

ANALISIS METODE ESTIMASI BIAYA PADA PERANGKAT LUNAK BESERTA FAKTOR-FAKTOR YANG MEMPENGARUHI: A SYSTEMATIC LITERATURE REVIEW

Amelia Devi Putri Ariyanto¹, Luthfiyatul 'Azizah², Umi Laili Yuhana³

¹Institut Teknologi Sepuluh Nopember, Surabaya
Email: ¹ameliaDEV26@gmail.com, ²luthfiAZIZAH.if@gmail.com, ³yuhana@if.its.ac.id

(Naskah masuk: 15 Januari 2021, diterima untuk diterbitkan: 15 Agustus 2022)

Abstrak

Estimasi biaya memiliki peran yang penting pada berjalannya proyek dan menjadi penentu keberhasilan suatu proyek perangkat lunak. Kegagalan estimasi biaya dalam perencanaan proyek perangkat lunak dapat menyebabkan proyek tidak berjalan dengan baik dan menimbulkan kerugian bagi perusahaan. Oleh karena itu, banyak peneliti sampai saat ini masih mencari dan melakukan penelitian untuk mendapatkan estimasi terbaik dengan memperhatikan faktor-faktor estimasi biaya. Tujuan penelitian ini adalah membuat *Systematic Literature Review* (SLR) yang berisi rangkuman dan analisis perkembangan penelitian terbaru tentang estimasi biaya pada perangkat lunak, khususnya pada metode yang digunakan serta faktor-faktor yang mempengaruhi. Penelitian ini berhasil mengkaji 21 penelitian dalam lima tahun terakhir (2015-2020) dan didapatkan 24 metode usulan yang terbagi menjadi tiga jenis metode yang sering digunakan dalam melakukan estimasi biaya perangkat lunak yaitu nonparametrik, parametrik dan semiparametrik. Selain itu, penelitian ini juga berhasil menemukan metode dan kombinasi metode terbaik berdasarkan ketepatan akurasi yaitu COCOMO II dan kombinasi *Genetic Algorithm* dengan *Artificial Bee Colony*, serta didapatkan lima faktor utama yang mempengaruhi estimasi biaya sehingga bisa berguna bagi para peneliti atau praktisi lain untuk mengembangkan estimasi biaya pada proyek perangkat lunak.

Kata kunci: Faktor Estimasi Biaya, Metode Estimasi Biaya, Perencanaan Proyek Perangkat Lunak

COST ESTIMATION METHOD ANALYSIS ON THE SOFTWARE AND AFFECTING FACTORS: A SYSTEMATIC LITERATURE REVIEW

Abstract

Cost estimation has an important role because it affects the project's progress and determines the success of a software project. Failure to estimate costs in software project planning can cause the project to not run well and cause losses to the company. Therefore, many researchers are still looking for and researching to get the best estimation by considering the cost estimation factors. The purpose of this study is to create a *Systematic Literature Review* (SLR) which contains a summary and analysis of the latest research developments on cost estimation in software, especially in the methods used and the factors that affect cost estimation. This study successfully reviewed 21 other studies in the last five years (2015-2020) and obtained 24 planning methods which are divided into three types of methods that are often used in conducting software cost research, namely nonparametric, parametric and semiparametric. Besides, this study also succeeded in finding the best method and combination of methods based on best accuracy, namely COCOMO II and the combination of *Genetic Algorithm* and *Artificial Bee Colony*, along with the five main factors that influence cost estimation so that it can be used by researchers or other practitioners to develop cost estimates for software projects.

Keywords: Cost Estimation Factor, Cost Estimation Method, Software Project Planning

1. PENDAHULUAN

Permasalahan dalam hal estimasi dan kegagalan pada proyek perangkat lunak diakibatkan karena kurangnya perencanaan, tidak menentukannya tujuan dari proyek tersebut serta kurangnya pengawasan terhadap tenggat waktu dan anggaran selama

mengimplementasikan proyek perangkat lunak (Suliman & Kadoda, 2017). Oleh karena itu, peran manajer proyek pengembangan perangkat lunak menjadi penting untuk mengatur segala aktivitas pengembangan perangkat lunak agar terhindar dari kegagalan proyek perangkat lunak. Untuk membantu

peran manajer proyek dalam mengembangkan proyek perangkat lunak, diperlukan perkiraan biaya yang akurat sebagai langkah untuk pengambilan keputusan yang tepat supaya tidak mengalami kerugian selama mengembangkan proyek perangkat lunak (Kholed Langsari et al., 2018).

Pada dasarnya, estimasi biaya menjadi bagian penting dalam pendapatan perusahaan karena pendapatan perusahaan itu bergantung pada bagaimana cara perusahaan tersebut mengestimasi biaya perangkat lunak, yang mana bisa berbeda-beda berdasarkan jenis produk perangkat lunak yang dihasilkan. Beberapa perusahaan perangkat lunak melakukan berbagai strategi dalam penentuan biaya yang sering berubah sesuai dengan produk, pasar, dan perilaku pelanggan (Nagacharan & Mary, 2017). Melebih-lebihkan biaya dapat menimbulkan kegagalan dalam penawaran produk perangkat lunak kepada pelanggan karena komitmen terhadap sumber daya menjadi berlebih sehingga membuat evaluasi kontrak menjadi tidak kompetitif jika dibandingkan dengan produk perangkat lunak pada perusahaan lain, sedangkan meremehkan biaya dapat menimbulkan pengelolaan sumber daya yang tidak baik sehingga membuat tuntutan terhadap anggaran yang harus fleksibel tetapi kualitas kinerja yang buruk dan pada akhirnya dapat mengakibatkan kegagalan dalam pengiriman produk perangkat lunak sesuai tenggat waktunya (Desai & Mohanty, 2018).

