

BAB II

TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Penelitian ini dilakukan berdasarkan penelitian-penelitian terdahulu yang pernah dilakukan dan digunakan sebagai bahan kajian. Hasil-hasil penelitian yang digunakan berhubungan dengan penelitian ini yaitu mengenai manajemen risiko proyek perangkat lunak dengan pendekatan *just in time*. Pendekatan *just in time* menggunakan SERIM (*The Software Engineering Risk Model*) sebagai pendekatan untuk menghitung risiko proyek perangkat lunak.

Penerapan manajemen risiko pada proyek perangkat lunak dengan pendekatan *Just In Time* (JIT) pada perusahaan pengembang perangkat lunak dapat digunakan untuk memperbaiki dan meminimalkan kendala yang mungkin atau sudah terjadi (Suselo, 2007). Pendekatan *Just In Time* (JIT) mengenali hampir seluruh risiko proyek dan memberikan pengetahuan kepada pengembang perangkat lunak dalam mengelola, mengukur, menilai, dan memprediksi risiko dengan metodologi agar membuat setiap proyek perangkat lunak sukses dan berhasil (Afrizal & Harjoko 2009).

Manajemen Risiko perangkat lunak dengan pendekatan JIT menggunakan SERIM untuk menghitung risiko. Hasil secara kuantitatif risiko proyek perangkat lunak pada organisasi dapat ditunjukkan oleh SERIM (Suselo, 2007). Tindakan yang dikembangkan oleh SERIM adalah mengelola risiko secara terfokus dan terencana sebelum risiko tersebut menjadi kendala dalam proyek perangkat lunak

(Stern & Arias, 2011). SERIM memungkinkan penilaian faktor-faktor risiko dalam proyek pengembangan perangkat lunak dari berbagai perspektif, berfokus pada perencanaan tindakan untuk manajemen risiko sebelum risiko tersebut terjadi, dan bisa fokus pada area risiko prioritas tertinggi. Kekurangan SERIM adalah kurangnya panduan untuk menggunakan informasi dalam identifikasi risiko (Stern & Arias, 2011).

2.2. Risiko dan Manajemen Risiko

4.2.1. Definisi mengenai risiko

Risiko mempunyai definisi berhubungan dengan kejadian yang akan datang. Risiko melibatkan perubahan, seperti perubahan pikiran, pendapat, aksi, atau tempat. Risiko juga melibatkan pilihan dan ketidakpastian bahwa pilihan itu akan dilakukan (Charette, 1989).

2.2.2. Strategi Risiko Reaktif dan Proaktif

Strategi dibutuhkan untuk menghadapi risiko, terdapat 2 strategi risiko adalah sebagai berikut:

1. Strategi Reaktif memonitor proyek terhadap kemungkinan risiko. Sumber-sumber daya dikesampingkan, padahal seharusnya sumber-sumber daya menjadi masalah yang sebenarnya atau penting.
2. Strategi Proaktif dimulai sebelum kerja teknis diawali. Risiko potensial diidentifikasi, probabilitas dan pengaruh proyek diperkirakan, dan diprioritaskan menurut kepentingan, kemudian

membangun suatu rencana untuk manajemen risiko. Sasaran utama adalah untuk menghindari risiko.

4.2.3. Risiko Perangkat Lunak

Risiko mempunyai 2 karakteristik, yaitu ketidakpastian dan kerugian. Risiko mempunyai 3 kategori risiko yaitu sebagai berikut:

1. Risiko Proyek

Risiko proyek mengancam rencana proyek. Bila risiko proyek menjadi kenyataan maka ada kemungkinan jadwal proyek akan mengalami slip, dan biaya akan menjadi bertambah. Risiko proyek mengidentifikasi biaya, jadwal, personil, sumber daya, pelanggan, dan masalah persyaratan.

2. Risiko Teknis

Risiko teknis mengancam kualitas dan ketepatan perangkat lunak yang akan dihasilkan. Bila risiko teknis menjadi kenyataan maka implementasinya menjadi sangat sulit atau tidak mungkin. Risiko teknis mengidentifikasi desain potensial, ambiguitas, implementasi, spesifikasi, interfacing, ketidakpastian teknik, verifikasi, keusangan teknik, masalah pemeliharaan, dan teknologi yang *leading edge*.

