

BAB II

TEORI ANALISIS DAN PERANCANGAN SISTEM INFORMASI PENJUALAN

2.1 Sistem Informasi Akuntansi

Sistem informasi akuntansi dapat dipahami melalui penggalan kata, penggalan kata tersebut adalah sistem, informasi dan akuntansi. Sistem dapat didefinisikan sebagai sebuah proses untuk menghasilkan sesuatu. Bermula pada *input*, kemudian diproses sehingga menghasilkan *output* (Considine, Parkes, Olesen, Blount, & Speer, 2012). Sistem dirancang dengan tujuan untuk menyelesaikan permasalahan. Dalam rangka mencapai tujuannya, sistem secara keseluruhan akan dibantu oleh subsistem.

Informasi merupakan data yang sudah diproses sedemikian rupa sehingga memiliki sebuah arti (Romney, Steinbart, Summers, & Wood, 2021). Data sebagai data mentah tidak memiliki arti. Data sendiri merupakan kumpulan fakta-fakta yang berkaitan pada sebuah kejadian (*events*). Perusahaan akan mengubah data tersebut menjadi informasi agar memiliki arti. Proses transformasi tersebut dapat dilakukan oleh sistem informasi.

Dalam proses transformasi informasi, data akan dikumpulkan terlebih dahulu. Data yang dikumpulkan kemudian diproses. Sedangkan data yang sudah diproses, harus direkam dan disimpan. Hal ini disebabkan informasi memiliki nilai tambah bagi perusahaan. Informasi menjadi dasar dari pengambilan keputusan.

Akuntansi dikenal sebagai rangkaian proses perekaman seluruh transaksi keuangan milik perusahaan. Transaksi keuangan akan disusun secara sistematis dan komprehensif. Proses perekaman meliputi kegiatan penyatuan, menganalisa dan melaporkan seluruh aktivitas keuangan. Hasil rekam keuangan perusahaan akan disajikan kepada pihak-pihak tertentu. Pihak tersebut merupakan pihak yang memiliki kepentingan terhadap perusahaan. Contohnya adalah pihak manajemen, investor, pihak eksternal dan perpajakan.

Secara keseluruhan, sistem informasi akuntansi merupakan sistem yang bertanggung-jawab dalam merekam, menyimpan dan memproses data sehingga menghasilkan informasi bagi pembuat keputusan (Romney, Steinbart, Summers, & Wood, 2021). Sistem tersebut meliputi manusia, prosedur, data, *software*, infrastruktur teknologi dan kontrol terhadap internal. Sistem informasi akuntansi dapat berbentuk sistem pengerjaan manual tulis tangan atau berbentuk sistem terkomputerisasi. Namun, sistem informasi akuntansi dapat menerapkan kedua bentuk sekaligus.

Sistem informasi akuntansi harus dapat melakukan fungsi pengumpulan data, pemrosesan data menjadi informasi, penyimpanan dan pelaporan informasi. Terlepas dari bentuk sistem yang dipergunakan. Tujuan utama dari sistem informasi adalah untuk menghasilkan informasi. Dalam menghasilkan informasi, sistem informasi akuntansi terdiri dari enam komponen (Romney & Steinbart, 2018). Keenam komponen tersebut adalah sebagai berikut :

- 1) Manusia yang menggunakan sistem.
- 2) Prosedur dan Instruksi yang digunakan dalam pengumpulan, pemrosesan dan penyimpanan data.
- 3) Data yang berkaitan dengan perusahaan dan aktivitasnya.
- 4) *Software*.
- 5) Infrastruktur teknologi informasi yang dipergunakan pada sistem informasi akuntansi.
- 6) Kontrol internal dan keamanan data.

Penerapan sistem informasi akuntansi, menambahkan nilai perusahaan dibanding perusahaan pesaing. Sistem informasi akuntansi yang didesain secara baik menambah nilai sebuah perusahaan. Berikut adalah beberapa nilai tambah yang dimiliki perusahaan apabila menerapkan sistem informasi akuntansi dengan baik:

- 1) Mengurangi biaya yang dibebankan kepada produk atau jasa.
- 2) Meningkatkan efisiensi kinerja perusahaan.
- 3) *Sharing knowledge*.
- 4) Meningkatkan kontrol terhadap internal.

- 5) Meningkatkan kualitas pengambilan keputusan.