Oleh karena itu, cara mendapatkan estimasi biaya yang akurat dalam perencanaan proyek perangkat lunak menjadi salah satu masalah utama dan membuat banyak penelitian untuk mengatasi permasalahan tersebut dalam beberapa tahun terakhir ini (Baiquni et al., 2017). Ketepatan akurasi menjadi tujuan utama yang ingin diraih dalam melakukan estimasi biaya karena dengan estimasi biaya yang akurat dapat menjamin hubungan yang baik antara pelanggan dengan perusahaan perangkat lunak sehingga perusahaan perangkat lunak tidak kehilangan pendapatan dan posisinya dalam pasar perangkat lunak (Gandomani et al., 2019). Berdasarkan penelitian *Systematic Literature Review* yang telah dilakukan oleh (El Bajta et al., 2015), berhasil meringkas sembilan pertanyaan penelitian tentang estimasi biaya pada perangkat lunak yang mana didalamnya juga membahas tentang performa estimasi biaya, namun dalam penelitian tersebut belum menjelaskan lebih tentang faktor apa saja yang berpengaruh terhadap teknik estimasi biaya. Selanjutnya, penelitian yang dilakukan oleh (Qasim et al., 2018) berhasil menganalisis perkembangan estimasi biaya dengan menjawab dua pertanyaan penelitian berdasarkan 32 studi penelitian dalam kurun waktu 2009-2017, namun dalam penelitian tersebut juga belum membahas lebih tentang faktor yang berpengaruh terhadap estimasi biaya. Oleh karena itu, penelitian ini bertujuan untuk melakukan *Systematic Literature Review* (SLR) tentang analisis dan evaluasi beberapa studi primer pilihan, yang

dapat membantu merekomendasikan metode estimasi biaya atau kombinasi metode estimasi biaya terbaik berdasarkan ketepatan akurasi teknik tersebut beserta faktor-faktor yang paling banyak digunakan oleh penelitian-penelitian terkini dalam melakukan estimasi biaya.

2. METODE PENELITIAN

Dalam penelitian ini digunakan metode *systematic review*. Metode tersebut digunakan untuk evaluasi dan interpretasi dari beberapa penelitian relevan yang diakses dari berbagai macam database dengan tujuan agar dapat menjawab pertanyaan penelitian, memperoleh bahan materi pembelajaran, mencari peristiwa yang menarik atau untuk mencari celah dari penelitian sebelumnya yang berguna untuk penelitian yang baru (Kitchenham et al., 2006). Tahapan dalam melakukan SLR dibagi menjadi tiga, yaitu tahap perencanaan, tahap pelaksanaan, kemudian yang terakhir tahap pelaporan tinjauan pustaka. Pada tahap pertama yaitu tahap perencanaan, dilakukan proses identifikasi permasalahan perlunya melakukan penelitian *systematic review* (Wahono, 2007)(Izzati & Najwa, 2018). Tahapan tersebut sudah disajikan pada bagian 1 yaitu bagian pendahuluan. Untuk tahap kedua yaitu tahap pelaksanaan, terdapat beberapa tahap protokol review didalamnya, yaitu mencari, menyeleksi, mengekstrak dan menilai kualitas studi primer, kemudian dilakukan sintesis data berdasarkan studi primer tersebut, dengan tujuan agar pelaksanaan review lebih terarah dan menghindari bias pada peneliti (Wahono, 2007). Tahap kedua dalam melakukan SLR akan dijelaskan pada bagian metode penelitian ini, sedangkan untuk tahap ketiga yaitu tahap pelaporan tinjauan pustaka akan disajikan pada bagian hasil dan pembahasan.

2.1. Pertanyaan Penelitian

Penentuan pertanyaan penelitian dilakukan agar dalam pelaksanaan review pada penelitian ini menjadi lebih fokus. Pertanyaan penelitian dibuat dengan bantuan struktur PICO (*Population, Intervention, Comparison, Outcome*) seperti yang dilakukan oleh (Kitchenham et al., 2006) dan (Wahono, 2007) sebagai acuan referensi dalam melakukan penelitian SLR ini. PICO tersebut digunakan untuk merumuskan pertanyaan penelitian dan kata kunci pencarian studi primer tentang estimasi biaya. Tabel 1 dibawah ini menjelaskan struktur PICO yang digunakan.

Tabel 1. Struktur PICO yang Digunakan

PICO	Detail
<i>Population</i>	<i>Software</i>
<i>Intervention</i>	<i>Software Cost Estimation, Pricing Strategies</i>
<i>Comparison</i>	Tidak ada
<i>Outcome</i>	Menganalisis dan mengevaluasi metode estimasi biaya atau kombinasi metode estimasi biaya terbaik beserta faktor yang mempengaruhi estimasi biaya

Setelah menentukan struktur PICO, didapatkan beberapa pertanyaan penelitian dan tujuan dari setiap pertanyaan penelitian tersebut, seperti yang ditunjukkan pada Tabel 2.

Tabel 2. Pertanyaan Penelitian Beserta Tujuannya

ID	Pertanyaan Penelitian	Tujuan
RQ1	Jenis metode apa yang banyak digunakan pada estimasi biaya?	Menganalisis jenis metode yang sering digunakan dalam melakukan estimasi biaya
RQ2	Metode apa saja yang diusulkan dalam estimasi biaya?	Mengidentifikasi metode-metode yang sudah diusulkan dalam melakukan estimasi biaya
RQ3	Metode mana yang menghasilkan performa terbaik dalam estimasi biaya?	Mengidentifikasi metode estimasi biaya yang memiliki performa terbaik
RQ4	Faktor apa saja yang mempengaruhi estimasi biaya?	Menganalisis faktor yang berpengaruh terhadap estimasi biaya