3. Risiko Bisnis

Risiko bisnis mengancam validitas perangkat lunak yang akan dibangun. Risiko bisnis membahayakan proyek dan produk. Risiko bisnis mempunyai 5 risiko utama adalah:

- a. Pembangunan produk atau sistem yang baik sebenarnya tidak pernah diinginkan oleh setiap orang (risiko pasar).
- b. Pembangunan sebuah produk yang tidak sesuai dengan keseluruhan strategi bisnis bagi perusahaan (risiko strategi).
- c. Pembangunan sebuah produk dimana sebuah bagian pemasaran tidak tahu bagaimana harus menjualnya.
 - a. Kehilangan dukungan manajemen senior sehubungan dengan perubahan pada fokus atau perubahan pada manusia (risiko manajemen).
 - b. Kehilangan hal-hal yang berhubungan dengan biaya atau komitmen personal (risiko biaya).

4.2.4. Manajemen Risiko

Manajemen risiko adalah suatu pendekatan terstruktur atau metodologi dalam mengelola ketidakpastian yang berkaitan dengan ancaman dan merupakan suatu rangkaian aktifitas manusia yaitu dengan penilaian risiko, pengembangan strategi untuk mengelolanya dan mitigasi risiko dengan menggunakan pemberdayaan sumber daya. Strategi yang diambil adalah memindahkan risiko kepada pihak lain, menghindari risiko, mengurangi efek negatif risiko, dan menampung sebagian atau semua konsekuensi risiko tertentu. Manajemen risiko tradisional berfokus pada risiko-risiko yang timbul oleh penyebab fisik atau legal seperti bencana alam atau kebakaran, kematian, dan tuntutan hukum.

Sasaran dari pelaksanaan manajemen risiko adalah untuk mengurangi risiko yang berbeda-beda berkaitan dengan bidang yang telah dipilih pada tingkat yang dapat diterima oleh masyarakat. Hal ini dapat berupa berbagai jenis ancaman yang disebabkan oleh lingkungan, teknologi, manusia, organisasi, dan politik. Di sisi lain pelaksanaan manajemen risiko melibatkan segala cara yang tersedia bagi manusia, khususnya bagi entitas manajemen risiko yaitu manusia, staff, dan organisasi (Karolak, 1998).

Tahapan-tahapan dalam manajemen risiko adalah sebagai berikut :

1. Identifikasi
2. Analisis
3. Prioritas
4. Mitigasi

2.2.4.1. Identifikasi

Sebelum dapat mengelola sesuatu hal yang pertama adalah harus mengetahui terlebih dahulu. Demikian dengan manajemen risiko, diawali dengan mengenali risiko dan memprediksikan konsekuensinya.

2.2.4.2. Analisis

Analisis risiko merupakan tahap kedua yang bertujuan untuk mengestimasi peluang terjadinya risiko dan memprediksi letak potensi risiko itu berada dengan menggunakan metode tertentu dengan persamaan-persamaan matematis. Dimana besar probabilitasnya direpresentasikan dalam bentuk angka.

2.2.4.3.Prioritas

Prioritas adalah tahap seluruh variabel yang memiliki probabilitas tinggi akan terjadinya risiko, dianalisa berdasarkan dampak yang akan ditimbulkan. Dengan menggunakan metode tertentu, dampak diukur sesuai metode dan persamaan yang digunakan.