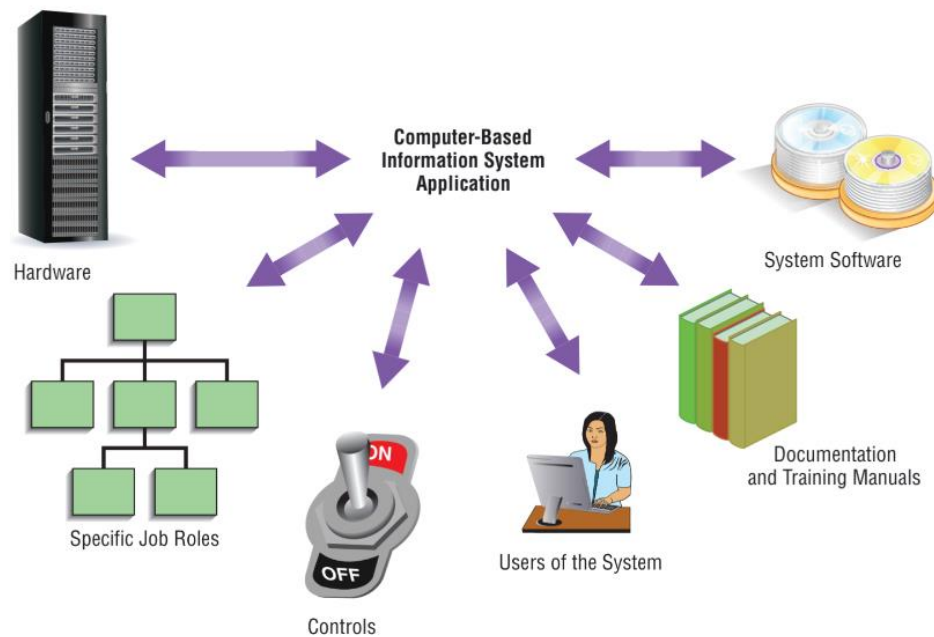
2.2 Analisis dan Perancangan Sistem Informasi

Analisis dan perancangan sistem informasi dapat diartikan sebagai proses mengembangkan dan memelihara sebuah sistem informasi (Valacich, George, & Hoffer, 2015). Hal ini dilakukan, agar sistem informasi tetap dapat menjalankan fungsi-fungsi dasarnya. Fungsi dasar sistem informasi yang berkaitan dengan proses bisnis adalah melacak dan menyimpan nama *customer* serta menyimpan alamat *customer*. Fungsi lain-nya adalah memproses pesanan penjualan, memeriksa stok persediaan dan membayar gaji karyawan.

Analisis dan perancangan sistem informasi dilakukan oleh analis sistem. Dengan melakukan analisis terhadap data (*input*) dan proses transformasi data menjadi informasi (*ouput*). Analis sistem berusaha untuk mengidentifikasi masalah yang muncul pada sistem dan menemukan solusi yang tepat (Kendall & Kendall, 2014). Proses identifikasi masalah dan penemuan solusi dapat dibantu dengan menerapkan sistem informasi yang sudah terkomputerisasi.

Sistem informasi didesain agar dapat menunjang fungsi-fungsi tertentu atau proses yang dijalankan pada perusahaan. Tujuan utama dari sistem informasi adalah untuk menghasilkan informasi. Pada pelaksanaannya sistem informasi harus memiliki komponen-komponen agar dapat beroperasi (Valacich, George, & Hoffer, 2015), selain komponen *software* terdapat komponen lain, yakni :

- 1) *Hardware* atau perangkat keras yang menjalankan *software*.
- 2) Dokumentasi dan materi pelatihan yang berkaitan dengan sistem informasi.
- 3) *Specific job roles*.
- 4) Kontrol terhadap keamanan data, dari pencurian dan penipuan.
- 5) *User* yang menggunakan *software*.



(Sumber : Valacich, George, & Hoffer, 2015)

Gambar 2.1

Komponen Sistem Informasi

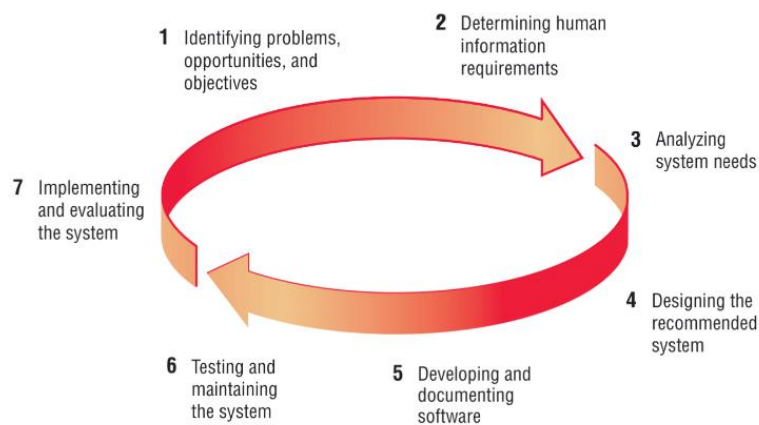
Tujuan utama dari pelaksanaan analisis dan perancangan sistem informasi adalah meningkatkan sistem informasi yang digunakan oleh perusahaan. Sistem informasi memiliki peran penting dalam perusahaan, khususnya perusahaan bisnis. Proses bisnis perusahaan sangat bergantung pada sistem informasi. Sistem informasi berfungsi sebagai alat yang digunakan dalam mengolah data menjadi informasi.

2.2.1 Metodologi Analisis dan Perancangan Sistem Informasi

Dalam melakukan analisis dan perancangan sistem informasi (disingkat menjadi APSI), terdapat beragam metodologi. Berdasarkan pemahaman peneliti, terdapat empat metodologi yang diketahui. Metodologi tersebut adalah *SDLC*, *Prototyping*, *RAD* dan *Agile*. Masing-masing metodologi memiliki kelebihan beserta kekurangannya. Peneliti bertujuan untuk menentukan metodologi yang cocok untuk digunakan dalam penelitian ini.