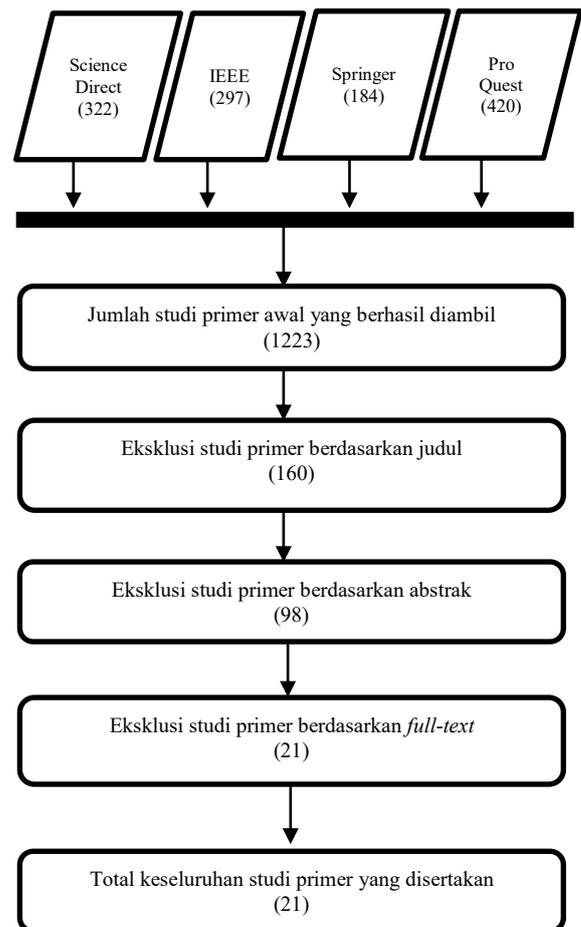
2.2. Strategi Pencarian

Dalam strategi pencarian studi primer tentang estimasi biaya, kata kunci untuk melakukan pencarian tersebut dilakukan dengan mengidentifikasi istilah dari PICO yang sudah dibuat, terlebih pada bagian *Population* dan *Intervention*, selanjutnya mengidentifikasi istilah yang ada pada pertanyaan penelitian serta mengidentifikasi kata-kata sinonim dari istilah pencarian. Berikut adalah kata kunci yang berhasil didefinisikan dengan menggabungkan setiap istilah yang didapat dengan operator logika supaya mendapatkan hasil pencarian yang relevan, yaitu “*Software Cost Estimation*” OR “*Pricing Strategies*” AND *Software*.

Berdasarkan kata kunci yang sudah didefinisikan, kemudian dilakukan pencarian pada beberapa *database* ilmiah digital yaitu Science Direct, IEEE, Springer dan ProQuest. Selama melakukan pencarian pada *database* tersebut, pencarian hanya dibatasi untuk makalah jurnal dan konferensi yang berbahasa Inggris serta diterbitkan antara tahun 2015 hingga 2020.

2.3. Strategi Seleksi Kriteria Inklusi dan Eksklusi

Pada proses seleksi studi primer dilakukan penentuan kriteria inklusi yaitu makalah lengkap jurnal dan konferensi, serta publikasi penelitian yang membahas tentang estimasi biaya dalam lingkup penelitian rekayasa perangkat lunak. Kriteria eksklusi yang ditentukan adalah publikasi non-ilmiah dan publikasi penelitian tentang estimasi biaya yang di luar lingkup penelitian ini. Hasil total pencarian studi primer berdasarkan proses seleksi kriteria inklusi dan eksklusi digambarkan pada Gambar 1 dibawah ini. Sebanyak 21 studi primer yang berhasil didapatkan dari seleksi kriteria inklusi dan eksklusi, kemudian studi primer pilihan tersebut akan dilakukan proses ekstraksi, penilaian kualitas dan sintesis data.



Gambar 1. Hasil total keseluruhan studi primer yang disertakan dalam penelitian

2.4. Strategi Ekstraksi, Penilaian Kualitas dan Sintesis Data

Studi primer yang berhasil diseleksi kemudian diekstraksi. Pada proses ekstraksi ini dirancang untuk mengumpulkan data dari studi primer pilihan guna menjawab pertanyaan penelitian. Terdapat dua properti yang berhasil diidentifikasi berdasarkan analisis dan pertanyaan penelitian yang dapat dilihat pada Tabel 3.

Tabel 3. Pemetaan Properti Ekstraksi Data Terhadap Pertanyaan Penelitian

Properti Ekstraksi Data	Pertanyaan Penelitian
Metode Estimasi Biaya Perangkat Lunak	RQ1, RQ2, RQ3
Faktor Estimasi Biaya Perangkat Lunak	RQ4

Dalam melakukan proses ekstraksi data tersebut, ditentukan kriteria penilaian kualitas studi primer pilihan melalui tiga pertanyaan dibawah ini. Berikut adalah kriteria pertanyaan penilaian kualitas dari studi primer pilihan:

- Apakah metode estimasi biaya pada rekayasa perangkat lunak dilaporkan?

- Apakah perhitungan akurasi dari metode estimasi biaya yang diusulkan dilaporkan?
- Apakah faktor yang mempengaruhi estimasi biaya dilaporkan?

