

Manajemen Resiko Proyek Pengembangan Perangkat Lunak Berbasis *Open/Free Source* Dengan Menggunakan Metode AS/NZS 4360 Dan *Analytical Hierarchy Process (AHP)*

Lili Rasyidi^{1*}; Utami Wahyuningsih¹; Arief Suardi Nur Chairat¹; Dejan Abiyyu¹;

1. Institut Teknologi PLN, Menara PLN, Jl. Lingkar Luar Barat, Duri Kosambi, Cengkareng, Jakarta Barat, DKI Jakarta 11750, Indonesia

*Email: lili_rasyidi@itpln.ac.id

Received: 12 Januari 2024 | Accepted: 15 Februari 2024 | Published: 7 Juni 2024

ABSTRACT

The need for information and communication technology is currently growing very rapidly and has influenced the emergence of various projects, especially software development projects, where these software development projects often fail, either partially or completely. The open source technology that has emerged and exploded in the last few years is really needed by various parties, but it still has several weaknesses that must be overcome by users and developers. In contrast to licensed (proprietary) technology which always gets technical support, support for open source software is not always available. This research aims to identify, compile and assess risk factors for software development projects using open source technology. This research uses two methods based on respondents' perceptions, namely the AS/NZS 4360 qualitative method and the Analytical Hierarchy Process (AHP) method. These two methods allow mapping of respondents' perceptions as well as classification and ranking of these risk factors. Identification of risk factors concludes that the permission/license selection attribute is an important part of the software development process because it influences project success. The AS/NZS 4360 qualitative risk assessment classifies risk factors into three categories: extreme, high and moderate. AHP risk analysis produces risk factor ratings and changes the risk priority weights into four ratings, namely extreme, high, medium and low.

Keywords: *software Development, risk analysis, software development risk taxonomy, qualitative risk analysis, analytical hierarchy process*

ABSTRAK

Kebutuhan akan teknologi informasi dan komunikasi saat ini berkembang sangat pesat dan mempengaruhi munculnya berbagai proyek khususnya proyek pengembangan perangkat lunak, dimana proyek pengembangan perangkat lunak ini sering kali mengalami kegagalan baik gagal sebagian atau gagal total. Teknologi open source yang muncul dan meledak dalam beberapa tahun terakhir sangat dibutuhkan oleh berbagai pihak, hanya saja masih memiliki beberapa kelemahan yang harus diatasi oleh pengguna dan pengembang. Berbeda dengan teknologi berlisensi (proprietary) yang selalu mendapatkan dukungan teknis, dukungan untuk open source software tidak selalu tersedia. Penelitian ini bertujuan untuk mengidentifikasi, menyusun dan menilai faktor resiko proyek pengembangan perangkat lunak menggunakan teknologi open source. Penelitian ini menggunakan dua metode yang berdasarkan persepsi responden, yaitu metode kualitatif AS/NZS 4360 dan metode Analytical Hierarchy Process (AHP). Kedua metode ini memungkinkan pemetaan persepsi responden serta klasifikasi dan pemeringkatan faktor risiko tersebut. Identifikasi faktor risiko menyimpulkan bahwa atribut pemilihan izin/lisensi menjadi salah satu bagian penting dalam proses pengembangan perangkat lunak karena mempengaruhi keberhasilan proyek. Penilaian resiko kualitatif AS/NZS 4360 mengklasifikasikan faktor risiko menjadi tiga kategori: ekstrim, tinggi dan

sedang. Analisis resiko AHP menghasilkan rating faktor risiko dan mengubah bobot prioritas resiko menjadi empat peringkat yaitu ekstrim, tinggi, sedang, dan rendah.

Kata kunci: *Pengembangan Perangkat Lunak, analisis risiko, taksonomi risiko pengembangan perangkat lunak, analisis risiko kualitatif, proses hierarki analitis*

1. PENDAHULUAN

Proyek pada umumnya unik dan bersifat kompleks/ rumit. Seringkali proyek mengambil tambahan waktu dan tambahan sumberdaya, baik manusia, finansial, fasilitas, material dan intelektual. Salah satu bagian dari manajemen proyek yang penting adalah adanya manajemen resiko proyek[1]. Secara umum, banyak proyek yang dapat dikatakan gagal. Kegagalan ini disebabkan berbagai faktor, baik internal maupun eksternal. Manajemen resiko proyek bertujuan untuk meminimasi resiko-resiko yang dapat menggagalkan tercapainya tujuan pengerjaan proyek dan para *stakeholder*, selain itu manajemen resiko proyek juga bertujuan untuk mengidentifikasi dan mencari peluang keuntungan. Secara khusus, manajemen resiko membantu manajer-manajer proyek menentukan prioritas, alokasi sumber daya dan implementasi aksi-aksi dan proses yang mengurangi resiko kegagalan pencapaian tujuan proyek. Manajemen proyek diperlukan juga untuk jenis proyek pengembangan *software* yang banyak bermunculan saat ini[2], [3].