2.2.1.1 Metodologi *Systems Development Life Cycle (SDLC)*

Dimulai dengan metodologi *systems development life cycle* (disingkat menjadi *SDLC*), merupakan metodologi yang umumnya digunakan oleh banyaknya perusahaan. Merupakan salah satu metodologi yang sering dipergunakan dalam APSI. Metodologi ini memiliki tahap-tahap perancangan atau fase. Tidak ada ketentuan pasti, berapa banyak jumlah fase yang ada.



(Sumber : Kendall & Kendall, 2014)

Gambar 2.2
Tujuh Tahap *SDLC*

Menurut (Kendall & Kendall, 2014), terdapat tujuh tahapan dalam metodologi *SDLC*. Ketujuh fase atau tahapan yang terdapat pada *SDLC* adalah sebagai berikut :

- 1) Identifikasi permasalahan, peluang dan tujuan.
- 2) Menentukan kebutuhan informasi.
- 3) Analisis kebutuhan Sistem
- 4) Desain sistem yang direkomendasikan
- 5) Merancang dan mendokumentasikan *software*.
- 6) Uji coba sistem.
- 7) Implementasi dan evaluasi sistem.

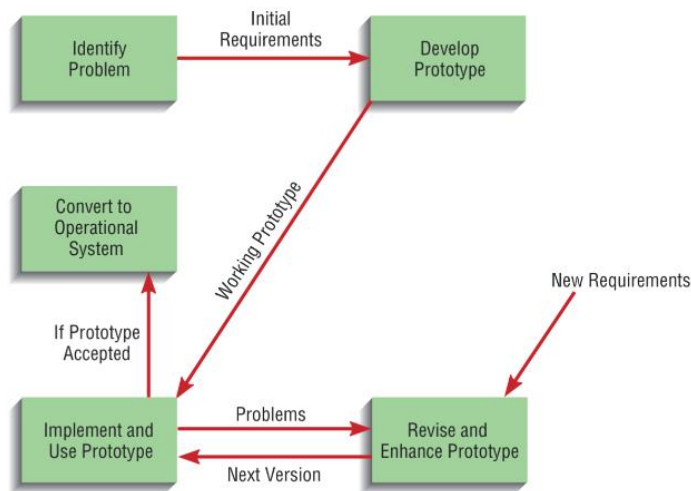
Tabel 2.1
Kelebihan dan Kekurangan Metodologi *SDLC*

No.	Kelebihan Metodologi <i>SDLC</i>	Kekurangan Metodologi <i>SDLC</i>
1.	Digunakan dalam proyek dengan skala besar dan kompleks.	Memakan waktu dan biaya lebih banyak, mengingat fase yang dilewati lebih banyak dibanding metode lain.
2.	Dokumentasi yang dihasilkan di tiap fase yang sudah selesai, mempermudah untuk melakukan maintenance.	Metodologi <i>SDLC</i> , memiliki tingkat kompleksitas tinggi dibanding metodologi lainnya.
3.	-	Metodologi <i>SDLC</i> , memiliki potensi untuk sulit diubah, apabila terjadi perubahan kebutuhan.

(Sumber : Kendall & Kendall, 2014)

2.2.1.2 Metodologi *Prototyping*

Metodologi selanjutnya adalah *Prototyping*. Teknik pengembangan sistem informasi ini berfokus pada pembuatan *prototype* dari sistem informasi yang diinginkan. *Prototype* dirancang sebagai bentuk dasar dari sistem informasi. Pembuatan *prototype* dilakukan agar perusahaan dapat melihat dan mencoba sistem informasi yang akan dirancang (Kendall & Kendall, 2014). Tujuan pembuatan *prototype* adalah membantu analisis sistem dalam merancang sistem informasi yang sesuai dengan kebutuhan dan keinginan perusahaan.



(Sumber : Valacich, George, & Hoffer, 2015)

Gambar 2.3
Metodologi *Prototyping*

Metodologi *prototyping* digambarkan pada gambar 2.3, pada gambar tersebut menggambarkan cara atau langkah dalam melakukan metode ini. Langkah-langkah pada metodologi *prototyping* adalah sebagai berikut :

1) Identifikasi permasalahan

Pertama analisis sistem harus menentukan kebutuhan terhadap sistem terlebih dahulu. Hal ini dapat dilakukan apabila analisis sistem dapat menemukan sumber masalah yang ada. Menemukan sumber masalah dapat membantu analisis, dalam menentukan bentuk sistem informasi yang diinginkan oleh perusahaan.

2) Merancang *prototype*

Setelah mengetahui kebutuhan *users*, maka *prototype* dapat langsung dirancang. *Prototype* merupakan gambaran secara umum, mengenai tampilan sistem nantinya. Tidak hanya tampilan tatap muka dari sistem, namun dapat berupa beberapa fungsi dasar sistem informasi. *Prototype* bukanlah sistem yang utuh, memiliki potensi

untuk dikembangkan menjadi sistem utuh. Agar *prototype* dapat berfungsi sesuai dengan tujuan, maka peran *users* dalam memberikan *feedback* menjadi penting. *Users* diharapkan memberikan tanggapan terhadap *prototype*, sehingga jika tidak sesuai dapat diperbaiki.

3) Implementasi dan uji coba *prototype*

Pada tahap ini, rancangan *prototype* akan diimplementasikan. *Users* akan melakukan uji coba terhadap *prototype*. Pada saat uji coba, *prototype* berpotensi untuk diterima atau tidak diterima. Jika diterima, maka rancangan tersebut akan dikonversi menjadi sistem yang beroperasi. Sedangkan, jika tidak diterima maka sistem akan direvisi untuk ditingkatkan hingga memenuhi kebutuhan.