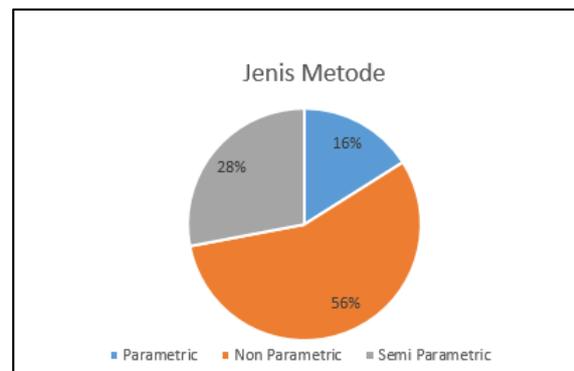
Penilaian kualitas studi primer pilihan digunakan sebagai acuan dalam menginterpretasikan temuan dari sintesis data sehingga dapat mendefinisikan kekuatan pada kesimpulan yang nantinya akan didapatkan (Wahono, 2007). Setelah melalui proses ekstraksi dan penilaian kualitas, didapatkan bahwa dari 21 studi primer pilihan, semuanya sudah memenuhi penilaian kualitas karena di dalam studi primer pilihan tersebut sudah menyertakan semua poin penting yang terdapat pada kriteria pertanyaan penilaian kualitas studi yaitu metode estimasi biaya perangkat lunak pada rekayasa perangkat lunak, perhitungan akurasi dari metode estimasi biaya yang diusulkan dan faktor yang mempengaruhi estimasi biaya. Setelah dilakukan penilaian kualitas pada 21 studi primer pilihan, kemudian mensintesis data agar dapat menjawab pertanyaan penelitian.

3. HASIL DAN PEMBAHASAN

3.1. Jenis Metode Pada Estimasi Biaya Perangkat Lunak

Berdasarkan 21 studi primer pilihan yang sudah didapatkan, jenis metode pada estimasi biaya perangkat lunak dibagi menjadi tiga yaitu nonparametrik, parametrik dan semiparametrik. Pada Gambar 2, jenis metode yang banyak digunakan adalah jenis metode nonparametrik. Jenis metode

nonparametrik adalah metode yang dihitung berdasarkan pada teknik *artificial intelligence* dan *machine learning* (Qasim et al., 2018). Penggunaan teknik *machine learning* pada estimasi biaya perangkat lunak banyak digunakan karena dapat membantu memberikan prediksi yang lebih akurat (BaniMustafa, 2018). Berdasarkan Tabel 4, yang termasuk dalam non-parametrik adalah algoritma BAT, Naïve Bayes, *Logistic Regression*, *Random Forest*, algoritma *Dolphin*, kombinasi *Genetic Algorithm* dan *Artificial Bee Colony*, algoritma *Flower Pollination*, *Particle Swarm Optimization*, *hybrid Genetic Algorithm* dan *Tabu Search*, *Chaotically Modified Particle Swarm Optimization – Dilation Erosion Perceptron* (CMPSO-DEP), kombinasi algoritma *Artificial Neural Network* (ANN) dan *Cuckoo Optimization Algorithm* (COA), kombinasi *Multi-Objective Particle Swarm Optimization* (MOPSO) dan *Fuzzy Logic*, kombinasi *Principal Component Analysis* dan *Constraint Solving*.



Gambar 2. Jenis metode pada estimasi biaya perangkat lunak

Tabel 4. Analisis metode yang diusulkan dan hasil MMRE setiap metode

Jenis Metode	Metode Usulan	MMRE	Paper
Parametrik	COCOMO II	6.27%	(M. Ullah et al., 2020)
	Bailey-Basili	20.9%	(M. Ullah et al., 2020)
	HMBDE	18%	(Singh & Kumar, 2017)
	Use Case Point (UCP)	36%	(Kurniawan et al., 2017)
Non-Parametrik	Algoritma BAT	49.54%	(Padmaja & Haritha, 2017)
	Naïve Bayes	17%	(BaniMustafa, 2018)
	Linier Regression	19%	(BaniMustafa, 2018)
	Random Forest	14%	(BaniMustafa, 2018)
	Hybrid algoritma <i>Dolphin</i> dan BAT	14.57%	(Fadhil et al., 2020)
	Kombinasi <i>Genetic Algorithm</i> dan <i>Artificial Bee Colony</i>	7.69%	(Gharehchopogh, Maleki, et al., 2015)
	<i>Flower Pollination Algorithm</i>	34.19%	(A. Ullah et al., 2019)
	<i>Particle Swarm Optimization</i> (PSO)	34.19%	(K Langsari & Sarno, 2017)
	Hybrid <i>Genetic Algorithm</i> dan <i>Tabu Search</i>	13.09%	(Gharehchopogh, Rezaei, et al., 2015)
	CMPSO-DEP	10.27%	(Bilgaiyan et al., 2018)
	Kombinasi ANN dan COA	14.8%	(Desai & Mohanty, 2018)
Kombinasi MOPSO dan <i>Fuzzy Logic</i>		11.89%	(Kholed Langsari et al., 2018)
	Kombinasi <i>Principal Component Analysis</i> dan pendekatan <i>Constraint Solving</i>	50.63%	(Garg & Gupta, 2015)
Semi Parametrik	Kombinasi <i>Neural Network</i> dengan <i>Fuzzy Logic</i> dan COCOMO II	18.27%	(Sarno et al., 2015b)
	Kombinasi ANN dan COCOMO	21.22%	(Sarno et al., 2015a)
	Kombinasi <i>Harmony Search</i> dan COCOMO	21%	(Sabbagh Jafari & Ziaaddini, 2016)
	Kombinasi <i>Artificial Bee Colony</i> dan COCOMO II	60.82%	(Pratama et al., 2017)
	Kombinasi <i>Fuzzy Logic</i> dan COCOMO	15%	(Chhabra & Singh, 2016)
	Kombinasi <i>Fuzzy Logic</i> dengan <i>Local Calibration</i> dan COCOMO	73.41%	(Baiquni et al., 2017)
Kombinasi FPA dan COCOMO	25%	(Ahmad & Samat, 2018)	