Proyek pengembangan perangkat lunak/*software*[4], [5], [6], [7] sering mengalami kegagalan baik secara total ataupun secara parsial. Secara umum, beberapa masalah yang terjadi pada proses ini antara lain: keterlambatan waktu penyelesaian, biaya yang melebihi perkiraan, dan lainnya. Banyak kasus pengembangan *software* yang dikerjakan oleh vendor pengembang *software* pada awalnya direncanakan untuk dikerjakan dalam waktu beberapa bulan, tapi ternyata dikerjakan dengan rentang waktu yang beberapa kali lipat. Melihat kejadian ini, seharusnya dari awal proyek sudah dapat diidentifikasi dan diprediksi pada bagian apa saja yang kira-kira akan menghambat penyelesaian proyek atau menjadi faktor resiko proyek[8], [9].

Carr et al, dari *software* Engineering Institute (SEI) melakukan identifikasi resiko dengan cara membuat hirarki proses pengembangan *software* dan membentuknya kedalam sebuah taksonomi[10], dimana taksonomi ini berisikan kelas, elemen dan atribut resiko. Kelas yang terdapat pada taksonomi tersebut terdiri dari 3 bagian yaitu *engineering specialities*, *process development* dan *interface program*. Peer, [11] mengidentifikasikan faktor resiko dalam pengembangan *software* menjadi enam faktor yaitu *defects*, *security*, *vendor lock in*, *sustainability*, *migration* dan *legal*. *Defects* lebih dikenal sebagai *error* atau *bugs* merupakan resiko paling umum yang terjadi dalam sebuah pengembangan sistem *software*. *The Federal Financial Institutions Examination Council/FFIEC*, [12] membagi resiko proyek pengembangan *software open source* kedalam tiga bagian utama yaitu *strategic risk*, *operational risk* dan *legal risk*. *Requirements* dari sebuah *software* bergantung dari tujuan bisnis strategis dari institusi tertentu, hal inilah yang menimbulkan dimunculkannya *strategic risk* sebagai salah satu faktor resiko. *Operational risk* muncul dalam lingkungan mengoperasikan infokom. *Operational risk* mempertimbangkan penggunaan *software open source* yang menggaransi kepentingan-kepentingan yang meliputi integritas kode, kecukupan dokumentasi, rencana kontingensi dan dukungan. *Legal risk* berhubungan dengan masalah legalitas dan lisensi dari sebuah *software open source* yang digunakan dan yang dihasilkan.

Taksonomi yang disusun oleh Carr et al [10], sudah dapat mewartakan resiko-resiko yang muncul pada saat itu, akan tetapi dengan berkembangnya zaman, dan mulai munculnya berbagai jenis *framework software* memunculkan masalah baru yaitu pemilihan *framework* pengembangan tersebut. Pada saat *framework* yang dipilih tersebut adalah *open source*[12], maka didalam *framework* itu sendiri terdapat pemilihan lisensi juga.

Setelah identifikasi dilakukan, tahap selanjutnya adalah melakukan penilaian resiko. Penilaian resiko sangat berhubungan dengan faktor subyektifitas dari para penilai itu sendiri dimana subyektifitas merupakan pemetaan dari persepsi para penilai, dalam hal ini adalah para responden ahli/pakar. Untuk itu, metode penilaian resiko yang digunakan adalah metode yang dapat memetakan persepsi dari responden tersebut[13], [14]. Metode penilaian resiko yang dipilih pada penelitian ini

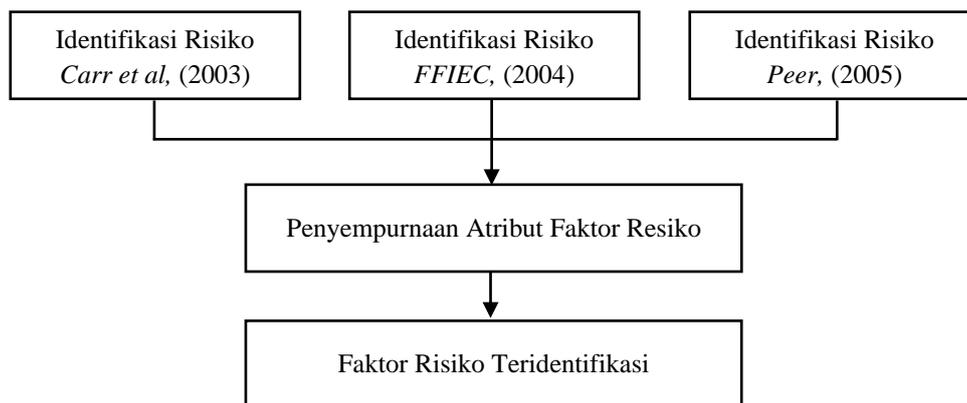
adalah metode kualitatif[15] dari AS/NZS 4360 dan metode AHP[16] dari Saaty. Selain dianggap dapat memetakan persepsi dari responden, Kedua metode ini dapat memberikan informasi tingkat dan klasifikasi resiko.

