa telah memenuhi kebutuhan yang dispesifikasikan pada dokumen kebutuhan fungsional. Dengan diarahkan oleh staf penjamin kualitas (*quality assurance*) dan *user*. Menghasilkan laporan analisis pengujian.
8. Implementasi (*implementation*), termasuk pada persiapan implementasi, implementasi perangkat lunak pada lingkungan produksi (lingkungan pada *user*) dan menjalankan resolusi dari permasalahan yang teridentifikasi dari fase integrasi dan pengujian.
9. Operasi dan pemeliharaan (*operations and maintenance*), mendeskripsikan pekerjaan untuk mengoperasikan dan memelihara sistem informasi pada

lingkungan produksi (lingkungan pada *user*), termasuk implementasi akhir dan masuk pada proses peninjauan.

10. Disposisi (*disposition*), mendeskripsikan aktifitas akhir dari pengembangan sistem dan membangun data yang sebenarnya sesuai aktivitas *user*.

2.1.10 UML (*Unified Modeling Language*)

UML (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. (Shalahuddin.M, 2013) UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, dan dokumentasi dari sistem perangkat lunak.

Bahasa pemrograman berorientasi objek yang pertama dikembangkan dikenal dengan nama Simula-67 yang dikembangkan pada tahun 1967. Bahasa pemrograman ini kurang berkembang dan dikembangkan lebih lanjut, namun dengan kemunculannya telah memberikan sumbangan yang besar pada developer pengembang bahasa pemrograman berorientasi objek selanjutnya. Perkembangan aktif pemrograman berorientasi objek mulai menggeliat ketika perkembangan bahasa pemrograman Smalltalk pada awal 1980-an yang kemudian diikuti dengan perkembangan bahasa pemrograman berorientasi objek lainnya seperti C objek, C++, Eiffel dan CLOS. Sekitar lima tahun setelah Smalltalk berkembang, berkembang pula metode pengembangan berorientasi objek. Metode yang pertama

diperkenalkan oleh Sally dan Stephan Mellor (Shlaer-Mellor, 1988) dan Peter Coad dan Edward Yourdon (Coad-Yourdon, 1991).

Karena banyaknya metodologi-metodologi yang berkembang pesat saat itu, maka muncullah ide untuk membuat sebuah bahasa yang dapat dimengerti semua orang. Usaha penyatuan ini banyak mengambil dari metodologi-metodologi yang berkembang saat itu. Maka dibuat bahasa yang merupakan gabungan dari beberapa konsep seperti konsep *Object Modelling Technique* (OMT) dari Rumbaugh dan Booch (1991), konsep *The Classes, Responsibilities Collaborators* (CRC) dari Rebecca Wirfs-Brock (1990), konsep pemikiran Ivar Rumbaugh, Grady Booch, dan Ivar Jacobson bergabung dalam sebuah perusahaan yang bernama *Rational Software Corporation* menghasilkan bahasa yang disebut *Unified Modeling Language* (UML).

Pada tahun 1996 *Object Management Group* (OMG) mengajukan proposal agar adanya standarisasi pemodelan berorientasi objek dan pada bulan September 1997 UML diakomodasi oleh OMG sehingga sampai saat ini UML telah memberikan kontribusinya yang cukup besar di dalamnya. Secara fisik, UML adalah kumpulan spesifikasi yang dikeluarkan oleh OMG. UML terbaru adalah UML 2.3 yang terdiri dari 4 macam spesifikasi, yaitu *Diagram Interchange, Specification*, *UML Infrastructure*, dan *Object Constraint Language* (OCL).

Penjelasan pembagian kategori dan macam-macam diagram, (Shalahuddin.M, 2013)

1. *Structure diagrams* yaitu diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan

2. *Behavior diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
3. *Interaction diagrams* yaitu kumpulan diagram yang digunakan untuk memnggambarkan interaksi sistem dengan sistem lain maupun interaksi antar subsistem pada suatu sisem

2.1.11 Use Case Diagram

Use case atau diagram *use case* merupakan pemodelan kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih actor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. (Shalahuddin.M, 2013)

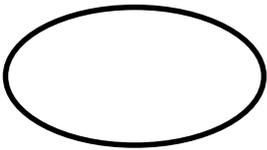
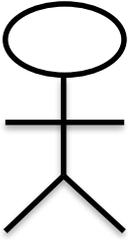
Syarat penamaan pada *use case* adalah nama didefenisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut *actor* dan *use case*

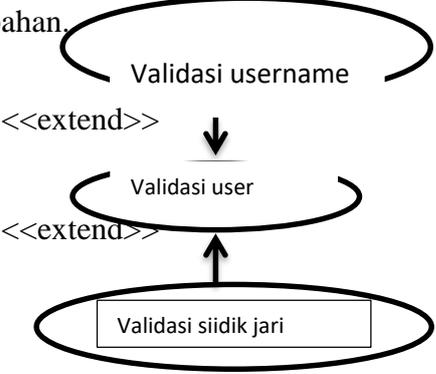
1. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang dibuat itu sendiri, jadi walaupun symbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
2. Use case merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang salaing bertukar pesan antar unit atau aktor.

Menurut (Triandini, 2012) *use case* merupakan tabel yang digunakan untuk membuat dan menjelaskan keterangan terperinci mengenai tiap-tiap *use case*.

Salah satu langkah awal untuk membuat Diagram Use Case adalah dengan mengidentifikasi aktor dan proses bisnis dasar. Setelah peran aktor teridentifikasi langkah selanjutnya adalah menyusun tujuan-tujuan yang ingin dicapai oleh peran-peran tersebut dalam penggunaan sistem. (Triandini, 2012)

Tabel 2. 1 Use Case Diagram

Simbol	Deskripsi
	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor biasanya dikatakan dengan menggunakan kata kerja awal di awal frase nama use case
<i>Aktor/actor</i>  <i>actor</i>	Orang, prose atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi. Biasanya dinyatakan dengan menggunakan kata benda di awal frase nama actor
<i>asosiasi/association</i> 	Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada use case atau <i>use case</i> memiliki interaksi dengan aktor

<p>Ekstensi/<i>extend</i></p> <p><code><< extend >></code></p> 	<p>Relasi <i>use case</i> tambahkan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan.</p>  <p>Arah panah mengarah pada <i>use case</i> yang ditambahkan; biasanya <i>use case</i> yang menjadi <i>extend</i>-nya merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya</p>
<p>Generalisasi/<i>generalization</i></p> <p><code><<uses>></code></p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih unun dari lainnya. Arah panah mengarah pada <i>use case</i> yang menjadi generalisasi (umum)</p>
<p>Menggunakan/<i>include/user</i></p> <p><code><<include>></code></p>  <p><code><<uses>></code></p> <p><code><<uses>></code></p> 	<p>Relasi <i>use case</i> tambahkan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini. Ada dua sudut pandang yang cukup besar mengenai <i>include</i> di <i>use case</i>:</p> <p>Include berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambhan dijalankan.</p>

2.1.12 Class Diagram

Diagram kelas atau *class* diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

1. atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas
2. atribut mendeskripsikan properti dengan sebaris teks di dalam kotak kelas tersebut.
3. operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

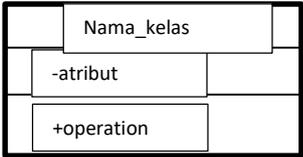
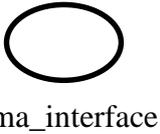
Diagram kelas mendeskripsikan jenis-jenis objek dalam sistem dan berbagai hubungan statis yang terdapat di antara mereka. Diagram kelas juga menunjukkan properti dan operasi sebuah kelas dan batasan-batasan yang terdapat dalam hubungan-hubungan objek tersebut. Diagram kelas menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti *containment*, pewarisan, asosiasi, dan lain-lain. (Telkom, 2009)

Susunan struktur kelas yang baik pada diagram kelas sebaiknya memiliki jenis-jenis berikut:

1. kelas main: kelas yang memiliki fungsi awal dieksekusi ketika sistem dijalankan.
2. Kelas *view*: kelas yang mendefinisikan dan mengatur tampilan ke pemakai
3. Kelas *controller* kelas yang menangani fungsi-fungsi yang harus ada diambil dari pendefinisian *use case*.

4. Kelas model: kelas yang digunakan untuk memegang atau membungkus data menjadi sebuah kesatuan yang diambil maupun akan disimpan ke basis data.