2.2.4.4.Mitigasi

Tahap ini adalah tahap terakhir dalam manajemen proyek, dimana probabilitas yang memiliki dampak besar terhadap proyek baik itu secara teknis, biaya dan waktu akan disediakan alternatif penanganannya. Tahap dalam mitigasi adalah :

1. Menghindari risiko
2. Menyediakan solusi bagi risiko
3. Menyerahkan risiko ke pihak lain
4. Menerima risiko (tidak melakukan apapun terhadap risiko yang datang)

2.3. Manajemen Risiko Perangkat Lunak

Manajemen risiko perangkat lunak adalah pengelolaan risiko dengan meminimalkan kegagalan yang mencakup aspek *technical*, *cost*, dan *schedule* pada pengembang perangkat lunak (Karolak, 1998). Pengembangan perangkat lunak mempunyai 3 area pokok risiko adalah sebagai berikut:

1. Ketidakjelasan kebutuhan perangkat lunak mengakibatkan ketidaktepatan fungsionalitas yang dikembangkan.

2. Ketidapahaman estimasi biaya yang digunakan untuk pengembangan perangkat lunak mengakibatkan biaya yang berlebih.
3. Ketidakmampuan mengukur kinerja tim proyek pengembang perangkat lunak dalam menyelesaikan pekerjaan dan besarnya fungsionalitas mengakibatkan semakin lamanya jadwal pengembangan perangkat lunak.

2.4. Pendekatan Just In Time

Karolak (1998) menyatakan pendekatan *Just In Time* adalah ide yang populer di industri manufaktur tradisional, dan kemudian digunakan juga sebagai teknik manajemen risiko perangkat lunak. Pendekatan JIT pada proyek pengembangan perangkat lunak menggunakan filosofi yang bertumpu pada fungsionalitas, biaya, dan jadwal. Langkah awal pendekatan JIT adalah mengidentifikasi satu set kategori risiko tingkat tinggi, kemudian mengaitkan kategori risiko dengan faktor risiko, metrik risiko, dan pertanyaan yang harus ditanyakan. Pertanyaan-pertanyaan ini berguna sebagai daftar periksa untuk mengidentifikasi kelas yang berbeda dari risiko. Pendekatan JIT pada proyek pengembangan perangkat lunak adalah pendekatan yang dilakukan pihak manajemen atau manajer proyek yang bersifat *risk-driven*, konsep pendekatan yang dilakukan adalah sebagai berikut :

1. Minimasi dan antipasi risiko dalam proyek pengembangan perangkat lunak.

2. Menangani risiko sejak dini dalam proyek pengembangan perangkat lunak sehingga mengurangi waktu siklus proses, yang berimbas pada pengurangan biaya, pemenuhan jadwal, dan kesesuaian fungsionalitas.

Kegiatan dalam manajemen risiko perangkat lunak (Karolak, 1998) adalah sebagai berikut :

1. *Risk Identification*, adalah melakukan identifikasi gejala risiko yang terjadi.
2. *Risk Strategy*, adalah merancang tahapan langkah untuk menanggulangi risiko.
3. *Risk Assessment*, adalah mengukur akibat yang disebabkan oleh risiko.
4. *Risk Mitigation*, adalah melakukan mitigasi dari hasil penilaian risiko.
5. *Risk Reporting*, adalah membuat laporan mengenai seluruh kegiatan manajemen risiko sehingga dapat digunakan untuk dasar analisis manajemen risiko berikutnya.
6. *Risk Prediction*, adalah membuat perkiraan risiko yang akan terjadi berikutnya dalam pengembangan perangkat lunak.

2.5. Model SERIM (the Software Engineering Risk Model)

SERIM (*the Software Engineering Risk Model*) adalah model yang digunakan untuk memberikan manajemen pemecahan alternatif risiko pada suatu proyek perangkat lunak. SERIM digunakan sebagai pendekatan untuk menghitung risiko

proyek perangkat lunak, yaitu pendekatan berdasarkan subyek-subyek kemungkinan berdasarkan pengalaman dan analogi kejadian. Menurut Holloway (1979) langkah-langkah dalam SERIM adalah :

1. Langkah pertama adalah untuk menganalisa alternatif, alternatif harus ada ketika memutuskan kegiatan berdasarkan risiko.
2. Langkah kedua adalah membuat model yang akan mengevaluasi alternatif, model harus membantu dalam proses pengambilan keputusan dengan menilai alternatif.
3. Langkah ketiga adalah membuat pilihan, jika pilihan tidak dibuat, dengan berlalunya waktu akan menentukan pilihan untuk anda.