2. METODE/PERANCANGAN PENELITIAN

Tahap-tahap penelitian ini disusun kedalam enam tahapan sistematis, yaitu tahap Persiapan Penelitian, Studi Pendahuluan, Desain Penelitian, Pengumpulan dan Pengolahan Data, Analisis, serta Kesimpulan dan Saran. Desain penelitian menggunakan Model analisis resiko yang kualitatif AS/NZS 4360 (Australia New Zealand Standard 4360) dan analisis resiko *Analytical Hierarchy Process*. Serta dengan menggabungkan beberapa penelitian yang dihasilkan maka diusulkan model manajemen resiko untuk kasus ini. Pengumpulan data dilakukan menggunakan metode survei dengan media kuesioner yang disebar ke sejumlah ahli pengembangan *software*, baik analis maupun *programmer* dengan kriteria tertentu.

2.1. Identifikasi Resiko

Identifikasi resiko dilakukan dengan menggali faktor-faktor resiko yang berkaitan proses pengembangan *software* berbasis *open source*. Identifikasi faktor resiko dilakukan dengan studi literatur pada tiga penelitian mengenai identifikasi yang pernah dilakukan sebelumnya yaitu identifikasi faktor resiko Carr et al [10], identifikasi faktor resiko yang dilakukan FFIEC[12] , serta identifikasi Peer [11].



Gambar 1. Proses Identifikasi Resiko

Seperti terlihat pada gambar diatas bahwa dari studi literatur yang dilakukan, akan didapatkan atribut baru yang menjadi bahan penyempurnaan bagi faktor-faktor resiko hasil identifikasi Carr et al [10] yang menjadi dasar penulisan thesis ini.

Identifikasi Faktor Resiko Carr et al.

Carr et al [10] mengidentifikasi resiko proyek *software* secara umum kedalam sebuah taksonomi proyek pengembangan *software*. Taksonomi ini diperinci kedalam bentuk hirarki beruntun yang terdiri dari kelas, elemen, dan atribut. Carr et al, (1993) menyatakan proses pengembangan *software* terdiri dari 3 kelas yaitu : *product engineering/* rekayasa produk, *development environment/* lingkungan pengembangan dan *program constraint/* pembatas program.

Elemen-elemen dalam kelas tersebut adalah:

1. Kelas Rekayasa Produk :

Product Engineering atau Rekayasa Produk adalah aspek-aspek teknik dari pekerjaan yang akan diselesaikan. Kelas ini terdiri dari elemen-elemen: *Requirements*/kebutuhan, Desain, Kode dan tes unit, Tes dan Integrasi, Kemampuan Khusus Rekayasa

2. Kelas Lingkungan Pengembangan

Development Environment atau Lingkungan Pengembangan terdiri dari metode-metode, prosedur dan tools yang digunakan untuk menghasilkan produk. Kelas ini terdiri dari elemen-elemen: Proses Pengembangan, Sistem Pengembangan, Proses Manajemen, Metode Manajemen, Lingkungan Kerja

3. Kelas Pembatas Program

Program Constraint atau Pembatas Program terdiri dari faktor kontraktual, organisasional dan operasional dimana *software* dikembangkan tetapi secara umum berada diluar kendali langsung dari manajemen lokal. Kelas ini terdiri dari elemen-elemen: Sumber Daya, Kontrak, Tampilan Antarmuka.

Setiap faktor yang terdapat pada elemen taksonomi memiliki atribut-atribut tersendiri. Atribut-atribut tersebut dijelaskan secara rinci pada bab landasan teori. Atribut-atribut ini merupakan aktifitas terkecil dari faktor-faktor resiko dalam proses pengembangan *software*.

Identifikasi Faktor Resiko FFIEC[12]

Identifikasi faktor resiko yang dilakukan oleh *Federal Financial Institution Examination Council (FFIEC)* menyimpulkan faktor resiko dalam manajemen resiko *Free and Open Source software (FOSS)* terdiri dari tiga faktor utama yaitu :