Jenis metode estimasi biaya perangkat lunak parametrik adalah metode perhitungan matematis yang didasarkan pada data historis (Qasim et al., 2018). Ada empat metode usulan yang termasuk dalam jenis metode parametrik yaitu COCOMO II, *Bailey-Basili*, *Homeostasis Mutation Based Differential Evolution* (HMBDE), dan *Use Case Point* (UCP). Dan yang terakhir, jenis metode semi parametrik adalah gabungan dari metode parametrik dan metode non parametrik (Qasim et al., 2018). Berdasarkan pada Tabel 4, beberapa metode semi parametrik yang diusulkan adalah kombinasi antara metode nonparametrik yang berupa *machine learning* atau *artificial intelligence* dan metode parametrik yang berupa *Constructive Cost Model* (COCOMO) saja, dimana tiga metode usulan lainnya pada metode parametrik seperti *Bailey-Basili*, HMBDE dan UCP jarang digunakan. Ada 7 metode yang termasuk dalam semiparametrik yaitu kombinasi ANN dan COCOMO, kombinasi algoritma *Harmony Search* dan COCOMO, kombinasi algoritma *Neural Network* dengan *Fuzzy Logic* dan COCOMO II, kombinasi *Artificial Bee Colony* dan COCOMO II, kombinasi *Fuzzy Logic* dan COCOMO, kombinasi *Fuzzy Logic* dengan *Local Calibration* dan COCOMO II, kombinasi FPA dan COCOMO.

3.2. Metode Pada Estimasi Biaya Perangkat Lunak

Berdasarkan Tabel 4, terdapat 24 metode usulan yang didapatkan dari 21 studi primer pilihan karena ada beberapa penelitian yang mengusulkan lebih dari satu algoritma dalam satu penelitian untuk melakukan estimasi biaya pada perangkat lunak. Metode COCOMO adalah metode estimasi biaya perangkat lunak yang paling banyak digunakan. Metode tersebut dirumuskan dengan menggunakan regresi dasar dan menggunakan beberapa parameter variabel yang nantinya akan dikalibrasi berdasarkan pertimbangan karakteristik proyek yang akan dikerjakan (Sharma & Purohit, 2018), sedangkan metode COCOMO II adalah suatu metode pengembangan dari metode COCOMO yang digunakan untuk estimasi proyek perangkat lunak di zaman modern. Metode tersebut memberikan pilihan untuk melakukan kalibrasi terhadap faktor-faktor umum pada proyek modern. Selain itu, metode COCOMO II juga dapat menghitung jumlah upaya yang dikeluarkan dalam memperbarui kode yang dapat digunakan ulang (Sharma & Purohit, 2018). Metode *Bailey-Basili* adalah salah satu dari beberapa model estimasi yang menggunakan perhitungan persamaan matematika. Perhitungan COCOMO II dan *Bailey Basili* dihitung dengan nilai *Line of Code* (LOC) dan faktor lainnya seperti *Effort Multiplier* dan *Scale Factor* (M. Ullah et al., 2020). Kemudian, metode *Homeostasis Mutation Based Differential Evolution* (HMBDE) adalah sebuah algoritma yang digunakan untuk mengembangkan parameter dari

metode COCOMO II. HMBDE dihitung dengan memodifikasi parameter COCOMO II dan menambahkan perhitungan vektor (Singh & Kumar, 2017). Selanjutnya, *Use Case Point* adalah sebuah metode algoritma untuk menghitung estimasi dengan menggunakan kompleksitas dari interaksi antara sistem dan aktor. *Use Case Point* (UCP) dihitung dengan mempertimbangkan kompleksitas *use case*, kompleksitas aktor, faktor kompleksitas teknis, dan faktor lingkungan dengan menggunakan 100 data historis dari proyek perangkat lunak (Kurniawan et al., 2017).

Metode BAT adalah suatu algoritma optimasi heuristik yang terinspirasi dari perilaku ekolokasi dari kelelawar (Padmaja & Haritha, 2017). Metode BAT tersebut digunakan untuk meningkatkan akurasi saat melakukan estimasi biaya perangkat lunak dengan menemukan nilai faktor dari setiap proyek yang terlibat yang kemudian dihitung nilai *fitness value* secara keseluruhan. Selanjutnya, pada metode *Dolphin*, ide yang dilakukan pada metode tersebut hampir sama dengan metode BAT, yang terinspirasi dari perilaku ekolokasi suatu hewan. Metode *Dolphin* digunakan untuk mendapatkan nilai koefisien yang baik sehingga mampu mengoptimasi metode COCOMO II dalam memprediksi estimasi biaya perangkat lunak (Fadhil et al., 2020). Penelitian yang dilakukan oleh (BaniMustafa, 2018) mengusulkan 3 metode *machine learning* yaitu *Naïve Bayes*, *Logistic Regression* dan *Random Forest*. Metode *Naïve Bayes* merupakan klasifikasi *supervised learning* dengan menggunakan teknik probabilitas, *Logistic Regression* adalah algoritma klasifikasi yang mencari hubungan dari variabel dependent dan variabel independent yang diberikan. *Random Forest* merupakan algoritma *supervised learning* yang digunakan untuk klasifikasi dan regresi. Dalam hal ini, ketiga metode tersebut, baik metode *Naïve Bayes*, *Logistic Regression* atau *Random Forest Classifier* digunakan untuk menyelidiki validitas teknik *machine learning* menggunakan *data mining* dalam melakukan estimasi biaya perangkat lunak sebagai alternatif dari penggunaan metode COCOMO. Selanjutnya, pada metode *Flower Pollination*, seperti namanya, metode tersebut merupakan metode yang terinspirasi dari proses penyerbukan pada bunga (A. Ullah et al., 2019). Di dalam metode tersebut algoritma *Flower Pollination* digunakan untuk mengoptimalkan koefisien dari metode COCOMO II dalam melakukan estimasi biaya perangkat lunak.