SERIM didasarkan pada premis bahwa alternatif manajemen pengembangan perangkat lunak selalu hadir. SERIM menggunakan bentuk pohon probabilitas dalam menangani keputusan alternatif dan penggunaan kemungkinan. Model ini menggunakan matematika probabilitas dan menggunakan konsep untuk mengatasi kemungkinan bahwa terjadinya peristiwa A terletak dalam ruang sampel S , dimana S adalah daftar semua hasil yang mungkin dari peristiwa. Aturan dari probabilitas adalah :

1. $P(A)$ adalah probabilitas dari peristiwa A ,
2. $0 \leq P(A) \leq 1$,
3. $P(S) = 1$, $P(\emptyset) = 0$, Jika $A_1, A_2, A_3, \dots, A_n$ adalah urutan peristiwa yang saling terpisah dari yang lain, maka $P(A_1 \cup A_2 \cup A_3 \dots A_n) = P(A_1) + P(A_2) + \dots + P(A_n)$.

SERIM menggunakan pendekatan probabilitas subjektif Bayesian untuk penilaian risiko perangkat lunak. Pendekatan ini memberikan probabilitas subjektif berdasarkan pengalaman sebelumnya atau analogi peristiwa masa lalu, yaitu pandangan pribadi mengukur kemungkinan atau kewajaran peristiwa A yang akan terjadi. Ini akan menarik untuk dicatat bahwa jika lebih dari 1 orang menilai probabilitas subjektif, kemudian perbedaan hasil dapat diharapkan.

Karolak (1996) menyatakan, untuk 2 peristiwa A dan B, $P(A)$ lebih besar dari atau sama dengan $P(B)$ jika dan hanya jika A dianggap lebih mungkin dibandingkan B. Probabilitas adalah ukuran dari keyakinan seseorang dalam terjadinya suatu peristiwa. Untuk SERIM, penugasan nilai numerik untuk pertanyaan metrik risiko perangkat lunak berbagi subjektivitas yang sama dalam arti bahwa orang yang berbeda mungkin berakhir dengan nilai yang berbeda berdasarkan masa lalu mereka dan pengalaman yang beragam, produk bisnis, dan lingkungan pengembangan perangkat lunak yang mereka nilai. Dengan demikian, probabilitas ditugaskan untuk peristiwa yang tidak menjadi nilai yang tetap, tetapi dapat berubah berdasarkan pengalaman tambahan.

Nilai-nilai numerik yang digunakan dalam SERIM ditetapkan oleh tanggapan terhadap pertanyaan metrik didefinisikan untuk melakukan wawancara. Berdasarkan tanggapan terhadap pertanyaan yang memiliki nilai antara 0 dan 1, probabilitas risiko dapat dihitung. Kemudian, pohon probabilitas digunakan untuk menghitung tingkat kesuksesan secara keseluruhan yang merupakan rata-rata probabilitas dari kejadian yang terkait dengan risiko.

SERIM berhubungan metrik risiko untuk fase life cycle perangkat lunak dan aktifitas manajemen risiko perangkat lunak. Dengan demikian, risiko perangkat lunak dapat diidentifikasi dengan fase pengembangan perangkat lunak dan berhubungan dengan setiap pertanyaan metrik yang digunakan dalam metode identifikasi risiko. Contoh menggunakan model SERIM menurut Karolak (1996) :

1. P(A) mewakili kemungkinan keberhasilan proyek perangkat lunak.
2. P(A1),P(A2), dan P(A3) mengidentifikasi kemungkinan sasaran teknis, biaya, dan penjadwalan masa depan berhasil terpenuhi.
3. P(A4) sampai dengan P(A14) mewakili kemungkinan keberhasilan faktor risiko perangkat lunak yang diidentifikasi menurut metodologi tertentu atau identifikasi risiko.
4. P(B) sampai dengan P(G) mewakili kemungkinan keberhasilan proyek perangkat lunak berdasarkan fase *life cycle* pengembangan perangkat lunak.
5. P(H) sampai dengan P(M) mengidentifikasi kemungkinan terpenuhi kegiatan manajemen risiko perangkat lunak yang diidentifikasi sebelumnya.