1. *Strategic Risk*
2. *Operational Risk*
3. *Legal Risk*

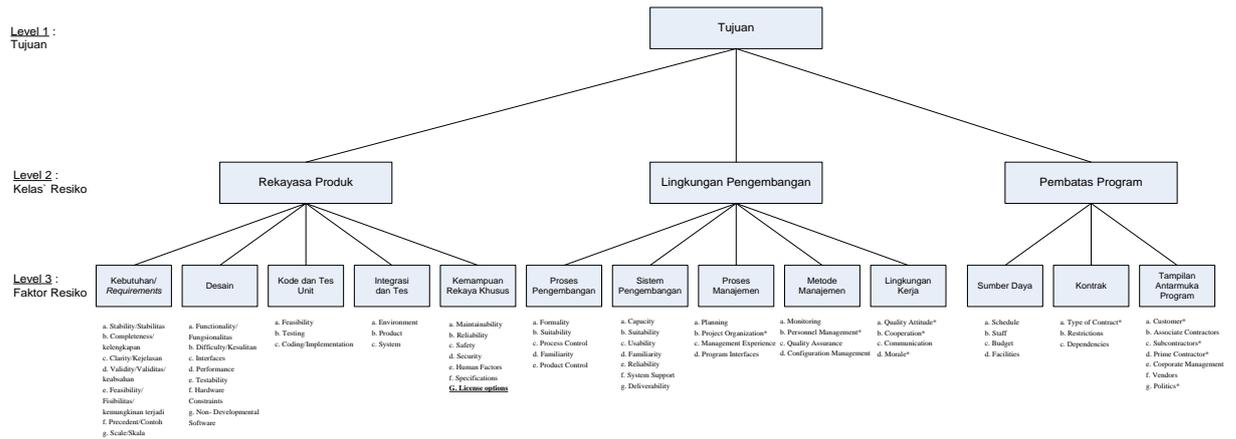
Identifikasi Faktor Resiko Peer[11]

Peer [11] menyatakan bahwa proses penilaian resiko pengembangan *software open source* memiliki 6 faktor resiko. Faktor resiko tersebut adalah

1. *Defects*
2. *Security*
3. *Vendor Lock-in*
4. *Sustainability*
5. *Migration Risk*
6. *Legal Risk*

Pengembangan Atribut

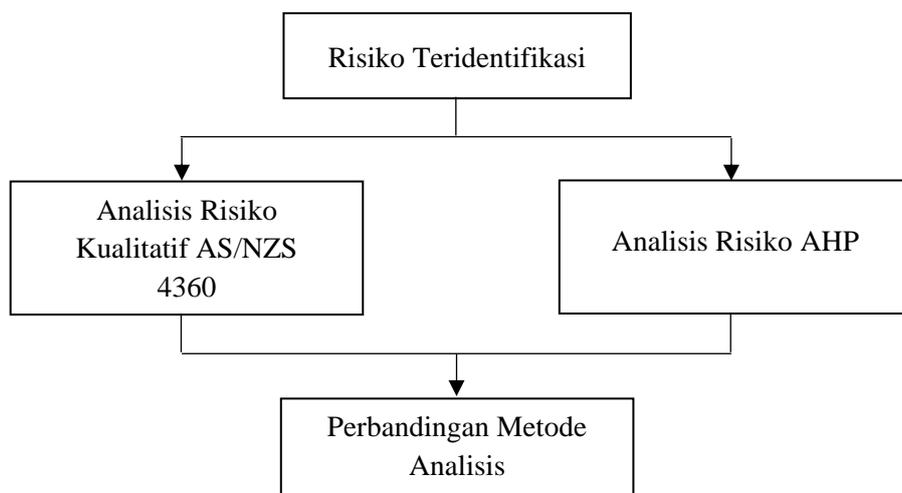
Resiko-resiko yang dimunculkan pada tahap ini merupakan hasil bangkitan dari teori yang dikembangkan oleh Carr et al [10] mengenai identifikasi resiko proyek pengembangan *software* yang kemudian disesuaikan berdasarkan penelitian identifikasi faktor resiko dari FFIEC[12] dan Peer[11] Jumlah resiko teridentifikasi yang muncul adalah 13, sama dengan jumlah elemen pada taksonomi, hanya saja terdapat pengembangan atribut dalam faktor-faktor resiko teridentifikasi, yaitu atribut pemilihan lisensi, dimana aktifitas ini berhubungan dengan faktor legal pada penelitian lain. Pemilihan lisensi yang dimaksud disini adalah lisensi *software open source* yang digunakan pada saat akan membuat *software* aplikasi yang diinginkan.