4) Konversi menjadi sistem operasional

Prototype yang sudah dirancang akan dikonversi menjadi sistem operasional. Hal ini dapat terjadi apabila *users* sudah menerima rancangan tersebut. Rancangan tersebut sudah dapat mengakomodasi kebutuhan. Sistem operasional dan dapat digunakan, hingga muncul kebutuhan baru. Apabila kebutuhan baru muncul maka proses akan pembaharuan dimulai kembali.

5) Revisi dan peningkatan *prototype*

Pada tahap ini, *prototype* akan dinilai apakah dapat memenuhi kebutuhan oleh *users*. Apabila *users* menilai tidak dapat memenuhi kebutuhannya, maka *prototype* ditolak. Hal ini berarti perlu perbaikan dan peningkatan pada *prototype*. Setelah diperbaiki maka *prototype* akan diuji kembali, hingga dapat memenuhi kebutuhan.

Prototyping sendiri memiliki keunggulan dan kelemahan dalam merancang sistem informasi. Keunggulan yang dimiliki *prototyping* menurut (Kendall & Kendall, 2014), adalah sebagai berikut :

- 1) Potensi *prototype* yang dirancang, mendekati sistem informasi yang memenuhi kebutuhan dan ekspektasi pengguna (*users*).

- 2) *Prototype* yang dinilai tidak sesuai dengan kebutuhan *users*, maka dapat langsung diperbaiki ulang agar memenuhi kebutuhan *users*. Sehingga proses pengembangan tidak memakan waktu yang lama.
- 3) Penggunaan metodologi *prototyping* memiliki potensi untuk dirubah pada awal tahap APSI, apabila dinilai tidak sesuai.

Sedangkan untuk kelemahan dari *prototyping* menurut (Kendall & Kendall, 2014) adalah sebagai berikut :

- 1) Timbulnya permasalahan kesulitan dalam mengelola *prototyping*, apabila proyek berskala besar.
- 2) *Prototype* yang sudah dibuat, pada kenyataannya belum tentu siap digunakan.

2.2.1.3 Metodologi *Rapid Application Development (RAD)*

Merupakan metodologi yang ditunjukkan untuk mengembangkan sistem informasi secara lebih cepat dan lebih berkualitas dibandingkan metodologi tradisional. Metode *RAD* hadir untuk menjawab permasalahan pada model waterfall. Pendekatan *RAD* dalam mengembangkan sistem informasi lebih menekankan pada proses adaptasi dibanding perencanaan.

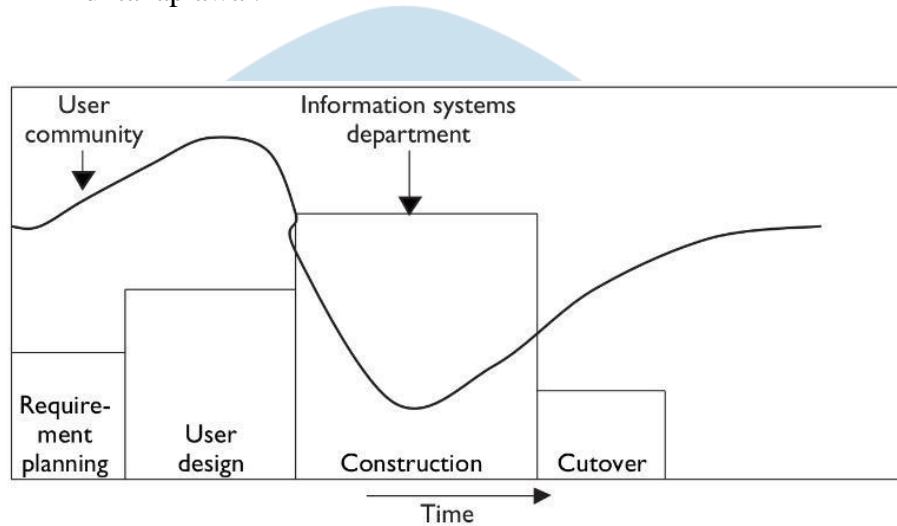
Menurut McLeod, Jr. & Jordan (2002), Metodologi *RAD* memiliki empat fase atau tahapan. Fase atau tahapan tersebut meliputi sebagai berikut :

- 1) *Requirement planning*
Pada tahap ini *users* bekerja sama dengan *developers* agar dapat mengidentifikasi permasalahan bisnis.
- 2) *User design*
Pada tahap ini sistem yang dibuat haruslah didasarkan pada kebutuhan. Saat proses perancangan, *users* akan dibantu oleh *developer*.
- 3) *Construction*
Developers menggunakan *software tools* untuk merancang *prototype*. Hasil rancangan akan di-*review* oleh *users*. *Review* akan dijadikan

dasar perbaikan *prototype*. Proses ini dapat diulang hingga menghasilkan sistem yang dapat diterima oleh *users*.

4) *Cutover*

Sistem yang baru akan diimplementasikan sesuai dengan perencanaan di tahap awal.