Pelaksanaan model SERIM menggunakan beberapa parameter dan persamaan yang harus diidentifikasi dan dipertimbangkan. Persamaan berikut digunakan untuk pohon kemungkinan menurut contoh yang diberikan oleh Karolak (1998) :

$$1. P(\text{Total}) = \left[\sum_{n=1}^3 w_n P(A_n) \right] / 3, \text{ diasumsikan jika nilai bobot dari ketiga}$$

elemen risiko berbeda maka $P(A) = w_1 P(A_1) + w_2 P(A_2) + w_3 P(A_3)$

dimana w_i adalah angka positif dan $w_1 + w_2 + w_3 = 1$.

$$2. P(\text{Elemen}) = \left[\sum_{i=4}^{13} w_n \cdot P(A_n) \right], \text{ dimana}$$

a. Sebuah metrik untuk faktor risiko diidentifikasi dan berhubungan dengan elemen yang diukur.

b. W_n adalah nilai bobot faktor risiko yang dipengaruhi terhadap elemen risiko.

$$3. P(\text{Faktor}) = \left[\sum_{i=1}^n P(Q_n) \right] / \text{total}, \text{ dimana } Q_n \text{ adalah nilai metrik untuk}$$

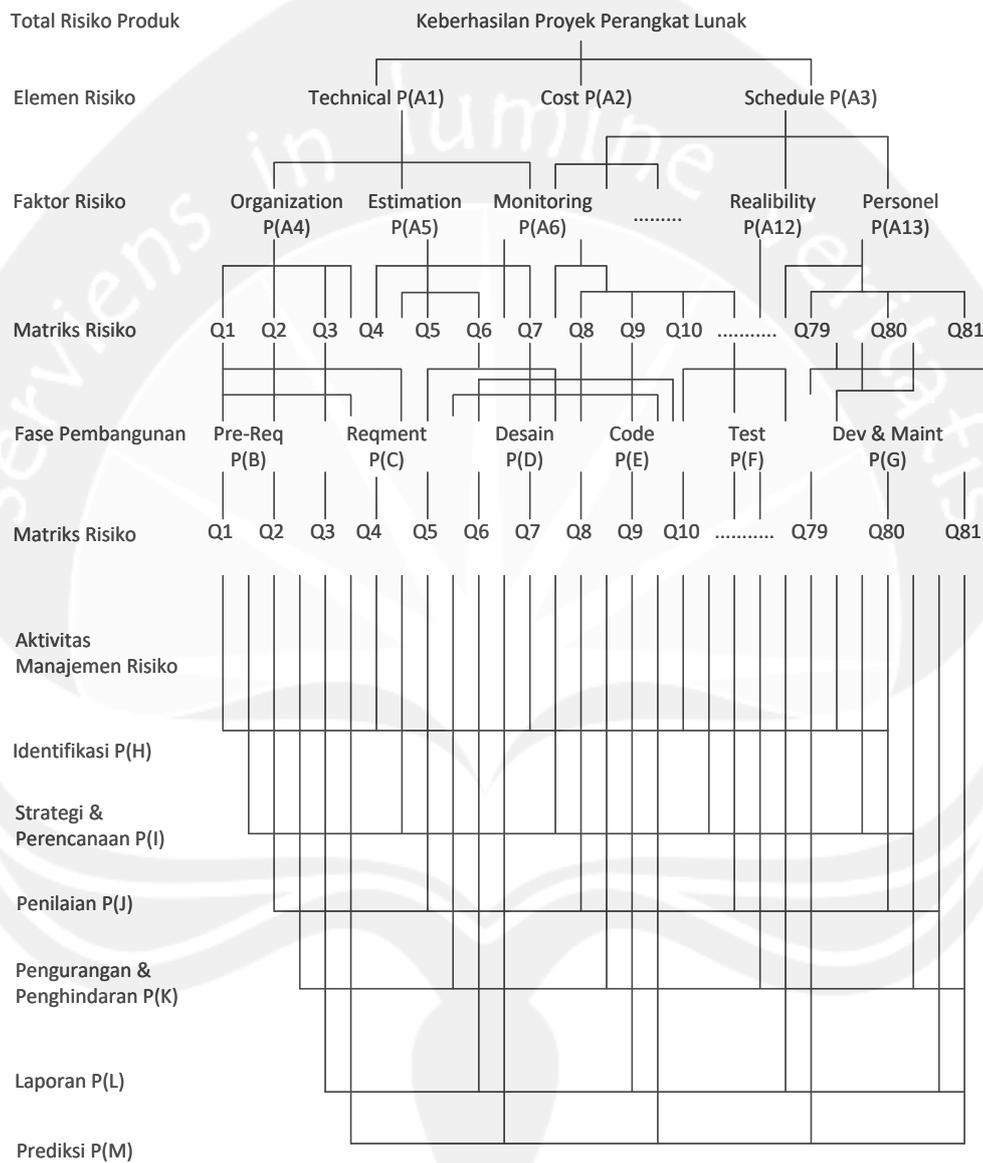
jawaban Q_n yang berhubungan dengan faktor risiko, dan total adalah jumlah jawaban yang berhubungan dengan faktor risiko.