Gambar 2. Proses Identifikasi Risiko

2.2. Analisis Identifikasi Resiko

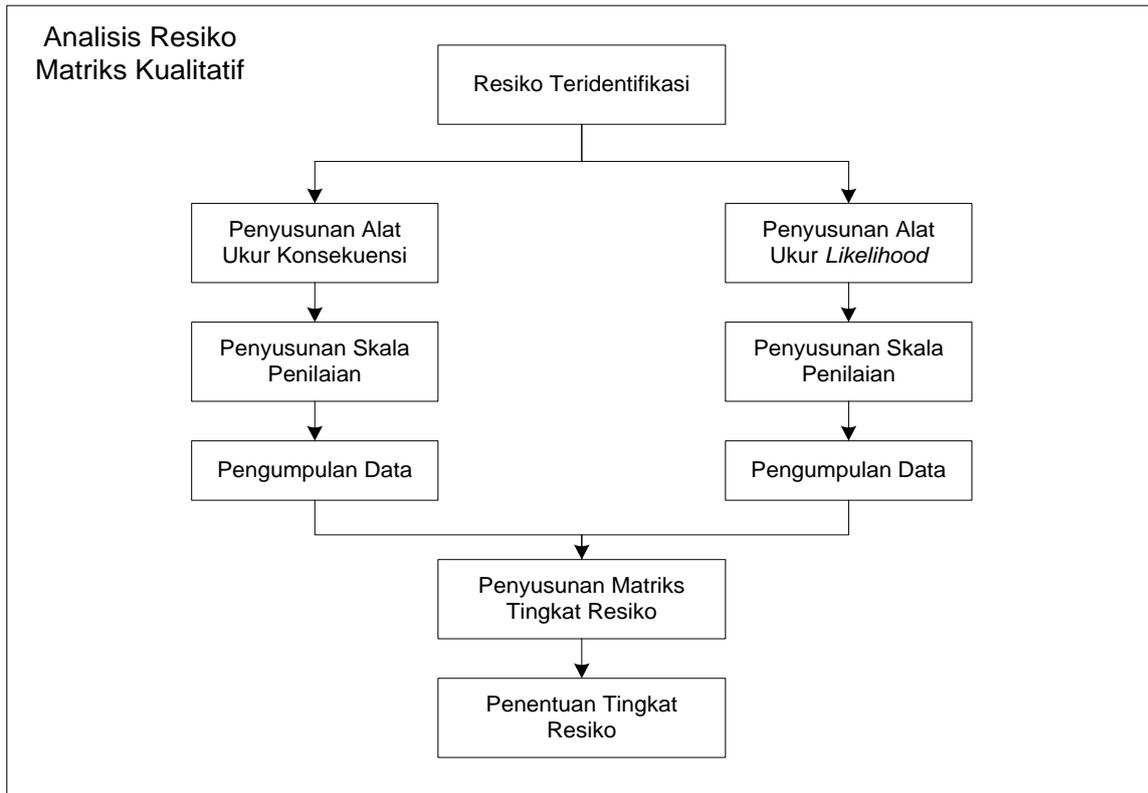
Pada tahap ini, semua resiko yang telah teridentifikasi dinilai dengan dua cara yaitu dengan menggunakan matriks kualitatif *likelihood-impact* dari AS/NZS 4360 dan menggunakan metode *Analytical Hierarchy Process*. Kedua metode ini digunakan untuk membandingkan hasil dari persepsi para responden mengenai faktor-faktor resiko yang muncul dalam pengembangan aplikasi *software* berbasis *open source*.



Gambar 3. Rumusan Analisa Risiko

2.2.1. Analisis Resiko AS/NZS 4360

Analisis ini merupakan analisis resiko kualitatif dimana penilaian resiko akan menghasilkan pernyataan klasifikasi tingkat resiko dari masing-masing faktor resiko yang dinilai. Tahapan yang dilakukan adalah menyusun kuesioner yang kemudian disebar kepada responden ahli. Kuesioner berisikan pertanyaan akan tingkat kemunculan/probabilitas kemunculan faktor-faktor resiko dan tingkat pengaruh masing-masing faktor resiko. Hasil isian kuesioner yang disebar tersebut merupakan pemetaan persepsi responden. Dari kedua isian ini AS/NZS 4360 membuat matrik baru yang menghasilkan penilaian resiko yang berbentuk klasifikasi resiko. Hasil penilaian ini akan membagi faktor resiko kedalam 4 (empat) tingkat klasifikasi yaitu ekstrim (*extreme*), tinggi (*high*), sedang (*moderate*) dan rendah (*low*).

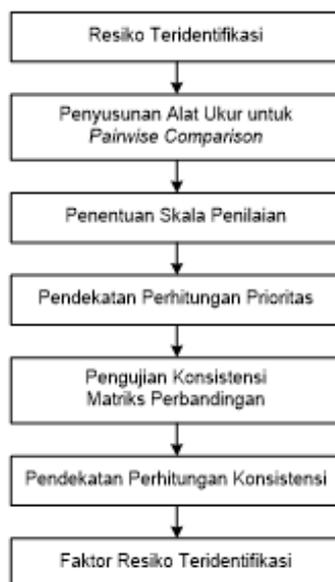


Gambar 4. Analisa Resiko Kualitatif

2.2.2. Analisis Resiko AHP

Penilaian resiko dengan metode ini bertujuan untuk menghasilkan penilaian resiko yang lebih rinci. Perincian yang dimaksud berupa prioritas atau ranking dari faktor resiko yang dinilai. Tahapan proses penilaian resiko dengan metode ini terlihat pada gambar 5.

Tahap pertama setelah identifikasi faktor telah berhasil dilakukan adalah menyusun alat ukur penilaian yang berupa kuesioner. Kuesioner yang dibangun berisikan penilaian perbandingan antar resiko yang nantinya diolah dengan metode *pairwise comparison*.