(Sumber : McLeod,Jr. & Jordan, 2002)

Gambar 2.4

RAD Life Cycle

Tabel 2.2

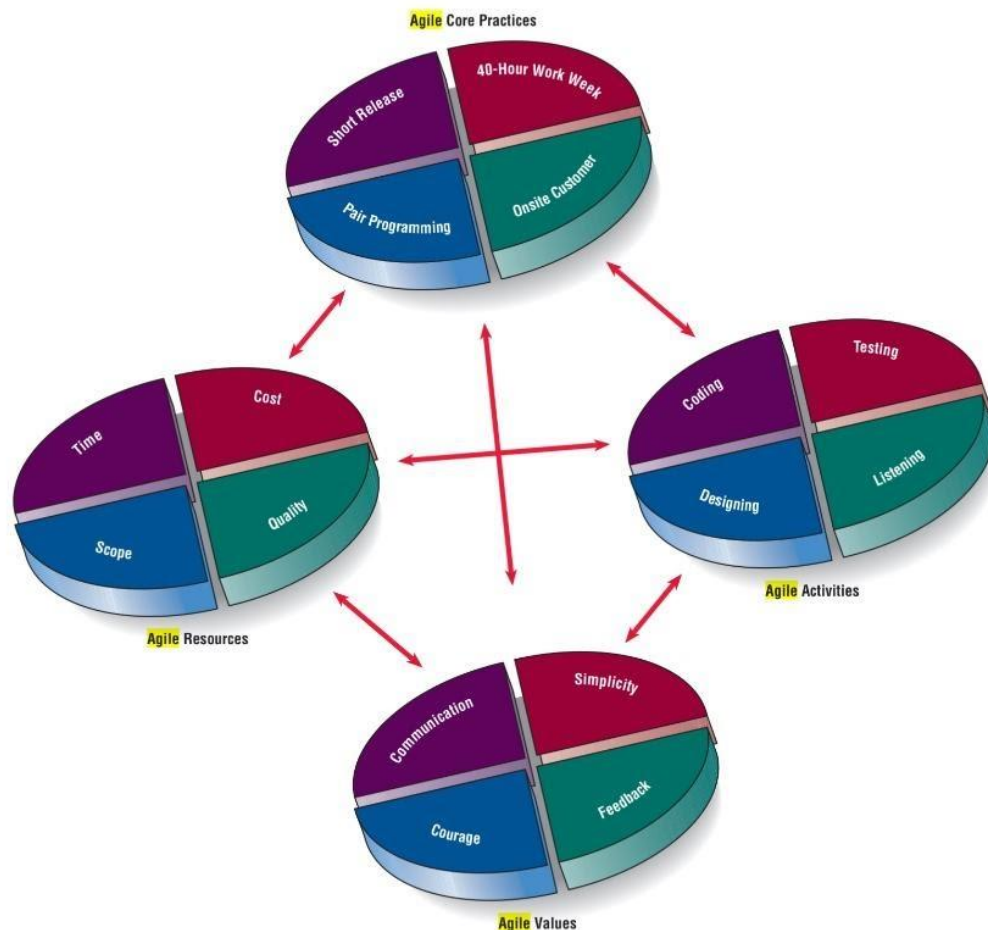
Kelebihan dan Kekurangan Metodologi *RAD*

Kelebihan Metodologi <i>RAD</i>	Kelemahan Metodologi <i>RAD</i>
1) Kualitas <i>software</i> yang lebih baik dibandingkan dengan <i>software</i> hasil pengembangan metode tradisional (<i>waterfall</i>).	1) Desain sistem yang buruk, hal ini dapat disebabkan karena perubahan yang dilakukan oleh <i>developers</i> tanpa memerhatikan arsitektur dari sistem.
2) <i>On-time</i> dan sesuai <i>budget</i> .	2) Tidak semua proyek, sesuai menggunakan metodologi <i>RAD</i> .

(Sumber : McLeod,Jr & Jordan, 2002)

2.2.1.4 Metodologi Agile, *Extreme Programming (XP) Approach*

Metodologi *agile* dengan pendekatan *extreme programming* (selanjutnya disebut *XP*), merupakan metodologi yang berfokus pada inovasi dan pengguna dalam mengembangkan sistem.



(Sumber : Kendall & Kendall, 2014)

Gambar 2.5
Metodologi Agile

Metodologi ini didasarkan pada empat poin utama yakni, *values*, *resources*, *principles*, *practices* dan *activities*. Pembahasan pertama terkait dengan nilai-nilai (*values*). Dalam melakukan pengembangan sistem, pengembang sistem (*developers*) akan berpedoman pada nilai-nilai (dijelaskan pada tabel 2.3).

Tabel 2.3

Values dalam Metodologi Agile

<i>Values</i>	Penjelasan
1) <i>Communication</i>	Agar sistem informasi dapat dikembangkan dalam waktu singkat, maka komunikasi antara pengguna dengan pengembang menjadi kunci.
2) <i>Simplicity</i>	Nilai kedua adalah <i>simplicity</i> , yang berarti pada saat memulai proyek. Mengerjakan pekerjaan yang dapat dikerjakan, meskipun pekerjaan tersebut sederhana.
3) <i>Feedback</i>	Nilai ketiga adalah <i>feedback</i> atau umpan balik. Dalam pendekatan <i>XP</i> , <i>feedback</i> menjadi penting. Perbaikan sistem akan didasarkan pada <i>feedback</i> pengguna. <i>Feedback</i> pada pendekatan <i>XP</i> , umumnya dibutuhkan secara cepat.
4) <i>Courage</i>	Nilai keempat berkaitan dengan tingkat kepercayaan dan kenyamanan di dalam tim proyek. <i>Courage</i> atau keberanian, memiliki arti harus berani memberikan <i>feedback</i> terkait dengan proses pengembangan.