$$4. P(\text{Development phase}) = \frac{\sum(\text{semua nilai yang ditetapkan berdasarkan jawaban yang berhubungan dengan } \textit{development phase})}{\text{banyaknya jumlah pertanyaan.}}$$

$$5. P(\text{Software management activity}) = \frac{\sum(\text{semua nilai yang ditetapkan berdasarkan jawaban yang berhubungan dengan } \textit{software management activity})}{\text{banyaknya jumlah pertanyaan.}}$$

Probabilitas untuk fase *life cycle*, faktor risiko, elemen risiko, dan aktivitas manajemen risiko bisa diwakilkan dengan pohon probabilitas yang digunakan untuk penilaian dengan persamaan untuk metrik risiko. Pada gambar dibawah

adalah pohon probabilitas yang digunakan untuk menyelesaikan persamaan-persamaan yang digunakan untuk penilaian risiko.



Gambar 1. Integrasi Model Manajemen Risiko (Karolak, 1996)

Pada gambar 1 mewakili teknik yang digunakan dalam metode SERIM, terdapat 5 *analytical perspectives* yang digunakan dalam menciptakan SERIM adalah:

1. Faktor risiko, adalah sudut pandang fundamental
2. Elemen risiko, adalah analisis bagaimana faktor risiko berdampak terhadap risiko *technical*, *cost*, dan *schedule*.
3. Kategori risiko, adalah analisis bagaimana faktor risiko terhadap proses dan produk perangkat lunak.
4. Aktivitas risiko, adalah analisis bagaimana faktor risiko berdampak terhadap identifikasi risiko, strategi dan perencanaan, penilaiian risiko, mitigasi dan penghindaran, pelaporan risiko, dan prediksi risiko.
5. Tahapan pembangunan, adalah menganalisa bagaimana faktor risiko berdampak terhadap setiap fase dari siklus pengembangan perangkat lunak yaitu *pre-requirement*, *requirement*, *design*, *coding*, dan *delivery and maintenance*.

Hubungan antara faktor risiko dan elemen risiko, menjelaskan faktor risiko perangkat lunak mengidentifikasi hubungan antara unsur-unsur risiko yang sebelumnya diidentifikasi dalam hal (produk) yang terkait dengan masalah perangkat lunak. Faktor risiko dapat berhubungan dengan lebih dari 1 unsur risiko.

Hubungan antara faktor risiko dan kategori risiko, menjelaskan mengkategorikan dalam hal risiko produk perangkat lunak dan perangkat lunak

mengakui risiko. Hubungan antara faktor risiko dan aktivitas risiko, menjelaskan kegiatan risiko yang mengidentifikasi untuk penilaian risiko dan dan model manajemen. Hubungan faktor risiko dan tahapan pengembangan, yaitu menjelaskan analisa bagaimana dampak faktor risiko pada setiap fase pengembangan perangkat lunak.