Metode *Particle Swarm Optimization* (PSO) merupakan salah satu algoritma optimasi heuristik yang telah banyak digunakan karena daya tahan, kesederhanaan serta fleksibilitas dari algoritma tersebut (Kholed Langsari et al., 2018). Penggunaan PSO dalam penelitian (K Langsari & Sarno, 2017) bertujuan untuk mengoptimasi parameter metode COCOMO II dengan menghasilkan waktu eksekusi yang jauh lebih sedikit. Selanjutnya, pada metode

Chaotically Modified Particle Swarm Optimization – Dilation Erosion Perceptron (CMPSO-DEP), dilakukan pendekatan yang berfokus pada kerangka morfologi matematika berbasis *hybrid artificial neuron*, atau disebut dengan *dilation erosion perceptron* (DEP), dengan fondasi aljabar dalam teori kisi lengkap atau *Complete Lattice Theory* (CLT). Dalam hal ini, *Chaotically Modified Particle Swarm Optimization* (CMPSO) digunakan untuk mengoptimalkan parameter DEP perceptron, dan metode yang diusulkan yaitu CMPSO-DEP dilakukan untuk memperkirakan biaya pada pengembangan perangkat lunak secara akurat dengan memperoleh nilai MMRE sebesar 10.27% seperti yang ditunjukkan pada Tabel 4, yang mana nilai tersebut didapatkan dengan menggunakan dataset COCOMO (Bilgaiyan et al., 2018).

Selanjutnya, penelitian yang dilakukan oleh (Gharehchopogh, Rezaii, et al., 2015) melakukan penggabungan dua metode optimasi heuristik yaitu *Genetic Algorithm* dan *Tabu Search*. Kedua algoritma tersebut digunakan untuk mendapatkan nilai parameter konstan yang optimal dari model COCOMO, karena berdasarkan penjelasan dari penelitian tersebut, salah satu masalah dari model COCOMO adalah penentuan nilai optimal untuk beberapa parameter yang ada. Dengan demikian, penggabungan dua metode optimasi heuristik tersebut dapat memperoleh hasil yang lebih efektif dan memiliki nilai MMRE sebesar 13.09% seperti terlihat pada Tabel 4. Nilai MMRE tersebut didapatkan dengan menggunakan dataset NASA(60) (Gharehchopogh, Rezaii, et al., 2015). Metode *Artificial Neural Network* (ANN) adalah metode yang terinspirasi dari neuron pada otak manusia, yang dapat membuat sistem komputasi lebih cepat dan lebih berguna daripada sistem terdahulu yang sudah ketinggalan zaman, serta dapat memodelkan hampir semua fungsi atau tugas berbeda seperti klasifikasi, perkiraan dan deteksi pola. Metode *Cuckoo Optimization Algorithm* (COA) adalah algoritma optimasi heuristik yang terinspirasi dari sifat bertelur burung kukuk. Berdasarkan penelitian yang dilakukan oleh (Desai & Mohanty, 2018) yang melakukan kombinasi antara metode ANN dan COA dengan menggunakan dataset ISBSG, dimana pemilihan parameter terbaik dilakukan oleh algoritma COA untuk mengoptimalkan metode ANN dan estimasi biaya perangkat lunak dilakukan dengan menggunakan algoritma ANN untuk mendapatkan hasil yang relatif baik.

Metode *Fuzzy Logic* adalah metode pembuat keputusan berdasarkan beberapa aturan-aturan tertentu yang dibuat untuk menyelesaikan masalah ketidakakuratan dalam suatu proses pengukuran (Sarno et al., 2015b)(Baiquni et al., 2017). Berdasarkan penelitian yang dilakukan oleh (Baiquni et al., 2017) yang menggunakan *Fuzzy Logic* dan *Local Calibration* untuk meningkatkan akurasi dari model COCOMO II. Penggunaan *Fuzzy Logic*

dengan *Gaussian Membership Functions* (GMF) diterapkan untuk mendesain ulang *Effort Multiplier* (EM), serta penggunaan *Local Calibration* membuat akurasi COCOMO II menjadi lebih baik dengan menggunakan nilai parameter pengali dan eksponen untuk persamaan perhitungan estimasi usaha pada dataset, yang nantinya akan digunakan sebagai input untuk melakukan perhitungan kalibrasi parameter dari metode COCOMO II. Selanjutnya, penelitian yang dilakukan oleh (Garg & Gupta, 2015) melakukan kombinasi *Principal Component Analysis* dan pendekatan *Constraint Solving* untuk meningkatkan ketepatan pada estimasi biaya perangkat lunak. *Principal Component Analysis* digunakan untuk mengurangi dimensi pada atribut yang dibutuhkan dan untuk mengidentifikasi kunci atribut yang memiliki nilai korelasi maksimal terhadap biaya pengembangan perangkat lunak, sedangkan pendekatan *Constraint Solving* digunakan untuk memenuhi kriteria yang ditentukan oleh *agile manifesto*. Kombinasi *Principal Component Analysis* dan pendekatan *Constraint Solving* terbukti cocok diterapkan untuk proyek pengembangan perangkat lunak *agile*.

3.3. Metode Berperforma Baik Pada Estimasi Biaya Perangkat Lunak

Pada Tabel 4 menunjukkan hasil perhitungan nilai eror menggunakan *Mean Magnitude of Relative Error* (MMRE). MMRE adalah ukuran relatif dari perbedaan antara nilai estimasi biaya perangkat lunak yang sebenarnya dan nilai estimasi biaya perangkat lunak yang diperkirakan (Garg & Gupta, 2015). Setiap metode yang diusulkan telah dihitung nilai MMRE dalam bentuk persentase. Semakin kecil nilai MMRE, maka performa metode yang diusulkan semakin bagus (Qasim et al., 2018). Berdasarkan Tabel 4, didapatkan bahwa persentase MMRE paling kecil dihasilkan oleh metode COCOMO II dengan nilai MMRE sebesar 6.27%. Nilai MMRE tersebut didapatkan dengan melakukan perbandingan nilai MMRE antara metode COCOMO II dengan metode Bailey-Basili pada penelitian (M. Ullah et al., 2020). Hal tersebut menunjukkan bahwa performa dari metode COCOMO II lebih baik dibandingkan dengan metode Bailey-Basili. Penggunaan COCOMO II pada penelitian tersebut digunakan untuk menghitung estimasi biaya pada *Person-Month* (PM) dan estimasi waktu. Dataset yang digunakan adalah dataset industri perangkat lunak Turki dan NASA(93) dengan mempertimbangkan *Scale Factor* (SF).