(Sumber : Schwaber & Sutherland, 2020)

Poin kedua adalah *resources* atau sumber daya. *Resources* merupakan salah satu nilai praktik yang harus diperhatikan. *Resources* yang dimaksud dalam metodologi *agile* berkaitan dengan waktu, biaya, kualitas dan cakupan dari proyek. Poin pembahasan selanjutnya adalah prinsip.

Prinsip atau dapat disebut sebagai *principles*. Prinsip atau *principles* merupakan pedoman dalam proses pengembangan. Prinsip-prinsip yang dipegang adalah sebagai berikut :

- Memuaskan *customer* dengan menerapkan *software*.
- Terbuka akan perubahan.
- Mendorong *users* dan *developers* untuk bekerja sama.
- Mempersiapkan umpan balik yang cepat.

- Mengadopsi prinsip *simplicity*.

Pembahasan selanjutnya adalah mengenai *practices*. Terdapat empat *core practices*, yang menjadikan pendekatan ini berbeda dengan perbedaan lainnya.

Keempat *core practices* tersebut adalah sebagai berikut :

- Perilisan *software* dalam jangka waktu yang pendek.
- *Fourty-hour workweek*, berarti tim pengembang harus bekerja secara intens dalam waktu 40 jam per minggu.
- *On site customer*, merupakan pengguna yang paham terkait pengembangan sistem, harus terdapat di lokasi selama proses pengembangan terjadi.
- *Pair programming*.

Selanjutnya adalah *activities*, dalam pendekatan ini terdapat 4 kegiatan utama. Kegiatan-kegiatan tersebut adalah :

- *Coding*, berarti harus membuat kode yang kemudian akan dites. Hal ini diperlukan agar dapat melihat apakah program dapat berjalan atau tidak.
- *Testing*, merupakan aktivitas dasar yang harus dilakukan sepanjang proyek berjalan. *Testing* atau pengetesan untuk memeriksa coding, fungsi dan performa software yang dirancang.
- *Listening*, merupakan aktivitas dasar yang harus dilakukan oleh *developers*. Agar *software* yang dirancang dapat sesuai, maka *developers* harus mendengarkan apa yang menjadi kebutuhan dari *users*.
- *Designing*, merupakan aktivitas dasar terakhir dimana *developers* mendesain struktur sistem. Desain yang dibuat harus dapat bersifat fleksibel, dalam artian jika terdapat perubahan selama proses pengembangan maka dapat diubah.

Proses pengembangan sistem dengan pendekatan *XP*, umumnya meliputi hal berikut :

- Mendengarkan pengalaman *users*.
- Membuat *logical workflow model*, agar dapat memahami proses bisnis.
- Membuat narasi baru berdasarkan *logical workflow model*.
- Membuat *prototype*, agar *users* dapat melihat tampilan antarmuka yang akan dirancang.
- Didasari oleh *feedback* dari *prototype*, maka perancangan sistem akan dilakukan hingga menghasilkan *physical data model*.

2.2.1.5 Metodologi Agile, Scrum Approach

Menurut Schwaber & Sutherland (2020), *Scrum* merupakan sebuah kerangka yang memudahkan *people*, *teams* dan *organizations* dalam menghasilkan solusi yang adaptif untuk masalah yang kompleks. Pendekatan *scrum* didasarkan pada *empiricism* dan *lean thinking*. *Empiricism* berarti ilmu yang diperoleh didasarkan pada pengalaman dan mengambil keputusan berdasarkan apa yang dilihat. Sedangkan *lean thinking* berfokus pada hal-hal penting berkaitan dengan proyek.

Pendekatan *scrum* berfokus pada transparansi, inspeksi dan adaptasi. Ketiga hal tersebut saling berkaitan. Transparansi mengharuskan pekerjaan harus terbuka, baik bagi tim maupun di luar tim. Tanpa adanya transparansi maka inspeksi dapat menyesatkan. Inspeksi dapat berarti pemeriksaan.

Inspeksi dilakukan terhadap proses yang berjalan, agar memastikan dapat mencapai tujuan yang sama. Selanjutnya inspeksi memungkinkan untuk hadirnya adaptasi. Apabila dalam pelaksanaan inspeksi, menemukan adanya perubahan maka tim harus dapat beradaptasi. Tim harus beradaptasi terhadap perubahan yang terjadi.