2.5.1. Ruang Lingkup SERIM (Software Engineering Risk Management)

Beberapa penelitian mengenai manajemen risiko telah diperkenalkan dan dikembangkan oleh beberapa peneliti. Kumpulan penelitian tersebut tidak dapat dibandingkan antara satu dengan lainnya, disebabkan ruang lingkup penelitian manajemen risiko yang digunakan berbeda-beda. Manajemen risiko proyek perangkat lunak harus dapat dianalisis, dinilai dan dievaluasi dari berbagai ruang lingkup proyek. Pada SERIM ruang lingkup manajemen risiko terdiri dari : elemen risiko, aktivitas risiko, faktor risiko, matrik risiko dan metodologi *life cycle*.

2.5.1.1. Elemen Risiko

Penerapan manajemen risiko pada proyek perangkat lunak tidak lepas dari pertimbangan teknologi dan bisnis. Perspektif teknologi menjelaskan alat bantu (*tools*), teknik dan lingkungan, dimana perangkat lunak tersebut diterapkan. Perspektif bisnis menjelaskan sumber daya, jadwal dan dampak bisnis (keberhasilan pembangunan perangkat lunak).

Perangkat lunak JIT mampu untuk mengelola risiko perangkat lunak, baik menurut perspektif teknologi maupun bisnis. Tidak semua risiko dalam perspektif diatas masuk ke dalam risiko perangkat lunak. Hanya terdapat 3 elemen dari risiko yang digunakan yaitu teknologi, biaya, dan penjadwalan.

Elemen teknologi berhubungan dengan kinerja perangkat lunak, yaitu : kehandalan, kualitas, fungsi, pemeliharaan dan kegunaan kembali. Elemen biaya berhubungan dengan biaya perangkat lunak selama pembangunan perangkat lunak yaitu *variable cost*, *fix cost* dan *budget*. Sedangkan elemen penjadwalan berhubungan dengan jadwal proyek selama pembangunan perangkat lunak, yaitu : jadwal realisasi, jadwal pertemuan dengan pelanggan dan anggota pengembang dan jadwal perubahan waktu proyek.

2.5.1.2. Aktivitas Risiko

Aktivitas risiko merupakan cara melakukan evaluasi terhadap risiko berdasarkan pandangan dari operasional, strategi, teknologi, bisnis, industri dan para praktisi. Terdapat 6 (enam) aktivitas yang dilakukan dalam mengevaluasi manajemen risiko perangkat lunak yaitu :

1. Identifikasi risiko yaitu melakukan pengumpulan informasi mengenai proyek perangkat lunak dan mengklasifikasikan informasi tersebut untuk menentukan risiko yang paling potensial dari suatu proyek. Informasi dikumpulkan dengan merujuk data pada proyek perangkat lunak yang pernah dikerjakan.

2. Strategi dan perencanaan risiko yaitu mengembangkan alternatif-alternatif risiko yang akan muncul selama pembangunan perangkat lunak.
3. Penilaian risiko adalah memutuskan dampak risiko yang paling potensial melalui suatu penilaian.
4. Pengurangan atau penghindaran risiko adalah aktivitas yang dilakukan dalam meminimalkan atau menghindari efek risiko.
5. Membuat laporan digunakan untuk mendokumentasikan pengelolaan risiko dari proyek perangkat lunak, termasuk melakukan perbandingan status risiko dengan risiko proyek yang pernah dikerjakan.
6. Prediksi risiko adalah melakukan prediksi tentang perkembangan risiko dari proyek dengan menggunakan iterasi data dan pengetahuan.

2.5.1.3. Faktor risiko

SERIM mengidentifikasi sepuluh faktor risiko, yaitu faktor-faktor yang merangkum risiko yang terkait dengan proyek pengembangan perangkat lunak berdasarkan pengalaman dan literatur. Masing-masing faktor risiko dinilai secara kuantitatif dengan menetapkan nilai antara 0 sampai 1, semakin rendah nilai maka proyek cenderung lebih berisiko. Walaupun secara tidak langsung berpengaruh terhadap perangkat lunak, faktor risiko sangat bermanfaat dalam menjelaskan karakteristik proyek yang dikerjakan pada masa lalu. Penelitian dari Mc Call et. al. (1977) dan Boehm (1991) menjelaskan terdapat 10 faktor risiko perangkat

lunak, dimana faktor risiko tersebut berhubungan dengan kualitas dan kehandalan produk perangkat lunak.