Gambar 5. Analisa Resiko AHP

3. HASIL DAN PEMBAHASAN

Pengumpulan data dilakukan dengan cara memberikan kuesioner kepada responden ahli dengan syarat para ahli yang menjadi responden telah bekerja di bidang pengembangan *software* Minimal telah 2 tahun bekerja dan telah memiliki pengalaman melakukan proses pengembangan/pembangunan *software* baik dengan *proprietary software* (berlisensi) maupun *open source software*. Responden terdiri dari *programmer* ataupun *System Analis* berpengalaman.

Kuesioner yang diberikan terdiri dari dua jenis isian. Isian pertama mengenai tingkat kemunculan faktor resiko yang telah teridentifikasi sebelumnya dan juga tingkat pengaruh/konsekuensi pada proyek. Isian kedua berisikan penilaian akan prioritas faktor-faktor dalam proses pengembangan *software*. Penilaian ini merupakan pendekatan penilaian berpasangan.

Teknologi *Open source* yang merupakan salah satu bagian dari atribut/faktor resiko Sistem Pengembangan/ *Development System* dan dari kelas Lingkup Pengembangan/ *Work environment*.

Penilaian resiko sangat berkaitan erat dengan subyektifitas para ahli dalam bidang yang difokuskan dimana subyektifitas ini perlu dihitung dan dirumuskan. Untuk itu penilaian resiko dilakukan dengan metode-metode yang dapat mengadopsi subyektifitas tersebut. Penilaian dan klasifikasi faktor resiko dilakukan pada penelitian ini dilakukan dengan metode kualitatif yang diusulkan oleh AS/NZS 4360[17] dan metode AHP dari Saaty [18].

3.1. Analisis Resiko Matriks Kualitatif AS/NZS 4360

Penilaian resiko merupakan proses pengklasifikasian berdasarkan AS/NZS 4360. Klasifikasi resiko ini dibagi kedalam empat tingkat yaitu resiko yang bersifat ekstrim, tinggi, medium, dan rendah. Klasifikasi ini merupakan kombinasi antara tingkat kemunculan resiko dan tingkat pengaruh resiko tersebut pada keberhasilan proyek.

Tabel 1. Matriks Analisis Resiko Kualitatif - Tingkat Resiko

Faktor Resiko	Likelihood	Konsekuensi	Kelas
1: Requirements	1	4	E
2: Design	1	4	E
3: Engineering Specialities	2	4	E
4: Code and Unit Test	2	3	H
5: Development Process	2	3	H
6: Development System	2	3	H
7: Work Environment	3	3	H
8: Resources	2	3	H
9: Test and Integration	3	2	M
10: Management Process	4	3	M
11: Management Method	4	3	M
12: Contract	4	3	M
13: Program Interfaces	4	3	M

E : Resiko Ekstrim; Dibutuhkan aksi secepatnya

H : Resiko Tinggi; Diperlukan perhatian manajemen Senior

M : Resiko Sedang; Tanggung jawab manajemen harus lebih spesifik

L : Resiko Rendah; Cukup diatur oleh prosedur rutin

Hasil pengumpulan data penelitian ini menghasilkan bahwa semua faktor ternyata berada dalam tingkat resiko medium keatas. Tidak muncul adanya faktor resiko dalam tingkat rendah. Pada penilaian tingkat resiko ini juga didapatkan tiga faktor resiko berada dalam kategori ekstrim. Faktor tersebut adalah *requirements*, *desain* dan *engineering specialities*. Kelompok tingkat resiko lain adalah tingkat tinggi/high. Kelompok ini terdiri dari faktor resiko *code and unit test*, *development*

process, development system, work environment dan *resource*. Kelompok resiko lainnya adalah kelompok resiko medium yaitu faktor resiko *test and integration, management process, management method, contract* dan *interface program*. Interpretasi hasil fungsi probabilitas ini bagi klasifikasi ekstrim adalah bahwa faktor resiko pada tingkat ini memiliki tingkat resiko paling besar diantara yang lain. Perlu penanganan lebih pada saat melakukan penanganan. AS/NZS mengarahkan untuk segera dilakukan aksi yang secepatnya. Untuk faktor resiko berada pada klasifikasi tinggi maka diperlukan perhatian lebih dari manajemen senior. Hal ini berarti tingkat perhatian harus lebih daripada faktor resiko biasanya. Klasifikasi faktor resiko moderate cukup ditangani oleh beberapa bagian spesifik dari manajemen.

3.2. Analisis Resiko Metode Analytical Hierarchy Process /AHP

Metode lain yang digunakan dalam penilaian resiko pada penelitian ini adalah metode AHP. Metode ini sudah sangat dikenal luas kekuatannya. Resiko proses pengembangan *software* yang dibagi menjadi 3 elemen utama kemudian diturunkan kembali menjadi 13 faktor resiko. Dua tingkatan ini dinilai kepentingannya serta bobot perbandingannya.