Persentase MMRE terkecil kedua adalah metode kombinasi dari *Genetic Algorithm* (GA) dan *Artificial Bee Colony* (ABC) dengan nilai MMRE 7,69%. Nilai MMRE tersebut lebih rendah dibandingkan dengan nilai MMRE dari metode pembanding lain yaitu metode COCOMO. Kombinasi dari metode *Genetic Algorithm* (GA) dan *Artificial Bee Colony* (ABC) digunakan untuk mendapatkan nilai akurasi tinggi saat melakukan

estimasi biaya dengan menggunakan dataset NASA(60) dan mempertimbangkan faktor *Effort Multiplier* (EM).

Hasil nilai MMRE yang ditunjukkan pada Tabel 4 tentu saja tidak bisa dibandingkan secara langsung karena nilai tersebut didasarkan pada faktor-faktor yang berbeda, seperti dataset yang digunakan, ukuran kode atau faktor lainnya. Namun, nilai eror atau MMRE yang ditunjukkan pada Tabel 4 dapat dijadikan acuan penelitian untuk melakukan penelitian lebih lanjut, dengan menyesuaikan faktor-faktor yang telah digunakan pada acuan penelitian tersebut.

3.4. Faktor yang Mempengaruhi Estimasi Biaya Perangkat Lunak

Berdasarkan Tabel 5, ada lima faktor umum yang mempengaruhi estimasi biaya yaitu *Line of Code* (LOC), *Function Point* (FP), *Effort Multiplier* (EM), *Scale Faktor* (SF) dan *historical data*. Pemilihan lima faktor umum tersebut didasarkan pada mayoritas 21 studi primer pilihan telah menggunakan faktor-faktor tersebut. *Line of Code* (LOC) adalah input dasar dari perhitungan estimasi biaya yang berisi baris kode dari proyek perangkat lunak dan berfungsi untuk mengukur besarnya perangkat lunak. LOC ini digunakan dalam perhitungan pada metode COCOMO, yang tidak termasuk baris komentar dan baris kosong (M. Ullah et al., 2020). *Function Point* (FP) digunakan untuk mengukur sebuah perangkat lunak berdasarkan pada data proyek sebelumnya dan fungsionalitas program. *Function Point* sendiri terbagi menjadi lima komponen yaitu *External Input* yang merupakan fungsi untuk mengontrol jenis masukan dari pengguna, *External Inquiry* berupa fungsi masukan interaktif yang membutuhkan tanggapan, *External Output* adalah jenis data keluaran ke pengguna yang meninggalkan sistem, *Internal Logical File* merupakan *file* untuk digunakan dan dibagikan di dalam sistem serta *External Interface File* adalah *file* yang dikirimkan atau dibagikan antar sistem. Masing-masing dari komponen *Function Point* tersebut akan dikelompokkan menurut tingkat kerumitannya (Maqдум et al., 2019). Kemudian, pada *Effort Multiplier* (EM) terbagi menjadi 4 jenis atribut yaitu pertama, atribut produk yaitu terdiri dari *Required Software Reliability*, *Database Size*, *Product Complexity*, *Developed for Reusability*, *Documentation Match to Life Cycle Need*. Kedua, atribut komputer yang terdiri dari *Execution Time Constraint*, *Main Storage Constraint* dan *Platform Volatility*. Ketiga adalah atribut personal terdiri dari *Analyst capability*, *Programmer Capability*, *Personnel Continuity*, *Application Experience*, *Platform Experience* dan *Language and Tool Experience*. Keempat adalah atribut proyek yang terdiri dari *Use of Software Tools*, *Multisite Development* dan *Required Development Schedule* (Sarno et al., 2015b). Pada penelitian (Chhabra &

Singh, 2016) tidak menggunakan semua jenis dari *Effort Multiplier*, tetapi hanya menggunakan jenis *Personnel Attributes* saja. Selanjutnya, *Scale Faktor* (SF) adalah proses perhitungan untuk menentukan karakteristik serta usaha dari suatu proyek yang sedang berjalan (Maqдум et al., 2019). *Scale Faktor* terdiri dari lima atribut yaitu *Precedentedness* (PREC), *Development Flexibility* (FLEX), *Architecture / Risk Resolution* (RESL), *Team Cohesion* (TEAM), *Process Maturity* (PMAT) (Sarno et al., 2015b). Dan faktor terakhir adalah data proyek software sebelumnya atau *historical data* yang digunakan sebagai acuan estimasi biaya.