Tabel 2. 2 Class Diagram

Simbol	Deskripsi
<p>Kelas</p> 	<p>kelas pada struktur sistem</p>
<p>Antarmuka/interface</p> 	<p>Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek</p>
<p>Asosiasi/association</p> 	<p>Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain,asosiasi biasanya juga disertai dengan <i>multiplicity</i></p>
<p>asosiasi berarah</p> 	<p>Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i></p>

<p>Generalisasi</p> <p style="text-align: center;"><<uses>></p> 	<p>Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus)</p>
<p>Ketergantungan/<i>dependency</i></p> 	<p>Relasi antar kelas dengan makna ketergantungan antar kelas</p>
<p>Agregasi/aggregation</p>  <p style="text-align: center;">* 1 -End6nd5</p>	<p>Relasi antar kelas dengan makna semuabagian (whole-part)</p>

2.1.13 Activity Diagram

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada di perangkat lunak. (Shalahuddin.M, 2013) Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Diagram aktivitas juga banyak digunakan untuk memdefinisikan hal hal berikut: (Shalahuddin.M, 2013)

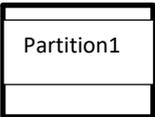
1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefenisikan

2. Urutan atau pengelompokan tampilan dari sistem/*user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
4. Rancangan menu yang ditampilkan pada perangkat lunak

Diagram aktivitas merupakan suatu jenis diagram khusus dari *statechart* diagram yang merepresentasikan state-state dan transisi-transisi yang terjadi pada akhir operasi-operasi. Diagram aktivitas dapat digunakan untuk merepresentasikan state-state serta transisi-transisi yang bersifat sinkron, sementara *statechart* diagram digunakan untuk merepresentasikan keadaan-keadaan asinkron. (Nugroho, 2009)

Tabel 2. 3 **Activity Diagram**

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem sebuah diagram aktivitas memiliki sebuah status awal.
aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
Percabangan/ <i>decisison</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Penggabungan/ <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu

status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

2.1.14 Objek Diagram

Diagram objek menggambarkan struktur sistem dari segi penamaan objek dan jalannya objek dalam sistem. Pada diagram objek harus dipastikan semua kelas yang sudah didefinisikan pada diagram kelas harus dipakai objeknya, karena jika tidak, pendefinisian kelas itu tidak dapat dipertanggungjawabkan. Diagram objek juga berfungsi untuk mendefinisikan contoh atau isis dari atribut tiap kelas. (Shalahuddin.M, 2013)

Hubungan *link* pada diagram objek merupakan hubungan memakai dan dipakai dimana dua buah objek akan dihubungkan oleh *link* jika objek yang dipakai oleh objek lainnya.

Berikut adalah simbol-simbol yang ada pada diagram objek: (Shalahuddin.M, 2013)

Tabel 2. 4 **Objek Diagram**

Simbol	Deskripsi
Objek 	Objek dari kelas yang berjalan saat sistem dijalankan

<i>Link</i> 	Relasi antar objek

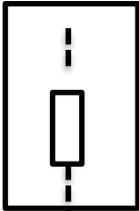
Objek diagram “*is the diagrams that shows a complete or partial view of the structure of a modelled system at the specifict time*” adalah diagram yang menunjukkan tampilan lengkap atau sebagian dari struktur sistem model pada waktu yang ditentukan. (Gopal & Patil, 2011)

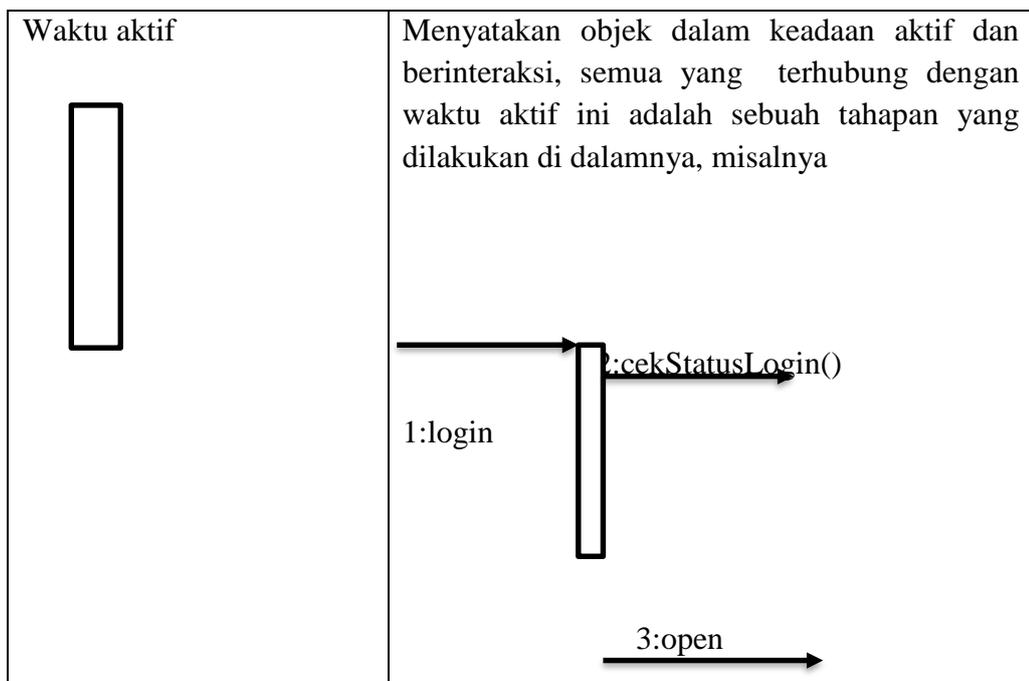
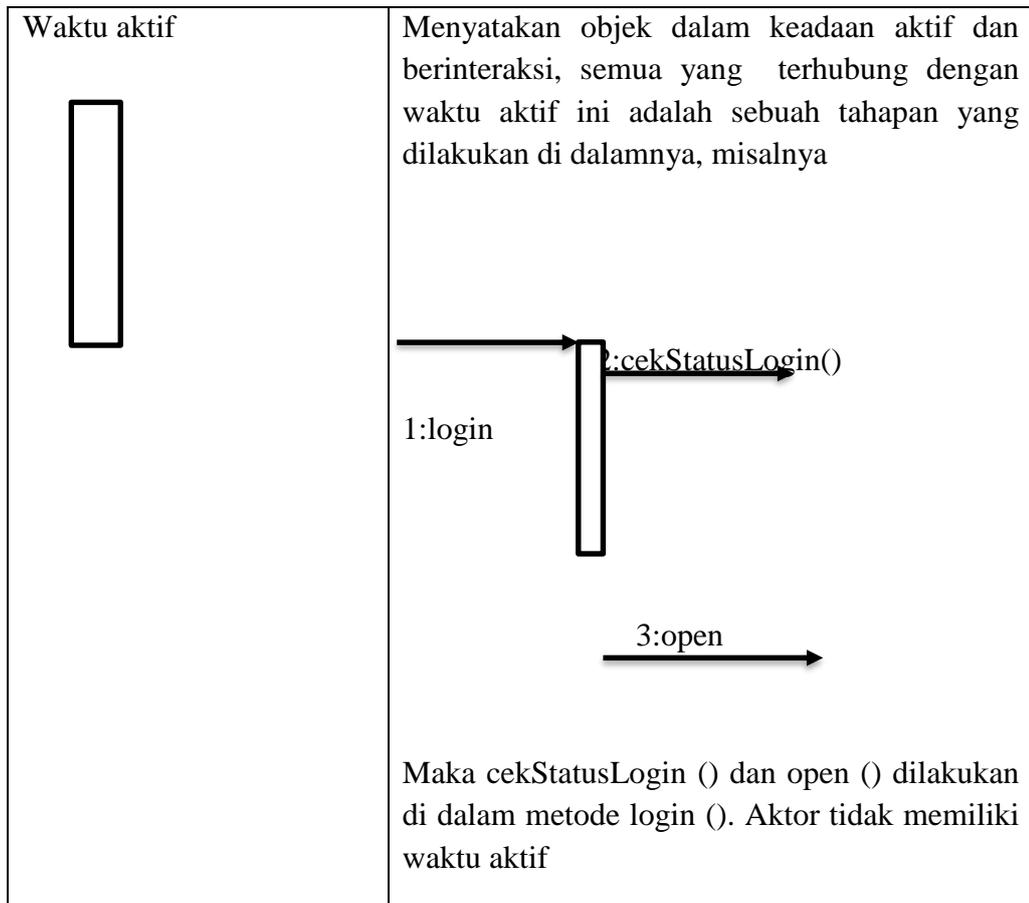
2.1.15 Diagram Sequence

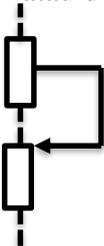
Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat scenario yang ada pada *use case*. (Shalahuddin.M, 2013)

Tabel 2. 5 Sequence Diagram

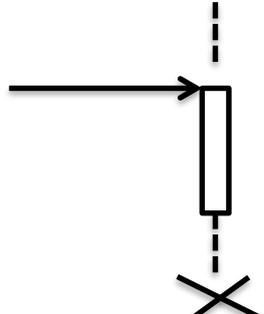
Simbol	Deskripsi
--------	-----------

<p>aktor</p>  <p>nama aktor</p> <p>atau</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;">Nama_aktor</div>	<p>Orang,proses,atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun symbol dari aktor adalah gambaran orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.</p>
<p>Garis hidup/<i>lifeline</i></p> 	<p>Menyatakan kehidupan suatu objek</p>
<p>objek</p> <div style="border: 1px solid black; padding: 2px; display: inline-block;"><u>Nama objek; nama</u></div>	<p>Menyatakan objek yang berinteraksi dengan pesan</p>