Analisis AHP Level 1 Tujuan

Penilaian AHP pada level 1 menghasilkan urutan resiko *product engineering, development environment* dan *program constraint* sebagai urutan 1, 2 dan 3. dengan bobot *product engineering* yang jauh lebih besar ini, menyatakan bahwa persepsi responden akan resiko terbesar dalam proyek pengembangan *software* adalah pada kelas *product engineering* ini.

Analisis AHP Level 2.1. Kelas Product Engineering

Penilaian AHP pada level 2.1 menghasilkan urutan resiko *engineering specialities, requirements, design, code and unit test* dan *test and integration* sebagai urutan 1,2,3,4 dan 5. Atribut *engineering specialities* ini menempati urutan pertama pada kelas ini sesuai persepsi responden bahwa pemilihan lisensi sangat beresiko dan berpengaruh pada keberhasilan proses.

Analisis AHP Level 2.2. Kelas Development System

Penilaian AHP pada level 2.2 menghasilkan urutan resiko *development process, development system, management method, management process* dan *work environment* sebagai urutan 1,2,3,4, 5. Atribut *development process* ini menempati urutan pertama pada kelas ini sesuai persepsi bahwa *development process* paling berpengaruh pada kelas *development system*.

Analisis AHP Level 2.3. Program Constrain

Penilaian AHP pada level 2.3 menghasilkan urutan resiko *resource, contract* dan *interface program* sebagai urutan 1,2 dan 3. Atribut *resource* ini menempati urutan pertama pada kelas ini sesuai persepsi responden bahwa *resource* paling berpengaruh pada kelas *program constrain*. Penilaian resiko dengan metode AHP dengan didapatkan urutan faktor-faktor resiko seperti yang terlihat pada tabel diatas. Hasil bobot masing-masing faktor yang telah terurut kemudian dilakukan konversi skala kedalam skala nominal agar dapat dibandingkan dengan metode penilaian dari AS/NZS 4360.

Tabel 2. Ranking Resiko

Ranking Resiko			
Ran	Item	Bobot	Klasifikasi
1	Engineering Specialities	0,155	Ekstrim
2	Requirements	0,109	Tinggi
3	Design	0,107	Tinggi
4	Code & Unit Test	0,091	Medium
5	Resource	0,089	Medium
6	Development Process	0,087	Medium
7	Contract	0,072	Medium
8	Development System	0,062	Rendah
9	Test & Integration	0,050	Rendah
10	Management Method	0,045	Rendah
11	Interface Program	0,044	Rendah
12	Work Environment	0,044	Rendah
13	Management Process	0,044	Rendah
	Jumlah	1,000	

Dari tabel diatas dapat dilihat bahwa semua faktor terbagi kedalam 4 kategori dengan *engineering specialities* adalah faktor resiko terbesar, dan masuk dalam skala ekstrim. Berbeda dengan pada hasil metode AS/NZS 4360, pada tabel ini terdapat klasifikasi rendah dan bahkan jumlahnya mendominasi. Hal ini dapat diinterpretasikan bahwa persepsi responden mengatakan 6 faktor resiko berada pada klasifikasi rendah atau tidak memiliki resiko yang secara signifikan dapat menggagalkan keberhasilan proyek.

3.3. Analisis Perbandingan Penilaian Resiko

Kedua metode penilaian yang diterapkan pada studi kasus ini menghasilkan nilai klasifikasi yang sedikit berbeda, walaupun kedua metode tersebut sama-sama berdasarkan persepsi responden ahli yang sama.

Tabel 3. Perbandingan Resiko

Faktor Resiko		AS/NZS	AHP
Rekayasa Produk	1. Requirements	Ekstrim	Tinggi
	2. Design	Ekstrim	Tinggi
	3. Code and Unit Test	Tinggi	Medium
	4. Test and Integration	Medium	Rendah
	5. Engineering Specilities	Ekstrim	Ekstrim
Lingkungan Pengembangan	1. Development Process	Tinggi	Medium
	2. Development System	Tinggi	Rendah
	3. Management Process	Medium	Rendah
	4. Management Method	Medium	Rendah
	5. Work Environment	Tinggi	Rendah
Pembatas Program	1. Resources	Tinggi	Medium
	2. Contract	Medium	Rendah
	3. Program Interfaces	Medium	Rendah

Dari tabel perbandingan diatas dapat dilihat bahwa terdapat perbedaan hasil klasifikasi antara dua metode ini. Hanya beberapa faktor yang memiliki hasil yang sama dalam klasifikasi tingkat resiko. Faktor resiko tersebut adalah *engineering specialities* sedangkan Faktor Resiko Development System berada di klasifikasi Tinggi untuk hasil penilaian menggunakan metoda AS/NZS dan klasifikasi rendah dengan metode AHP.