Pendekatan *scrum* memiliki nilai-nilai yang ditekankan pada saat proyek dilaksanakan. Terdapat lima nilai yang membuat pendekatan *scrum* dikatakan sukses atau bernilai. Kelima nilai tersebut adalah sebagai berikut :

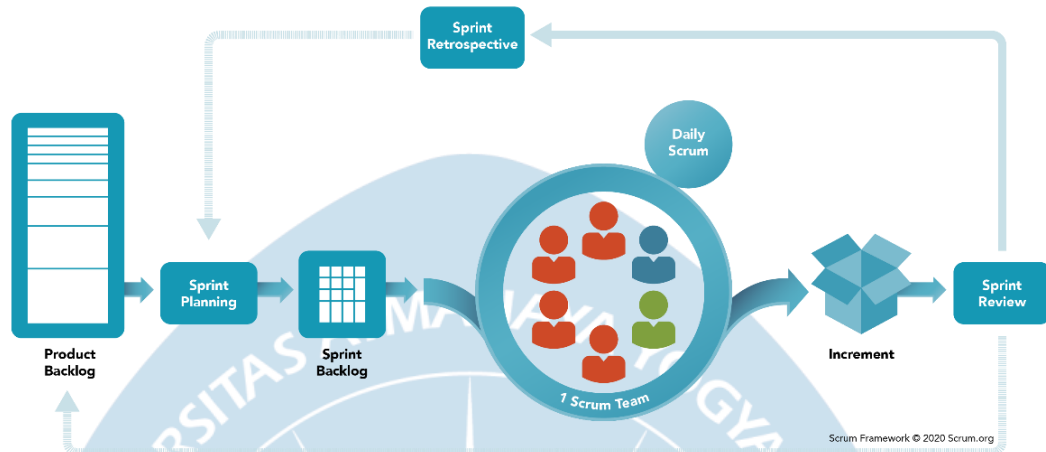
- Komitmen
- Fokus
- Keterbukaan
- *Respect*
- *Courage*

Dalam menjalankan proyek maka diperlukan tim. Pendekatan *scrum* mengharuskan adanya *scrum team*. Tim tersebut terdiri dari tiga komponen, komponen tersebut adalah :

- *Scrum Master*, merupakan pemimpin dari proyek. Bertugas untuk memimpin pelaksanaan dari *scrum*.
- *Product Owner*, merupakan individu yang merepresentasikan kebutuhan *stakeholders* pada *Product Backlog*.
- *Developers*, merupakan kumpulan individu yang memiliki komitmen untuk menciptakan *Increment* pada setiap *Sprint*.

Selanjutnya untuk memahami lebih jauh mengenai pendekatan *scrum*. Dalam pendekatan *scrum*, terdapat istilah *scrum artifacts*. *Scrum artifacts* merepresentasikan nilai yang di desain untuk memaksimalkan transparansi. Setiap *artifacts* mengandung komitmen, untuk memastikan informasi yang dapat meningkatkan transparansi dan fokus pada tiap *progress*. *Progress* diukur dengan :

- *Product Backlog* → *Product Goal*.
- *Product Backlog* merupakan list apa saja yang harus dilakukan untuk meningkatkan kualitas dari produk (*Product Goal*).
- *Sprint Backlog* → *Sprint Goal*.
- *Sprint Backlog* merupakan gambaran *real-time* dari pekerjaan yang ingin dicapai (*Sprint Goal*).
- *Increment* → *Definition of Done*.



(Sumber : Schwaber & Sutherland, What is Scrum?, 2021)

Gambar 2.6

Metodologi Agile : Pendekatan *Scrum*

Tahap-tahap yang ada pada pendekatan *scrum* adalah sebagai berikut :

1) *The Sprint*

Pada tahap *The Sprint*, ide atau gagasan dirubah menjadi nilai. Tahap ini umumnya memakan waktu satu bulan atau bahkan kurang. Tujuannya adalah konsistensi. Pada saat tahap ini berjalan maka mempertimbangkan beberapa hal berikut :

- Tidak ada perubahan yang dapat merubah tujuan *Sprint*.
- Kualitas tetap terjaga.
- *Product backlog* harus disempurnakan sesuai dengan kebutuhan.
- Cakupan proyek harus diperjelas.

2) *Sprint Planning*

Membuat perencanaan terkait pelaksanaan *The Sprint*. Tim perlu mempersiapkan apa saja yang dibutuhkan dalam melaksanakan *The Sprint*.

3) *Daily Scrum*

Tujuan dari *Daily Scrum* adalah untuk melakukan inspeksi progress terhadap tujuan dari *The Sprint*. Melakukan perubahan yang diperlukan pada *Sprint Backlog*.

4) *Sprint Review*

Tujuan dari *Sprint Review* adalah untuk melakukan inspeksi pada hasil *The Sprint*. Selain itu, juga menentukan adaptasi yang dilakukan di masa depan berdasarkan hasil *The Sprint*. Hasil *review* akan di presentasikan kepada *stakeholders*, serta berdiskusi terkait *progress Product Goal*.

5) *Sprint Retrospective*

Pada tahap ini, tim akan melakukan inspeksi bagaimana pelaksanaan *Sprint* yang sudah dilakukan. Tim akan memeriksa hal yang berkaitan dengan pelaksanaan *Sprint*, seperti berikut :

- Individu yang berkaitan.
- Interaksi.
- Proses yang dijalankan.
- *Tools* atau alat-alat yang digunakan.
- Pemahaman individu terkait *Definition of Done*.

Berdasarkan pembahasan metodologi yang diketahui peneliti. Terdapat kelebihan dan kekurangan pada tiap metodologi. Akan tetapi, peneliti harus menentukan metodologi yang tepat terhadap penelitian. Maka dengan pertimbangan tersebut, peneliti menentukan untuk menggunakan metodologi *SDLC*. Metodologi *SDLC* lebih cocok digunakan dalam penelitian ini ketimbang metodologi lainnya.