	<p>Maka cekStatusLogin () dan open () dilakukan di dalam metode login (). Aktor tidak memiliki waktu aktif</p>
<p>Pesan tipe <i>create</i></p> <p><<create>></p> 	<p>Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat</p>
<p>Pesan tipe call</p> <p>1 :nama_metode()</p> 	<p>Menyatakan suatu objek membuat objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri</p>  <p>1: nama_metode()</p> <p>Arah panah mengarah pada objek yang memiliki operasi/metode karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi</p>
<p>Pesan tipe send</p>	

<p>1 : masukan</p> 	<p>Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim</p>
--	--

<p>Pesan tipe return</p> <p>1 : keluaran</p> 	<p>Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian</p>
<p>Pesan tipe destroy</p> <p><<destroy>></p> 	<p>Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaliknya jika ada <i>create</i> maka <i>destroy</i></p>

Penomoran pesan berdasarkan urutan interaksi pesan. Penggambaran letak pesan harus berurutan, pesan yang lebih atas dari lainnya adalah pesan yang

berjalan terlebih dahulu. Semua metode di dalam kelas harus ada didalam diagram kolaborasi atau sekuen, jika tidak ada berarti perancangan metode di kelas itu kurang baik. Hal ini dikarenakan ada metode yang tidak dapat dipertanggung jawabkan kegunaanya. (Shalahuddin.M, 2013)

2.2 Tinjauan Teori Khusus

2.2.1. Jasa (Penyewaan)

Jasa adalah setiap tindakan atau aktivitas dan bukan benda, yang dapat ditawarkan oleh suatu kepada pihak lain, yang pada dasarnya bersifat intangible (tidak berwujud fisik), konsumen terlibat secara aktif dalam proses produksi dan tidak menghasilkan kepemilikan sesuatu. jasa merupakan suatu fenomena yang rumit. Kata “jasa” (*service*) itu sendiri mempunyai banyak arti, dan ruang lingkup, dari pengetahuan yang paling sederhana, yaitu hanya berupa pelayanan dari seseorang kepada orang lain, bisa juga diartikan sebagai mulai dari pelayanan yang diberikan oleh manusia, baik yang dapat dilihat (*explicit service*) maupun yang tidak dapat dilihat, yang hanya bisa dirasakan (*implicit service*) sampai kepada fasilitas-fasilitas pendukung yang harus tersedia dalam penjuwaan jasa. (Dinamika & Jamabi, 2017)

Philip Kotler menjelaskan bahwa jasa merupakan setiap tindakan atau unjuk kerja yang ditawarkan oleh salah satu pihak ke pihak lain yang secara prinsip intangible dan tidak menyebabkan perpindahan kepemilikan apapun. (Wibowo & Sismoro, 2012)

2.2.2. Lapangan Futsal

Lapangan adalah tempat atau tanah yang luas, tempat (gelanggang) pertandingan, bidang (kerjaan, pengetahuan, dan sebagainya).

Futsal adalah permainan sepak bola, dengan lapangan dan gawang lebih kecil, biasanya dimainkan di dalam ruang yang besar, setiap tim terdiri atas lima orang. (kbbi.kemdikbud.go.id, 2017)

2.2.3. Sistem Informasi Penyewaan Lapangan Futsal (Jasa)

Sistem informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan laporan yang dibutuhkan. (Dinamika & Jamabi, 2017). Sistem informasi penyewaan lapangan futsal (jasa) diartikan sebagai satu pembuatan pernyataan layanan, sedangkan kegiatannya akan dijelaskan melalui prosedur – prosedur yang meliputi urutan kegiatan dimulai dari jenis layanan, produk, kemudian dilakukannya aktivitas layanan (*service*) seperti diterimanya layanan (*service*) dari tukang, pelaksanaan pengerjaan yang disertai dengan pembuatan transaksi dan pencatatan atas layanan (*service*) tersebut.

2.2.4. Website

Website adalah halaman informasi yang disediakan melalui jalur internet sehingga bisa diakses dimana pun selama anda terkoneksi dengan jaringan internet. *Website* merupakan komponen atau kumpulan komponen yang terdiri dari teks, gambar, suara, animasi, sehingga lebih merupakan media informasi yang menarik

untuk dikunjungi. (Bsi, 2016). Web pertama kali diciptakan pada tahun 1991 di CERN, Laboratorium Fisika Partikel Eropa, Jenewa, Swiss. Tujuan awal diciptakannya web adalah untuk menciptakan media yang mudah untuk berbagi informasi para ilmuwan. Web menggunakan protocol yang disebut HTTP (*HyperText Transfer Protocol*) yang berjalan pada TCP/IP. Dokumen web ditulis dalam format HTML (*HyperText Markup Language*). Dokumen ini diletakkan dalam web *server* (*server* yang melayani permintaan halaman web) dan diakses oleh klien (pengakses informasi) melalui perangkat lunak yang disebut web *browser* atau sering disebut *browser* saja.

2.2.5. HTML (*Hyper Text Markup Language*)

HTML (*Hyper Text Markup Language*) bisa disebut bahasa paling dasar dan penting yang digunakan untuk menampilkan dan mengelola tampilan pada halaman website. HTML digunakan untuk menampilkan berbagai informasi didalam sebuah penjelajah *web* internet dan *formatting hypertext* sederhana yang ditulis ke dalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegrasi. (Saputra Agus, 2012)

Sejarah html berawal pada tahun 1980 ketika IBM berniat untuk membuat suatu bahasa kode untuk menggabungkan teks dengan pemformatan agar mengenali elemen dokumen. Bahasa yang menggunakan tanda-tanda ini dinamakan *Markup Language*. Namin pihak IBM memberi nama *Generalized Markup Language* (GML).

Dokumen html memiliki struktur yang harus kita ikuti aturan pembuatannya. Beberapa elemen-elemen htm yang wajib ada pada html adalah berikut:

1. Elemen HTML

Elemen HTML merupakan tag dasar apabila kita ingin memulai suatu dokumen html. Secara logika, jika kita menemukan tag ini, berarti secara jelas dapat didefinisikan bahwa dokumen ini merupakan perintah suatu dokumen html. Tag ini merupakan perintah wajib bagi pemrogram web untuk menuliskan tag pertama dalam dokumen html

```
<html> .....</html>
```

2. Elemen *Head*

Head merupakan tag berikutnya setelah elemen html, yang berfungsi untuk menuliskan keterangan tentang dokumen web yang akan ditampilkan.

Elemen ini nantinya akan diakhiri dengan tanda penutup <head>. Contoh:

3. Elemen *Title*

Elemen *title* merupakan suatu elemen yang harus dituliskan didalam elemen *head* yang digunakan untuk memberikan judul/ informasi pada *caption browser* web tentang topik/ tema atau judul dari suatu dokumen web yang ditampilkan pada browser. Berikut struktur penggunaannya:

```
<html>
  <head>
    <title>judul</title></head></html>
```

4. Elemen *Body*

Merupakan bagian utama dalam dokumen web. Jika kita ingin menampilkan suatu teks atau informasi atau yang dikenal dengan sebutan konten, maka kita

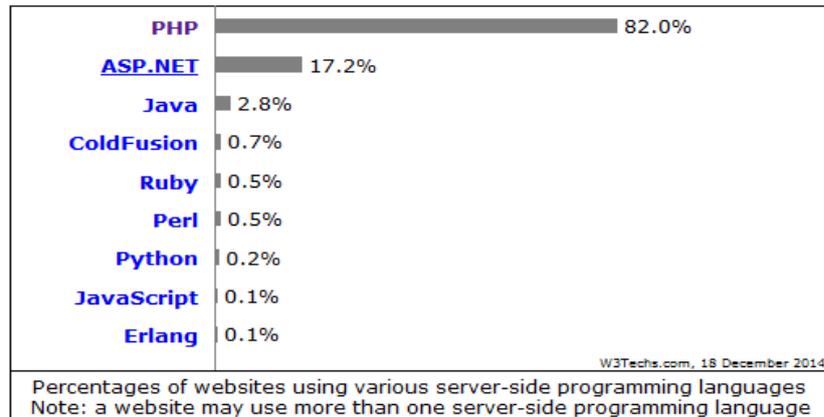
harus meletakkan teks tersebut pada elemen *body*. Struktur elemennya sebagai berikut:

```
<html>
  <head>
    <title>judul</title>
  </head>
  <body>isi konten</body>
</html>
```

2.2.6. PHP

PHP adalah *script* yang digunakan untuk membuat halaman web yang bersifat dinamis. Dinamis berarti halaman yang akan ditampilkan dibuat suatu halaman itu diminta oleh *client*. PHP juga bersifat *open source* sehingga setiap orang dapat menggunakan secara gratis. (Bsi, 2016) Pada awalnya PHP merupakan singkatan dari *Personal Home Page*. Sesuai dengan namanya, PHP digunakan untuk membuat *website* pribadi.