4. KESIMPULAN DAN SARAN

Dari hasil penelitian analisis resiko proses pengembangan *software* berbasis teknologi *open source* dapat disimpulkan bahwa Faktor-faktor resiko proses pengembangan *software* dapat dibangun dengan cara membuat struktur hirarki yang terdiri dari kelas, elemen dan atribut. Selain sebagai acuan kerja proyek, struktur hirarki tersebut menjadi acuan penentuan faktor resiko. Perbedaan level klasifikasi resiko akan menentukan perbedaan fokus dan prioritas penanganan resikonya. Teknologi *open source* sebagai teknologi yang digunakan dalam proses pengembangan *software* diidentikan sebagai salah satu bagian dari kelas lingkungan pengembangan/ *development working* dan element sistem pengembangan/*development system* terlihat berada didalam faktor resiko yang tinggi apabila menggunakan metode AS/NZS, dan resiko rendah bila menggunakan AHP. Konversi hasil penilaian resiko AHP kedalam bentuk klasifikasi seperti pada matriks kualitatif AS/NZS 4360 membuat interval klasifikasi faktor resiko yang berbeda. Perbedaan hasil klasifikasi dari kedua metode dapat terjadi karena adanya perbedaan cara pengisian dan skala awal penilaian resiko, serta perbedaan persepsi dari para ahli yang menjadi subyek pengisi kuesioner. Perbedaan juga dapat terjadi bisa terjadi karena ketidak-presisian persepsi dari responden akan penilaian tingkat resiko dengan dua metode yang berbeda.

UCAPAN TERIMAKASIH

Penulis mengucapkan terima kasih kepada semua pihak yang telah memberi dukungan yang membantu pelaksanaan penelitian dan atau penulisan artikel.

DAFTAR PUSTAKA

- [1] D. F. Cooper, S. Grey, and G. Raymond, "Project Risk Management Guidelines: Managing Risk in Large Projects and Complex Procurements," 2005.
- [2] R. E. Fairley, "Software Risk Management."
- [3] R. P. Higuera and Y. Y. Haimes, "Software Risk Management," 1996. [Online]. Available: <http://www.raai.com>
- [4] D. Brandon, *Project management for modern information systems*. IRM Press, 2006.
- [5] J. Masso, F. J. Pino, C. Pardo, F. García, and M. Piattini, "Risk management in the software life cycle: A systematic literature review," *Computer Standards and Interfaces*, vol. 71. Elsevier B.V., Aug. 01, 2020. doi: 10.1016/j.csi.2020.103431.
- [6] K. Irfandhi, "Risk Management in Information ... (Kornelius Irfandhi) RISK MANAGEMENT IN INFORMATION TECHNOLOGY PROJECT: AN EMPIRICAL STUDY."
- [7] P. L. Bannerman, "Risk and risk management in software projects: A reassessment," *Journal of Systems and Software*, vol. 81, no. 12, pp. 2118–2133, Dec. 2008, doi: 10.1016/j.jss.2008.03.059.
- [8] M. Boban, Ž. Požgaj, and H. Sertić, "STRATEGIES FOR SUCCESSFUL SOFTWARE DEVELOPMENT RISK MANAGEMENT," 2003.
- [9] P. L. Bannerman, "Risk and risk management in software projects: A reassessment," *Journal of Systems and Software*, vol. 81, no. 12, pp. 2118–2133, Dec. 2008, doi: 10.1016/j.jss.2008.03.059.
- [10] M. J. Carr, S. L. Konda, I. Monarch, F. Carol, U. Clay, and F. Walker, "Taxonomy-Based Risk Identification," 1993.
- [11] B. Edwin Peer, "Solutions to Open Source Risk Assessment: How and When Can Open Source Projects Be Classified As Mature?" [Online]. Available: <http://www.groklaw.net>

- [12] Federal Financial Institutions Examination Council, “Risk Management in Projects Based on Open-Source Software.” [Online]. Available: www.opensource.org. [End
- [13] Kindinger, “Risk Factor Analysis-New qualitative model,” 2000.
- [14] N. D. Linh, P. D. Hung, V. T. Diep, and T. D. Tung, “Risk management in projects based on open-source software,” in *ACM International Conference Proceeding Series*, Association for Computing Machinery, 2019, pp. 178–183. doi: 10.1145/3316615.3316648.
- [15] J. Emblemståg and L. E. Kjølstad, “Qualitative risk analysis: Some problems and remedies,” *Management Decision*, vol. 44, no. 3. pp. 395–408, 2006. doi: 10.1108/00251740610656278.
- [16] T. L. Saaty, “Decision making with the analytic hierarchy process,” 2008.
- [17] Standards Association of Australia. and Standards New Zealand., *Risk management*. Standards Association of Australia, 1999.
- [18] T. L. Saaty and J. M. Katz, “How to make a decision: The Analytic Hierarchy Process,” 1990.