2.3 Studi Kelayakan

Studi kelayakan atau *feasibility* didefinisikan sebagai investigasi dalam menetapkan apakah pengembangan sebuah proyek dapat dilakukan atau justru tidak dapat dilakukan (Romney, Steinbart, Summers, & Wood, 2021). Studi kelayakan dilakukan agar dapat menentukan keputusan apakah proyek layak untuk dilakukan atau tidak layak dilakukan. Penentuan keputusan jalan atau tidak-nya sebuah proyek, ditentukan pada pengumpulan data yang berkaitan dengan proyek. Hal ini menjadi penting dikarenakan dalam mengembangkan sebuah proyek, terdapat batasan waktu serta *budget*. Menurut Romney & Steinbart, terdapat lima aspek kelayakan di dalam melakukan studi kelayakan.

2.3.1 Aspek Ekonomi

Studi kelayakan ekonomi dibutuhkan karena menjadi dasar perencanaan proyek. Tujuan dari studi kelayakan ekonomi adalah untuk mengidentifikasi manfaat finansial yang diperoleh dan biaya yang dikeluarkan selama mengembangkan proyek. Penilaian yang digunakan adalah *capital budgeting*. Pada umumnya terdapat tiga metode atau teknik dalam melakukan penilaian *capital budgeting* (Romney, Steinbart, Summers, & Wood, 2021).

Ketiga metode atau teknik penilaian adalah *payback period*, *net-present value (NPV)* dan *internal rate of return (IRR)* (Romney, Steinbart, Summers, & Wood, 2021). *Payback period* merupakan metode yang bertujuan memperhitungkan jumlah waktu atau tahun yang dibutuhkan, agar *net savings* bernilai sama atau impas dengan biaya investasi pertama. Selanjutnya adalah metode *NPV*, yang bertujuan untuk menentukan nilai waktu uang. Metode ini menghitung estimasi jumlah aliran kas masa depan, yang sudah di-diskonto ke masa sekarang. Sedangkan metode *IRR*, merupakan metode yang memperhitungkan tingkat bunga yang membuat total biaya sekarang, bernilai sama dengan total *savings* sekarang.

Perhitungan metode *payback period* dipilih, agar dapat mengetahui berapa lama waktu / tahun yang dibutuhkan dalam mencapai *break even point*. *Payback period* ditunjukkan dengan rumus berikut :

$$\text{Payback Period} = \frac{\text{initial investment}}{\text{average annual net cash inflow}}$$

Sedangkan metode *net present value* dipilih, untuk mengetahui apakah sistem informasi yang dikembangkan merupakan proyek yang layak atau tidak layak. Metode *NPV* menggunakan rumus perhitungan sebagai berikut :

$$NPV = -C_0 + \frac{C_t}{(1+r)^t}$$

Dimana :

NPV = *Net Present Value*

$-C_0$ = Arus kas keluar sekarang

C_t = Arus kas masuk pada waktu t

r = *Discount Rate* (dalam bentuk desimal)

Metode *net-present value* digunakan untuk mengetahui apakah proyek memungkinkan untuk dilaksanakan atau tidak. Proyek layak dilaksanakan apabila *NPV* yang diperoleh bernilai positif, sedangkan tidak layak dilaksanakan jika bernilai negatif.

Perhitungan *internal rate of return* digunakan untuk mengetahui apakah sebuah proyek harus diterima atau ditolak. Metode *internal rate of return* (disingkat menjadi *IRR*), menggunakan rumus perhitungan sebagai berikut :

$$IRR = i1 + NPV1 - NPV2 \ i2 - i1$$

Dimana :

IRR = *Internal Rate of Return*

$i1$ = Tingkat diskonto yang mampu menghasilkan *NPV* positif

$i2$ = Tingkat diskonto yang mampu menghasilkan *NPV* negatif

NPV1 = *Net Present Value* Positif

NPV2 = *Net Present Value* Negatif

2.3.2 Aspek Teknikal

Aspek ini membahas apakah sistem dirancang dan diimplementasikan dengan menggunakan teknologi yang ada. Teknologi menyangkut *hardware* dan *software* yang akan digunakan. Kelayakan secara teknikal akan bergantung pada kemampuan perusahaan, mampu atau tidak mampu menyediakan *hardware* dan *software*.

2.3.3 Aspek Legalitas

Aspek legal membahas apakah sistem informasi yang dirancang tidak melanggar aturan hukum yang berlaku. Aspek legal mengharuskan sistem informasi yang dirancang dapat menjaga kerahasiaan informasi yang dimiliki perusahaan.

2.3.4 Aspek Penjadwalan

Aspek penjadwalan sangat mempertimbangkan apakah proyek dapat diselesaikan sesuai jadwal. Terdapat beragam metode dalam menggambarkan jadwal proyek.

2.3.5 Aspek Operasional

Aspek operasional menentukan apakah perusahaan memiliki sumber daya manusia yang dapat mendesain dan mengimplementasikan sistem. Kelayakan dari operasional perusahaan sangat bergantung pada sumber daya manusia yang dimiliki oleh perusahaan.