Dalam beberapa tahun perkembangannya, PHP menjelma menjadi bahasa pemrograman web yang powerful dan tidak hanya digunakan untuk membuat halaman web sederhana, tetapi juga *website* populer yang digunakan oleh jutaan orang seperti wikipedia, wordpress, joomla, dll. Kemudahan dan kepopuleran PHP sudah menjadi standar bagi programmer web di seluruh dunia. Dikutip dari situs w3techs.com, (diakses pada 18 Desember 2014), berikut adalah market share penggunaan bahasa pemrograman *server-side* untuk mayoritas *website* di seluruh dunia :



Gambar 2. 4 Presentasi Penggunaan Bahasa Pemrograman

2.2.7. MySQL (*My Structure Query Language*)

Sebuah *website* yang dinamis membutuhkan tempat penyimpanan data agar pengunjung dapat memberi komentar, saran, dan masukan atas *website* yang dibuat. Tempat penyimpanan data berupa informasi dalam sebuah tabel disebut dengan database. Program yang digunakan untuk mengolah dan mengelola database adalah MySQL yang memiliki kumpulan prosedur dan struktur sedemikian rupa sehingga mempermudah dalam menyimpan, mengatur, dan menampilkan data.

MySQL (*My Structure Query Language*) adalah salah satu *DataBase Management System* (DBMS) dari sekian banyak DBMS seperti Oracle, MS SQL, Postagre SQL, dan lainnya. MySQL berfungsi untuk mengolah database menggunakan bahasa SQL. MySQL bersifat *open source* sehingga kita bisa menggunakannya secara gratis. Pemrograman PHP juga sangat mendukung atau *support* dengan database MySQL. (Bsi, 2016)

2.2.8. Macromedia Dreamweaver8

Tampilan ruang kerja dari Macromedia Dreamweaver8 seperti yang terlihat pada gambar berikut:



Gambar 2. 5 Macromedia Dreamweaver8

Pada halaman awal Dreamweaver 8 terdapat beberapa menu yang dapat dipilih :

1. Open a Recent Item

Pada menu ini akan ditampilkan beberapa file yang sebelumnya pernah kita buka dengan menggunakan Dreamweaver 8. Atau di paling bawah ada Open yang dapat digunakan untuk membuka file yang lain.

2. Create New

Pada menu ini kita dapat memilih dokumen baru apa yang akan kita buat dengan menggunakan Dreamweaver 8. Adabanyak pilihan, diantaranya HTML, ColdFusion, PHP, ASP, JavaScript, CSS.

3. Create From Samples

Pada menu ini kita dapat membuat file berdasarkan contoh yang sudah diberikan oleh Dreamweaver.

4. Toolbar Dokumen

Toolbar dokumen digunakan untuk mengubah tampilan dan mengakses fungsi-fungsi penting secara cepat dan mudah. Pada *toolbar* dokumen terdapat menu untuk berpindah antar dokumen kerja window dan mengatur tampilan area kerja. Untuk mengatur tampilan kita bisa memilih *Code*, *Split* dan *Design*.

5. Menu Utama

Menu Utama berisi semua perintah yang dapat digunakan untuk bekerja pada Dreamweaver.

6. Insert Bar

Insert bar merupakan tempat semua perangkat kerja (tombol) yang digunakan untuk membuat halaman web. Insert bar mempunyai dua jenis tampilan, yaitu tampilan sebagai menu dan tampilan sebagai tab.

7. Tab common

Berisi semua tombol yang sering atau umum digunakan untuk membuat halaman web. Tombol yang ada di Tab *Common* antara lain *Hyperlink*, *Email Link*, *Named Anchor*, *Table*, *Images*.

8. Tab Layout

Tab layout digunakan untuk membuat layout halaman web. Terdapat tiga jenis layout yang dapat dipilih, yaitu *Standard*, *Expanded* dan *Layout*.

8.1. Untuk *Standard view* tampilan dokumen seperti biasa (berupa garis-garis tabel)

8.2. Untuk *Expanded view* menampilkan border tabel yang diregangkan sehingga semua rancangan tabel dapat dilihat dengan jelas baik itu baris dan kolomnya.

8.3. Untuk *Layout view* rancangan tabel ditampilkan sebagai kotak-kotak yang dapat di-drag, dan diatur ulang ukurannya dengan mudah.

9. Tab Form

Tab form digunakan untuk membuat elemen dalam form, misalnya saja *textarea, textfield, radio button, checkbox*

10. Tab Text

Tab text digunakan untuk membuat pengaturan *text*. Misalnya saja membuat *text italic, strong, underline*.

11. Tab HTML

Tab HTML digunakan untuk membuat garis horizontal, menambahkan *meta* tag dalam tag , dan *frame*.

12. Tab Application

Tab application digunakan jika aplikasi kita sudah berhubungan dengan suatu bahasa pemrograman dan sebuah database.

13. Tab Flash elements

Tab *flash elements* digunakan untuk memasukkan elemen flash dalam dokumen yang kita buat.

14. Code View

Kode *view* digunakan untuk melihat kode HTML dari halaman *web* yang sedang kita buat

15. Panel Properties

Panel properties merupakan panel yang digunakan untuk melihat dan mengubah property dari semua objek yang ada di area kerja. Masing-masing objek mempunyai property yang berbeda. Untuk melihat property dari objek yang diinginkan, seleksi dulu objek tersebut.

16. Design View

Design *view* digunakan untuk melihat tampilan *web* dari kode HTML yang kita buat.

2.2.9 Extreme Programming (XP)

Metode Extreme Programming sering juga dikenal dengan metode XP. Metode ini dicetuskan oleh Kent Beck, seorang pakar software engineering. Extreme programming adalah model pengembangan perangkat lunak yang menyederhanakan berbagai tahapan pengembangan sistem menjadi lebih efisien, adaptif dan fleksibel (Fatoni & Dwi, 2016: p.17).

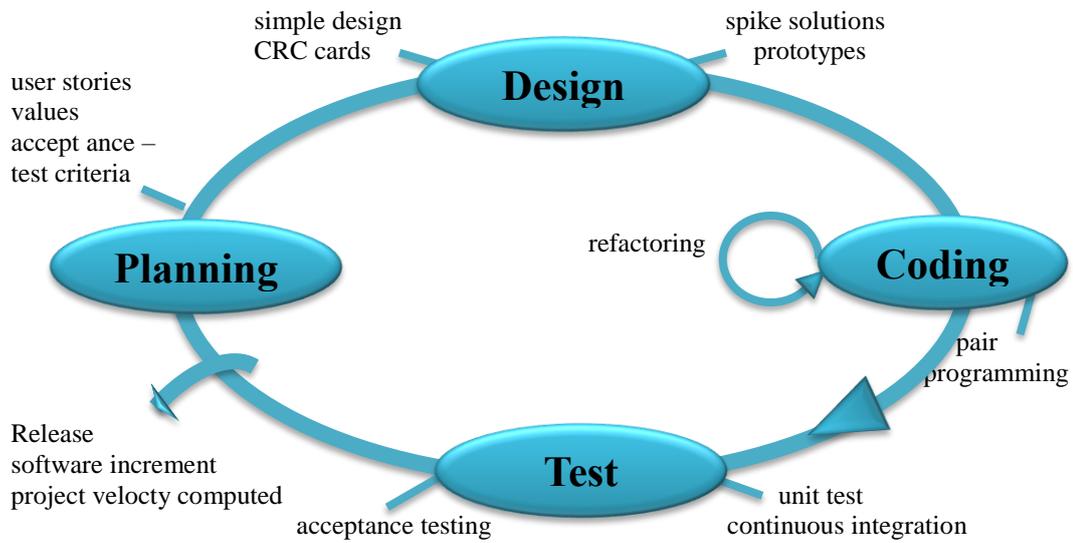
Lima nilai- nilai yang membentuk dasar bagi semua pekerjaan yang dibentuk sebagai bagian dari XP yaitu komunikasi, kesederhanaan, umpan balik, keberanian, dan rasa hormat. Masing- masing nilai yang penting ini digunakan sebagai pendorong untuk kegiatan, tindakan, dan tugas XP yang spesifik. Untuk mencapai komunikasi yang efektif yang seharusnya terjadi antara rekayasawan perangkat lunak dan para stakeholder lainnya (contoh: untuk membangun fitur-fitur dan fungsi-fungsi tertentu yang dibutuhkan bagi perangkat lunak), , menekankan pentingnya adaptasi terhadap umpan balik yang berkesinambungan, dan

menekankan pentingnya dokumentasi yang produktif sebagai suatu media komunikasi (Korowotjeng, Sengkey, Paturusi, & Tuturoong, 2014: p.4).

Extreme Programming (XP) merupakan sebuah proses rekayasa perangkat lunak yang cenderung menggunakan pendekatan berorientasi objek dan sasaran dari metode ini adalah tim yang dibentuk dalam skala kecil sampai medium serta metode ini juga sesuai jika tim dihadapkan dengan requirement yang tidak jelas maupun terjadi perubahan – perubahan requirement yang sangat cepat (Prabowo & Artwodini, 2013: p.477).