Satu faktor risiko dapat berhubungan lebih dari satu elemen risiko. Faktor risiko juga berpengaruh terhadap proses dan produk perangkat lunak. berdasarkan pengalaman industri perangkat lunak, setiap faktor risiko diberi pembobotn penilaian berupa tinggi, sedang, dan rendah seperti terlihat pada tabel 1, dimana bobot tersebut menyatakan derajat pengaruh faktor risiko terhadap elemen risiko.

Tabel 1. Derajat Pengaruh Faktor Risiko Terhadap Elemen Risiko

Faktor Risiko	Elemen Risiko Perangkat Lunak		
	Teknologi	Biaya	Penjadwalan
<i>Organization</i>	Rendah	Tinggi	Tinggi
<i>Estimation</i>	Rendah	Tinggi	Tinggi
<i>Monitoring</i>	Sedang	Tinggi	Tinggi
<i>Development Methology</i>	Sedang	Tinggi	Tinggi
<i>Tools</i>	Sedang	Sedang	Sedang
<i>Risk Culture</i>	Tinggi	Sedang	Sedang
<i>Usability</i>	Tinggi	Rendah	Rendah
<i>Correctness</i>	Tinggi	Rendah	Rendah
<i>Reliability</i>	Tinggi	Rendah	Rendah
<i>Personnel</i>	Tinggi	Tinggi	Tinggi

Pada tabel 1 menjelaskan derajat pengaruh faktor risiko terhadap elemen risiko. Faktor risiko *organization* adalah berkaitan dengan kedewasaan dari struktur, komunikasi, fungsi, dan kepemimpinan organisasi. Faktor risiko *organization* berpengaruh tinggi terhadap elemen risiko biaya, penjadwalan karena terkait efisiensi. Faktor risiko *estimation* berhubungan dengan ketidaktepatan estimasi pada sumber daya, jadwal, dan biaya yang diperlukan dalam proyek pengembangan perangkat lunak. Faktor risiko *monitoring* berhubungan dengan penentuan masalah. Faktor risiko *Monitoring* berpengaruh tinggi pada risiko yang berkaitan dengan pemenuhan patokan dan anggaran.

Faktor risiko *development and methodology* mengidentifikasi metode yang dipakai dalam proyek pengembangan perangkat lunak. Faktor risiko *development and methodology* berpengaruh sedang terhadap elemen risiko teknis dan berdampak besar terhadap elemen risiko biaya dan penjadwalan. Faktor risiko *tools* berkaitan dengan tools yang digunakan pada proyek perangkat lunak dan berpengaruh sedang pada elemen risiko teknis, biaya, dan penjadwalan. Faktor risiko *risk culture* berkaitan dengan proses pertimbangan dan pengambilan keputusan. Faktor risiko *usability* berkaitan dengan penundaan yang mengakibatkan tambahan pelatihan. Faktor risiko *correctness* berkaitan dengan kebutuhan klien yang telah ditentukan. Faktor risiko *reliability* berkaitan dengan terbebasnya perangkat lunak dari kesalahan eksekusi. Faktor risiko *personnel* berkaitan dengan kemampuan menggunakan metode, tools, dan pengetahuan proyek pengembangan perangkat lunak.

2.5.1.4. Matriks Risiko

Matriks Risiko digunakan untuk menilai faktor risiko dalam perangkat lunak. konsep ini ditemukan pertama kali oleh Mc Call et. al (1977) dan Boehm (1991) yang berfungsi untuk mendapatkan perangkat lunak yang berkualitas dan handal. Matrik risiko perangkat lunak merupakan kumpulan pertanyaan (kuisisioner) dengan jawaban yang diberi bobot nilai sesuai dengan pendapat responden dalam manajemen risiko proyek perangkat lunak.

2.5.1.5. Metodologi *life cycle*

Metodologi *life cycle* digunakan untuk menilai risiko fase pembangunan proyek perangkat lunak. Fase pembangunan dari perangkat lunak adalah:

1. *Pre-requirement*
2. *Requirement*
3. *Design*
4. *Code*
5. *Testing*
6. *Maintenance*