Untuk mencapai kesederhanaan, XP membatasi pengembang perangkat lunak melakukan perancangan hanya untuk kebutuhan- kebutuhan yang sifatnya mendesak alih-alih melakukan perancangan kebutuhan-kebutuhan yang diperlukan di masa depan. Tujuannya adalah untuk menciptakan rancangan yang sederhana yang dapat dengan mudah diimplementasikan dalam bentuk kode-kode program secara cepat. jika rancangan tersebut selanjutnya harus ditingkatkan, rancangan yang bersangkutan dapat di-refaktorisasi di waktu yang lain (Korowotjeng, Sengkey, Paturusi, & Tuturoong, 2014: p.4).

Pemrograman Ekstreme menggunakan suatu pendekatan 'berorientasi objek' sebagai paradigma pengembangan yang diinginkan dan mencakup di dalamnya seperangkat aturan dan praktik-praktik yang terjadi dalam konteks empat kegiatan kerangka kerja: perencanaan, perancangan, pengkodean, dan pengujian (Korowotjeng et al., 2014: p.4).



Gambar 2. 6 Extreme Programming

2.2.9.1. Perencanaan Kegiatan

Perencanaan biasanya dimulai dengan mendengarkan suatu kegiatan yang bertujuan untuk mengumpulkan kebutuhan-kebutuhan yang memungkinkan anggota teknis tim XP memahami konteks bisnis untuk perangkat lunak yang akan dikembangkan dan untuk merasakan perlunya output, fitur-fitur utama, dan fungsionalitas. Aktivitas-aktivitas mendengarkan pada dasarnya mengarah kepada pembuatan serangkaian "cerita" (juga disebut user stories) yang menggambarkan keluaran yang diperlukan, fitur-fitur, dan fungsionalitas-fungsionalitas yang akan dibangun menggunakan perangkat lunak yang akan dikembangkan. Setiap cerita umumnya ditulis oleh para pelanggan dan diletakkan pada kartu indeks. pelanggan-pelanggan memberikan suatu nilai (yaitu, suatu prioritas) pada cerita-cerita tertentu berdasarkan seluruh nilai bisnis dari fitur.

atau Fungsi Anggota tim XP kemudian menilai setiap cerita dan menetapkan biayanya yang diukur dalam bentuk minggu- minggu yang diperlukan untuk melakukan pengembangan cerita tersebut. jika cerita ini diperkirakan membutuhkan lebih dari tiga minggu pengembangan, pelanggan akan diminta untuk membagi cerita tersebut ke dalam cerita-cerita kecil dan perhitungan-perhitungan nilai dan biaya terjadi lagi. Penting untuk diperhatikan bahwa cerita baru dapat ditulis setiap saat. Pelanggan dan pengembang kemudian bekerja sama untuk memutuskan bagaimana mengelompokkan cerita kedalam kelanjutannya untuk dikembangkan oleh tim XP. Ketika komitmen dasar (kecocokan atas cerita yang dimasukkan, tanggal pengiriman, dan hal-hal lainnya yang terkait dengan proyek perangkat lunak) ini dirilis, tim XP mengatur cerita yang akan dikembangkan dalam satu dari tiga cara : (1) semua cerita akan segera dilaksanakan, (2) cerita yang memiliki nilai bisnis tertinggi akan dipindahkan ke dalam jadwal dan dilaksanakan pertama kali, atau (3) cerita yang paling berisiko akan dijadwalkan dan akan dilaksanakan pertama kali.

2.2.9.2.Perancangan

Perancangan XP dengan ketat mengikuti prinsip "tetap sederhana". Sebuah hasil perancangan yang sederhana selalu lebih disukai daripada gambar-gambaran yang lebih kompleks. Selain itu, perancangan XP akan memberikan panduan implementasi untuk suatu cerita ketika cerita itu ditulis, tidak kurang, tidak lebih. Rancangan-rancangan dan fungsionalitas-fungsionalitas tambahan (karena pengembang menganggap nantinya akan diperlukan) tidak terlalu disarankan.

XP mendorong penggunaan kartu CRC sebagai mekanisme yang efektif untuk berpikir tentang perangkat lunak dalam konteks berorientasi objek. Kartu CRC (class-responsibility-collaborator) digunakan untuk mengidentifikasi dan mengatur kelas-kelas dalam konteks "pemrograman berorientasi objek" yang relevan dengan peningkatan perangkat lunak saat ini. Tim XP selanjutnya akan melakukan latihan perancangan. Kartu CRC adalah satu-satunya produk kerja perancangan yang dihasilkan sebagai bagian dari proses pengembangan perangkat lunak cepat: XP.

Jika masalah perancangan yang sulit ditemui sebagai bagian dari perancangan suatu cerita, metode pengembangan cepat PX menyarankan pembuatan langsung dari prototipe operasional dari bagian perancangan tersebut. Disebut sebagai solusi spike, prototipe perancangan diimplementasikan dan selanjutnya dievaluasi. Tujuannya adalah untuk mengurangi risiko-risiko yang akan timbul ketika pelaksanaan yang sesungguhnya dimulai dan untuk memvalidasi keaslian perkiraan untuk cerita yang berisi masalah-masalah perancangan.

Refaktorisasi pada dasarnya adalah proses mengubah sistem perangkat lunak sedemikian rupa sehingga tidak mengubah perilaku eksternal kode, namun memperbaiki struktur internal yang ada di dalamnya. Ini merupakan cara yang disiplin untuk membersihkan kode dan memodifikasi/menyederhanakan rancangan internal, yang pada gilirannya akan meminimalkan kemungkinan akan munculnya kesalahan-kesalahan program. Pada intinya, ketika Anda melakukan refactor atau refaktorisasi, Anda pada prinsipnya memperbaiki rancangan kode-kode program Setelah kode-kode program itu ditulis. Karena perancangan perangkat lunak pada rekayasa perangkat lunak cepat (XP) hampir-hampir tidak menggunakan notasi

dan, walaupun ada, hanya menghasilkan beberapa saja, tidak menghasilkan produk-produk kerja selain kartu-kartu CRC dan solusi-solusi spike, maka perancangan pada XP dipandang sebagai buatan sementara yang dapat dan harus terus menerus diubah seiring majunya pelaksanaan konstruksi perangkat lunak. Tujuan dari refaktorisasi pada dasarnya adalah untuk mengendalikan modifikasi-modifikasi yang perlu dilakukan dengan cara menyarankan perubahan perancangan kecil yang "secara radikal dapat meningkatkan kualitas rancangan". Tetapi, dalam hal ini perlu diperhatikan bahwa upaya yang diperlukan untuk melakukan refaktorisasi dapat tumbuh secara drastis ketika ukuran aplikasi bertambah. Sebuah gagasan central dalam XP adalah rancangan yang terjadi baik sebelum dan sesudah pengkodean dimulai. Refaktorisasi berarti bahwa perancangan perangkat lunak terjadi terus-menerus ketika sistem/perangkat lunak dikonstruksi. Bahkan, kegiatan konstruksi itu sendiri akan memberikan tim XP pedoman bagaimana caranya meningkatkan kualitas rancangan.

2.2.9.3.Pengkodean.

Setelah cerita dikembangkan dan karya rancangan awal dilakukan, tim perangkat lunak cepat (XP) tidak langsung beralih ke kode-kode program, tetapi lebih dulu akan mengembangkan serangkaian unit pengujian yang akan menjalankan setiap cerita yang akan disertakan pada rilis yang ada (peningkatan perangkat lunak)."

Setelah unit pengujian dibuat, pengembang perangkat lunak akan lebih mampu berkonsentrasi pada apa yang harus diimplementasikan supaya lulus dari unit pengujian tersebut. Tidak ada yang ditambahkan (tetap sederhana). Ketika

kode- kode program telah selesai dituliskan, kode-kode program tersebut dapat langsung diuji menggunakan unit pengujian yang telah dirancang sebelumnya, sehingga bisa langsung memberikan umpan balik kepada para pengembang.

Konsep kunci selama kegiatan pengkodean adalah pemrograman berpasangan (pair programming). XP menyarankan bahwa dua orang pemrogram seharusnya bekerja sama pada satu komputer workstation untuk menuliskan kode-kode program untuk suatu cerita. Hal ini menyajikan mekanisme untuk pemecahan masalah yang dapat dilakukan secara real-time dan jaminan kualitas real-time (kode-kode program langsung ditinjau ketika ia dibuat). Hal ini juga membuat para pengembang perangkat lunak dapat berfokus pada masalah yang dihadapi. Dalam praktiknya, setiap orang mengambil peran yang sedikit berbeda. Misalnya, satu orang mungkin berpikir tentang rincian pengkodean dari bagian tertentu dari suatu perancangan, sementara orang yang lain memastikan standar pengkodean (merupakan bagian yang diharuskan dari XP) diikuti atau kode untuk cerita memenuhi unit pengujian yang telah dikembangkan untuk melakukan validasi kode terhadap cerita.

Ketika pasangan pemrogram tadi berhasil menyelesaikan pekerjaan mereka, kode-kode program yang mereka kembangkan kemudian diintegrasikan dengan hasil pengembangan sebelumnya. Dalam beberapa kasus, hal ini dilakukan biasanya oleh tim integrasi. Dalam kasus lain, pasangan pemrogram tadi sekaligus memiliki tanggung jawab untuk melakukan integrasi. Strategi "integrasi berkesinambungan" membantu untuk menghindari masalah kompatibilitas dan

interfacing dan memberikan lingkungan "smoke-testing" yang membantu menemukan kesalahan lebih dini.

2.2.9.4.Pengujian

Bahwa pembuatan unit pengujian sebelum pengkodean dimulai merupakan elemen kunci dari pendekatan pengembangan perangkat lunak cepat XP. Unit pengujian yang harus dibuat dan kemudian dijalankan menggunakan kerangka kerja yang memungkinkan mereka untuk diotomatisasi (sehingga mereka dapat dijalankan dengan mudah dan dapat dijalankan berulang kali). Hal ini mendorong strategi pengujian regresi terhadap kode- kode program setiap kali kode-kode program tersebut diubah (yang seringkali dikaitkan dengan falsafah refactoring XP). Sebagai unit pengujian individu yang diatur dalam "rangkaiian pengujian universal", pengujian integrasi dan validasi atas sistem/perangkat lunak pada umumnya dapat terjadi. Hal ini menyajikan bagi tim perangkat lunak XP suatu indikasi kemajuan dan juga dapat memberikan peringatan awal jika terjadi sesuatu yang keliru. Wells menyatakan: "Memperbaiki masalah-masalah kecil setiap beberapa jam membutuhkan waktu lebih sedikit daripada memperbaiki masalah besar sebelum batas waktunya."

Uji kelayakan XP, sering juga disebut uji pelanggan, dirincioleh para pelanggan dan pada dasarnya berfokus pada fitur- fitur dan fungsionalitas-fungsionalitas sistem/perangkat lunak secara keseluruhan yang dapat terlihat dan ditinjau kembali oleh para pelanggan. Uji kelayakan berasal dari cerita pengguna yang telah diimplementasikan sebagai bagian dari suatu rilis perangkat lunak.

2.3 Penelitian Terdahulu

Dasar atau acuan yang berupa teori-teori atau temuan-temuan melalui hasil berbagai penelitian sebelumnya merupakan hal yang sangat perlu dan dapat dijadikan sebagai data pendukung. Salah satu data pendukung yang menurut peneliti perlu dijadikan bagian tersendiri adalah penelitian terdahulu yang relevan dengan permasalahan yang sedang dibahas dalam penelitian ini. Dalam hal ini, fokus penelitian terdahulu yang dijadikan acuan adalah terkait dengan masalah sistem informasi dan jasa. Untuk memudahkan pemahaman terhadap bagian ini, dapat dilihat pada tabel berikut ini:

Tabel 2. 6 Penelitian Terdahulu

ISSN	JUDUL	HASIL PENELITIAN
2338-0403	Pembuatan Aplikasi Antar-Jemput Laundry Berbasis Web Service pada Platform Android	Aplikasi antar-jemput laundry pada platform Android yang dibuat menggunakan bahasa pemrograman Java dan akses basisdata melalui web service sudah dapat digunakan untuk mendata pakaian yang masuk dan keluar laundry maupun mendata pelanggan yang akan menggunakan jasa laundry secara tepat waktu.
2407-1846	Pembangunan Sistem Informasi Pelayanan Jasa Laundry Berbasis Web Dengan Fitur Mobile Pada 21 Laundry Padang	Dapat menghemat waktu dalam pencarian data transaksi pada saat mengelola data transaksi. Aplikasi mampu menghasilkan <i>output</i> berupa rekap transaksi bulanan, data master yang terbaru, serta penyerahan nota <i>order laundry</i>

2549-0796	Analisa Sistem Informasi Geografis Tempat Laundry Berbasis Android	Sistem Informasi tempat <i>laundry</i> di Kota Tegal berbasis <i>Android</i> , dapat disimpulkan bahwa: Sulitnya mencari informasi tempat <i>laundry</i> secara tepat dan efisien menjadi latar belakang utama dalam proses pembuatan sistem informasi tempat laundry di Kota Tegal. Media promosi bagi pemilik <i>laundry</i> yang masih menggunakan brosur atau <i>banner</i> yang tidak dapat tersebar secara luas dengan cepat.
2087-1224	<i>Qr Code Reader</i> Pada <i>Smartphone</i> Android Pada Layanan Restoran	Dengan menggunakan sistem yang terintegrasi antara aplikasi pada <i>smartphone</i> dengan aplikasi backend proses operasional restoran dalam mengelola data menu, karyawan, dan transaksi menjadi lebih baik dan rapi
2302-3805	Analisis Perancangan Aplikasi Layanan Laundry Berbasis Visual Basic 2010 Dan Sql Server 2000 Pada Gajayana Laundry	Aplikasi laundry sangat membantu para pelaku usaha laundry dalam melakukan proses transaksi dan pencatatan pendapatan yang diperoleh. Penerapan aplikasi laundry dalam pencatatan pendapatan sangat mudah dilakukan dan dapat meminimalisasi kesalahan dalam pencatatan pendapatan

2354-6654	Sistem Informasi Laundry Berbasis Metode OOAD	Sistem informasi ini diterapkan pada Bibi's Laundry untuk memberikan beberapa layanan seperti pemesanan secara online, komunikasi dengan staff secara online, dan beberapa layanan lain yang dapat memberikan kemudahan bagi pelanggan dan pegawai dari laundry tersebut. Sistem informasi ini diterapkan pada Bibi's Laundry untuk memberikan beberapa layanan seperti pemesanan secara online, komunikasi dengan staff secara online, dan beberapa layanan lain yang dapat memberikan kemudahan bagi pelanggan dan pegawai dari laundry tersebut.
2354-6654	Sistem Informasi Laundry Berbasis Metode OOAD	Sistem informasi ini diterapkan pada Bibi's Laundry untuk memberikan beberapa layanan seperti pemesanan secara online, komunikasi dengan staff secara online, dan beberapa layanan lain yang dapat memberikan kemudahan bagi pelanggan dan pegawai dari laundry tersebut
2549-6824	Perancangan Sistem Informasi Pelayanan Pelanggan Pada Citra Laundry Bogor	Sistem Informasi Pelayanan Pelanggan di CITRA LAUNDRY membantu pihak manajemen dalam menangani penyimpanan data yang besar. Data yang besar juga dapat terorganisir dengan baik dalam tabel-tabel yang dihasilkan sehingga dengan mudah dapat diakses

2406-7733	Rancang Bangun Sistem <i>Extreme Programming</i> Sebagai Metodologi Pengembangan Sistem	<p>secara cepat dan baik. Pihak manajemen juga bisa mendapatkan laporan pelayanan bulanan dengan lebih terinci dan rapi.</p> <p><i>Extreme Programming</i> bisa di terapkan pada perusahaan atau organisasi yang mempunyai tim khusus untuk pengembangan suatu sistem. Perusahaan atau organisasi yang ingin menerapkan <i>Extreme Programming</i> diharapkan untuk bisa memahami dan mengerti <i>Extreme Programming practice</i>, dan <i>scrum</i>.</p>
2337-3539	Rancang Bangun Aplikasi Web Informasi Eksekutif Pada Pemerintah Kabupaten XYZ	Aplikasi web informasi eksekutif ini merupakan sebuah aplikasi yang memiliki fasilitas untuk dapat menampilkan tampilan grafik untuk data – data yang dihasilkan dari laporan pertanggung jawaban pada suatu kabupaten
2301-8402	Sistem Informasi Pengarsipan Berbasis Web Di Kantor Jurusan Teknik Elektro Unsrat	<p>Hasil penelitian menunjukkan bahwa manajemen arsip digital mampu memberikan kemudahan dalam proses pencarian arsip serta memberikan kemudahan dalam pengelolaan dan penyimpanan surat masuk dan keluar, selain itu arsip digital juga mampu mengurangi biaya untuk perawatan dan penyimpanan karena arsip sudah didigitalisasi atau disimpan dalam bentuk digital</p>

BAB III

METODE PENELITIAN

3.1. Desain Penelitian

Metode pengembangan sistem yang akan digunakan dalam penelitian ini adalah metode Extreme Programming (XP). Adapun tahapannya antara lain (Keina, 2013): 1) Planning, Aktivitas planning dimulai dengan membentuk *user stories*. Anggota Extreme Programming (XP) team kemudian menilai setiap story dan menentukan cost diukur dalam *development week*. *Customer* dan Extreme Programming (XP) team bekerja bersama untuk memutuskan bagaimana group story untuk release berikutnya (software increment) berikutnya untuk dibangun oleh Extreme Programming (XP) team. Jika komitmen telah dibuat, Extreme Programming (XP) team akan membangun *story-story* dengan cara:

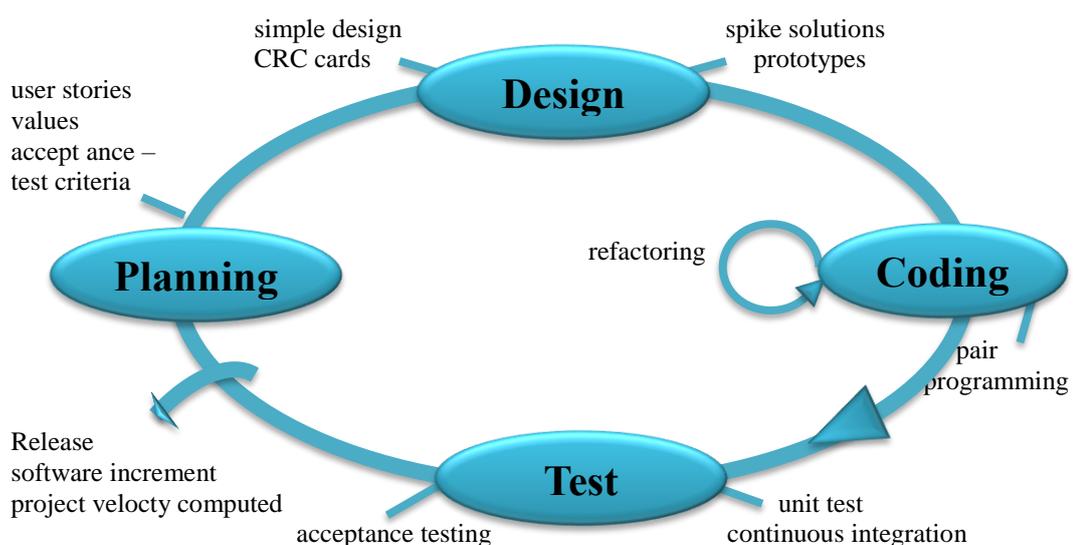
- a. Semua story segera diimplementasikan (dalam beberapa minggu).
- b. Story dengan *value* tertinggi akan dipindahkan dari jadwal dan diimplementasikan pertama.
- c. Story dengan resiko paling tinggi akan diimplemetasikan lebih dulu.

Setelah *project* pertama di-*release* dan di- *delivery*, *Extreme Programming* (XP) team memperhitungkan kecepatan *project*. Selama *development*, *customer* dapat menambah *story*, merubah *value*, membagi *story* atau menghapusnya.

Design. Extreme Programming menggunakan CRC card, untuk mengenali dan mengatur *object oriented class* yang sesuai dengan software increment

Coding. Sebelum membuat *code*, lebih baik membuat unit test tiap *story* untuk dimasukkan dalam *software increment*. *Extreme Programming* (XP) menyarankan agar dua orang bekerja bersama pada satu komputer workstation untuk membuat *code* dari satu *story* (*pair programming*), untuk menyediakan *real time problem solving* dan jaminan *real time quality*. Setelah *pair programming* selesai, *code* diintegrasikan dengan kerja lainnya (*continuous integration*).

Testing. Unit *test* yang telah dibuat harus diimplementasikan menggunakan suatu *framework* dan diatur ke dalam *universal testing suite*, integrasi dan validasi sistem dapat dilakukan setiap hari. *Customer test* (*acceptance test*) dilakukan oleh customer dan fokus pada keseluruhan fitur dan fungsional sistem. *Acceptance test* diperoleh dari *customer stories* yang telah diimplementasikan sebagai bagian dari *software release*.



Gambar 3. 1 Extreme Programming

Untuk memperoleh data yang digunakan dalam penyusunan penelitian ini maka dilakukan pengambilan data secara primer dan sekunder, yaitu:

1. Data primer, Data primer yaitu data atau informasi yang diambil langsung dari subjek penelitian bersumber dari Pondok Pesantren Qodratullah.
2. Data sekunder, Data sekunder yaitu sumber data penelitian yang diambil secara tidak langsung melalui media perantara (diperoleh dan dicatat oleh pihak lain) yaitu dari dokumen dan studi pustaka, baik yang dipublikasikan maupun yang tidak dipublikasikan.

Penelitian ini menggunakan metode penelitian deskriptif. Metode deskriptif adalah suatu bentuk penelitian yang ditujukan untuk mendeskripsikan fenomena-fenomena yang ada, baik fenomena alamiah maupun fenomena buatan manusia. Fenomena itu bisa berupa bentuk, aktivitas, karakteristik, perubahan, hubungan, kesamaan, dan perbedaan antara fenomena yang satu dengan fenomena lainnya. (Sukmadinata, 2006). Maka metode deskriptif sangat tepat jika digunakan untuk mengembangkan perangkat lunak pengembangan e-keuangan berdasarkan data-data yang ada.

3.2. Objek Penelitian

Penulis melakukan sampel yang menjadi objek penelitian dari penulis di Pelangi Futsal yang beralamat di Jl. Dapur 12, Sungai Pelunggut, Sagulung, Kota Batam, Kepulauan Riau 29444. Provinsi: Kepulauan Riau. Telepon: 0813-6465-9799



Gambar 3. 2 Pelangi Futsal

Pelangi futsal merupakan salah satu lapangan olahraga yang memiliki dua lapangan utama yang sering digunakan dalam bermain futsal. Lapangan Pelangi Futsal ini sendiri berlokasi di Jl. Dapur 12, Sungai Pelunggut, Sagulung, Kota Batam, Kepulauan Riau 29444. Provinsi: Kepulauan Riau. Dengan lokasi strategis yang dimiliki Pelangi Futsal ini maka dapat ditarik kesimpulan bahwa pencari lapangan futsal sangat tertarik sekali ingin mendapatkan tempat bermain di pelangi futsal ini. Sehingga penulis melakukan pengambilan sampel penelitian di Pelangi futsal dapur 12.

3.3. Analisa SWOT Program yang berjalan

Analisa SWOT adalah metode perencanaan strategis yang digunakan untuk mengevaluasi kekuatan atau *strength*, kelemahan atau *Weaknes*, peluang atau *Opportunities*, dan ancaman atau *Threat* dalam suatu proyek atau suatu spekulasi

bisnis. Hal ini dimaksudkan agar strategi yang akan diambil memiliki dasar fakta yang dapat dipertanggung jawabkan. Melakukan analisa SWOT peneliti dapat melihat permasalahan atau kekurangan yang tertadapat pada objek.

Adapun analisa SWOT untuk sistem yang sedang berjalan adalah sebagai berikut:

1. Kekuatan Sistem (*Strenght*)

- a. Proses Penyewaan lapangan futsal lebih muda dengan tersedianya layanan akses melalui media yang disediakan utamanya media web
- b. Biaya yang dikeluarkan relative lebih murah karena hanya diakses melalui *gadget* ataupun media computer yang terhubung dengan internet.
- c. Pemilik lapangan futsal lebih mudah dalam melakukan transaksi dan pemasaran lapangan yang dimiliki.

2. Kelemahan Sistem (*Weakness*)

- a. Informasi yang diberikan masih rentan dengan kesalahan, karena pengimputaan data lapangan futsal dilakukan masih dengan cara manual.
- b. Pengguna lapangan membutuhkan waktu lama untuk mendapatkan informasi lapangan futsal yang tersedia.
- c. Membutuhkan banyak informasi pemilik usaha lapangan futsal sebagai member untuk menunjang lancarnya aliran sistem informasi.

3. Peluang Sistem (*Oportunity*)

- a. Suatu sistem harus mampu menyampaikan informasi yang akurat bagi pengguna dan mampu menghasilkan data yang valid.

- b. Pencarian penyewaan lapangan futsal secara cepat dan professional dapat dilakukan dengan sistem informasi Penyewaan lapangan futsal berbasis web.
- c. Sistem informasi penyewaan lapangan futsal berbasis web yang secara terkomputerisasi akan mempermudah dalam pencarian lapangan futsal yang sesuai dan baik untuk para pengguna dan media bagi pemilik lapangan futsal untuk mempromosikan lapanganfutsal itu sendiri.

4. Ancaman (*Threat*)

Muncul sistem baru yang lebih efisien untuk menggantikan sistem annual yang saat ini sedang berjalan

4.1. Analisa sistem yang sedang berjalan

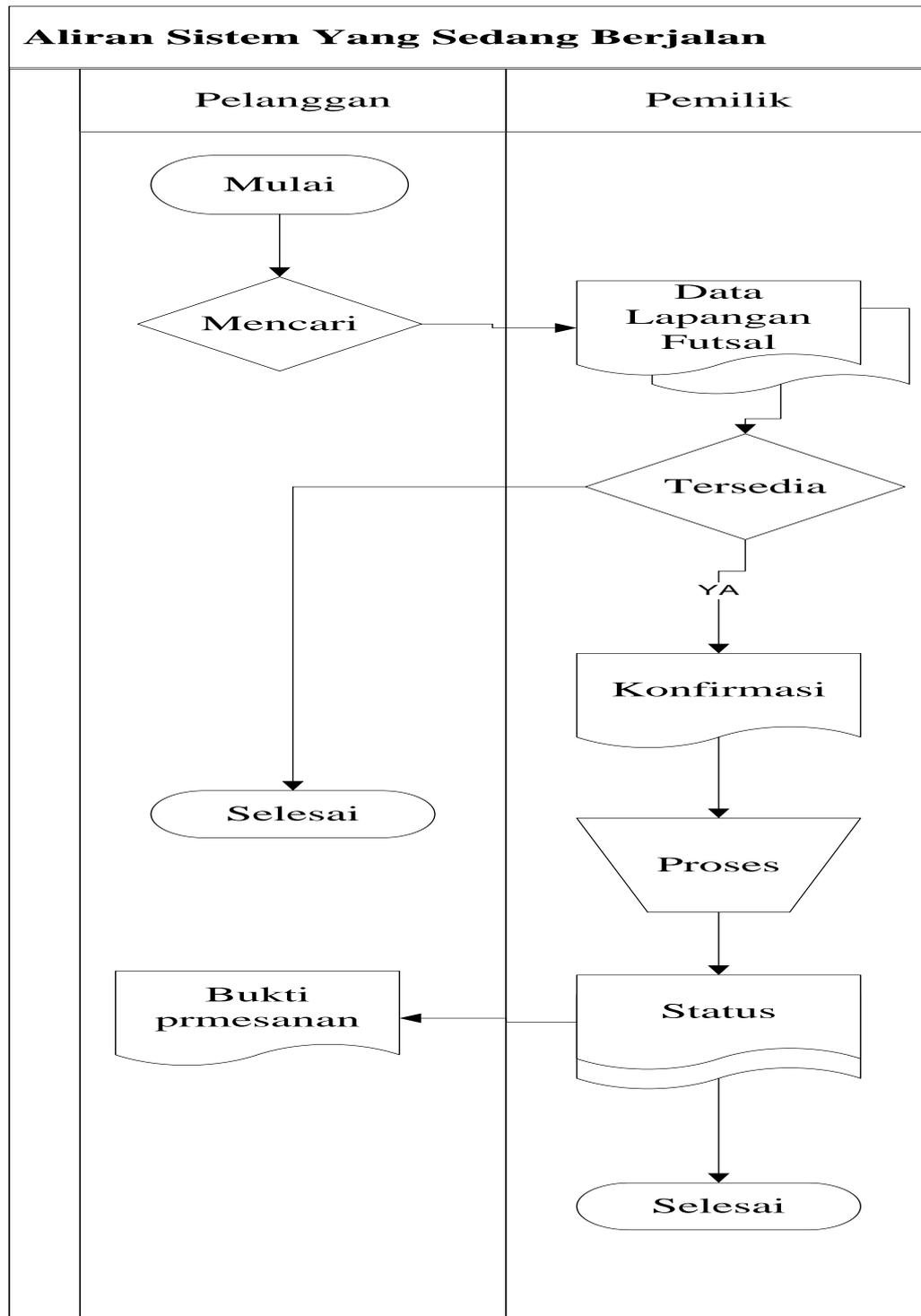
Sistem pencarian dan penyewaan lapangan futsal yang sedang berjalan saat ini masih terdapat kekurangan di era digital seperti saat ini. Sistem penyewaan lapangan futsal saat ini masih menggunakan cara yang lama dalam sistem penyewaan atau booking lapangan futsal, sehingga menimbulkan hilangnya biaya dan waktu dan kurang efektif dalam pencarian dalam penyewaan lapangan futsal, system informasi yang sedang berjalan saat ini masih menggunakan cara manual yaitu masih mencari penyewaaan lapangan futsal bermodalkan informasi dari teman sekitar masyarakat dan mencari lokasi ke lokasi lain tanpa bebekal informasi penyewaan lapangan futsal.

Kurangnya media untuk pencarian sistem penyewaan lapangan futsal itu sendiri berdampak pada orang yang membutuhkan atau masyarakat yang ingin

menyewa lapangan futsal. Penyewaan lapangan futsal yang ada pada sistem yang sedang berlangsung adalah orang-orang hanya berbekal informasi yang diperoleh dari teman atau masyarakat sekitar yang menyatakan bahwa disatu tempat ada lapangan futsal yang disewakan, dengancara seperti ini maka tidak jarang pemesan atau pengguna merasa kecewa karena belum tentu data yang didapat secara langsung sesuai dengan kenyataan yang terjadi, dan yang paling sering terjadi dalam penyewaan lapangan futsal adalah masalah biaya yang sering berubah sewaktu-waktu, ini menyebabkan para pengguna atau pemesan lapangan futsal merasa kecewa dan meras rugi dalam biaya dan waktu.

Pada aliran sistem yang akan digambarkan disini pada aliran sistem informasi yang sedang berlangsung, disini mengikutsertakan pemilik lapangan futsal atau penanggung jawab lapangan futsal karena pada aliran yang terjadi sering kali pemilik lapangan futsal tidak mengetahui harga yang diterima oleh pengguna karena kurangnya media untuk menentukan tarif yang sesuai yang dapat dilihat atau diakses dengan cepat. Fungsi pemilik lapangan futsal disini tidak bias dilepaskan dari penyewaan lapangan futsal, sehingga proses aliran sistem informasi yang sedang berjalan dapat berjalan dengan baik.

4.2. Aliran Sistem Informasi yang Sedang Berjalan



Aliran sistem informasi yang terjadi diatas adalah rangkaian kegiatan secara umum para aktor yang terjadi dalam proses penyewaan lapangan futsal hingga proses yang terjadi bagaimana bila pengguna tidak menemukan lapangan yang diinginkan . Pada gambar aliran sistem informasi yang sedang berjalan dapat dijelaskan bahwa:

1. Pada saat mencari lapangan futsal untuk disewakan maka pengguna harus mencari lapangan untuk bisa disewakan atau yang akan dipakai.
2. Pada saat lapangan sudah ditemukan maka pengguna akan menghubungi atau menemui pemilik lapangan untuk konfirmasi ketersediaan lapangan.
3. Dalam menentukan lapangan tidak semua pemesan atau pengguna lapangan futsala setuju mengenai tariff atau biaya yang akan dibayar maupun jam yang tersedia
4. Pada saat pencari lapangan tidak setuju dengan ketersediaan waktu maupun biaya maka pengguna akan mencari lapangan lain untuk digunakan hal ini akan memakan waktu yang akan terbuang.
5. Pada saat pengguna atau penyewa lapang futsal sepakat atau setuju dengan tarif yang ditawarkan atau dengan jam waktu lapangan yang dipakai maka pengguna akan melakukan konfirmasi pembayaran mengenai biaya yang akan dikeluarkan oleh penyewa lapangan.
6. Setelah proses itu selesai maka pengguna atau penyewa lapangan akan bias melakukan kegiatan dilapangan yang disewakan sesuai aturan yang berlaku yang ditetapkan oleh pemilik lapangan futsal.

4.3. Permasalahanyang sedang dihadapi

Dengan menggunakan sistem yang sedang berlangsung atau yang sedang berjalan pada penyewaan lapangan futsal, ada beberapa hal yang menjadi permasalahan yang sedang dihadapi diantaranya adalah sebagai berikut:

1. Belum tersedianya sistem informasi yang mempermudah para pencari lapangan futsal untuk yang dapat disewakan atau kurangnya media bagi pengguna untuk mempermudah memperoleh informasi lapangan futsal.
2. Banyak pencari lapangan futsal sehingga sulitnya mendapat informasi ketersediaan lapangan futsal untuk kebutuhan mereka.
3. Waktu dan biaya yang diperlukan untuk menemukan lapangan futsal akan semakin tinggi dengan sistem yang sedang berlangsung.
4. Kesulitannya memesan lapangan futsal dengan kesibukan yang begitu tinggi.
5. Kesulitan pemilik usaha lapangan futsal untuk mempromosikan lapangan futsal yang mereka miliki.

4.4. Usulan Pemecahan Masalah

Sebagaimana yang telah dijelaskan diatas, dari permasalahan yang sedang dihadapi dapat ditarik kesimpulan bahwa permasalahan pokok yang dihadapi adalah penyewaan lapangan futsal masih minim informasi yang diperoleh oleh pengguna menyangkut penyewaan lapangan futsal.

Untuk itu penulis memberikan ide atau masukan untuk memmbuatkan sebuah sistem informasi yang mampu menyediakan dan memberikan informasi penyewaan

lapangan futsal kepada pengguna yang dapat diakses dengan mudah dan mempermudah layanan pemilik usaha lapangan futsal dan tidak akan memakan banyak biaya dan waktu begitu juga dengan dengan pemilik lapangan futsal mempermudah dalam mempromosikan lapangan futsal