Tabel 5. Analisis Faktor yang Mempengaruhi Estimasi Biaya Perangkat Lunak

Faktor	Paper
<i>Line Of Code</i> (LOC)	(Padmaja & Haritha, 2017), (Fadhil et al., 2020), (M. Ullah et al., 2020), (Singh & Kumar, 2017), (Desai & Mohanty, 2018), (Sabbagh Jafari & Ziaaddini, 2016), (Pratama et al., 2017), (Kholed Langsari et al., 2018)
<i>Function Point</i> (FP)	(Padmaja & Haritha, 2017), (Sarno et al., 2015a), (Garg & Gupta, 2015)
<i>Effort Multiplier</i> (EM)	(Gharehchopogh, Maleki, et al., 2015), (Sarno et al., 2015b), (K Langsari & Sarno, 2017), (Baiquini et al., 2017), (Chhabra & Singh, 2016), (Bilgaiyan et al., 2018), (Ahmad & Samat, 2018), (Pratama et al., 2017), (K Langsari & Sarno, 2017)
<i>Scale Faktor</i> (SF)	(Sarno et al., 2015b), (K Langsari & Sarno, 2017), (Pratama et al., 2017), (Kholed Langsari et al., 2018)
<i>Historical Data</i>	(M. Ullah et al., 2020), (Singh & Kumar, 2017)

4. KESIMPULAN

Berdasarkan 21 penelitian yang diperoleh dari proses seleksi dan didasarkan pada empat pertanyaan penelitian, didapatkan 24 metode usulan, yang mana 24 metode usulan tersebut dibagi menjadi tiga jenis metode yaitu nonparametrik, parametrik, dan semiparametrik. Jenis metode yang banyak digunakan adalah jenis metode nonparametrik, yaitu sebanyak 13 metode usulan, yang didasarkan pada teknik *artificial intelligence* dan *machine learning*. Selain itu, dari 24 metode usulan tersebut, mayoritas bertujuan untuk mencari metode dengan ketepatan akurasi tinggi yaitu yang memiliki nilai eror rendah dalam melakukan estimasi biaya perangkat lunak. Hal tersebut dilakukan dengan mengoptimasi parameter metode COCOMO atau COCOMO II yang dikombinasikan dengan metode-metode optimasi heuristik, beberapa diantaranya adalah Genetic

Algorithm, PSO, BAT, Dolphin dan Flower Pollination.

Perhitungan nilai eror yang banyak digunakan adalah *Mean Magnitude of Relative Error* (MMRE). Apabila nilai persentase MMRE suatu metode semakin kecil, maka performanya semakin bagus. Didapatkan persentase MMRE paling kecil dengan menggunakan metode COCOMO II dan presentase MMRE terkecil kedua adalah metode kombinasi dari *Genetic Algorithm* (GA) dan *Artificial Bee Colony* (ABC). Selain itu, terdapat lima faktor umum yang mempengaruhi estimasi biaya yaitu *Line of Code* (LOC), *Function Point* (FP), *Effort Multiplier* (EM), *Scale Faktor* (SF) dan *historical data*. Pemilihan lima faktor umum tersebut didasarkan pada mayoritas 21 studi primer pilihan telah menggunakan faktor-faktor tersebut. Setiap metode usulan memiliki faktor-faktor yang berbeda yang disesuaikan dengan kebutuhan penelitian.

Paper SLR ini masih belum bisa dibandingkan berdasarkan nilai eror MMRE karena setiap metode usulan dalam penelitian memiliki faktor yang berbeda-beda. Untuk penelitian selanjutnya bisa difokuskan pada salah satu jenis metode atau kesamaan faktor pada metode usulan.

DAFTAR PUSTAKA

- AHMAD, S. F., & SAMAT, P. A., 2018. Extraction Cost of Quality and Testing in Software Project. 2018 IEEE Conference on E-Learning, e-Management and e-Services (IC3e), 109–115.
- BAIQUNI, M., SARNO, R., SARWOSRI, & SHOLIQ., 2017. Improving the accuracy of COCOMO II using fuzzy logic and local calibration method. 2017 3rd International Conference on Science in Information Technology (ICSITech), 284–289.
- BANIMUSTAFA, A., 2018. Predicting Software Effort Estimation Using Machine Learning Techniques. 2018 8th International Conference on Computer Science and Information Technology (CSIT), 249–256.
- BILGAIYAN, S., ADITYA, K., MISHRA, S., & DAS, M., 2018. A Swarm Intelligence based Chaotic Morphological Approach for Software Development Cost Estimation. *International Journal of Intelligent Systems and Applications*, 11(9), 13.
- CHHABRA, S., & SINGH, H., 2016. Simulink based fuzzified COCOMO. 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I), 847–851.
- DESAI, V. S., & MOHANTY, R., 2018. ANN-Cuckoo Optimization Technique to Predict Software Cost Estimation. 2018 Conference on Information and Communication Technology (CICT), 1–6.
- EL BAJTA, M., IDRI, A., FERNÁNDEZ-ALEMÁN, J. L., ROS, J. N., & TOVAL, A., 2015. Software cost estimation for global software development a systematic map and review study. 2015 International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE), 197–206.
- FADHIL, A. A., ALSARRAJ, R. G. H., & ALTAIE, A. M., 2020. Software Cost Estimation Based on Dolphin Algorithm. *IEEE Access*, 8, 75279–75287.
- GANDOMANI, T. J., FARAJI, H., & RADNEJAD, M., 2019. Planning Poker in cost estimation in Agile methods: Averaging Vs. Consensus. 2019 5th Conference on Knowledge Based Engineering and Innovation (KBEI), 66–71.
- GARG, S., & GUPTA, D., 2015. PCA based cost estimation model for agile software development projects. 2015 International Conference on Industrial Engineering and Operations Management (IEOM), 1–7.
- GHAREHCHOPOGH, F. S., MALEKI, I., & TALEBI, A., 2015. Using hybrid model of Artificial Bee Colony and Genetic Algorithms in Software Cost Estimation. 2015 9th International Conference on Application of Information and Communication Technologies (AICT), 102–106.
- GHAREHCHOPOGH, F. S., REZAI, R., & ARASTEH, B., 2015. A new approach by using Tabu search and genetic algorithms in Software Cost estimation. 2015 9th International Conference on Application of Information and Communication Technologies (AICT), 113–117.
- IZZATI, A. N., & NAJWA, N. F., 2018. Pengaruh Stakeholder Perspective Dalam Penerapan ERP: A Systematic Literature Review. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 5(1), 41.
- KITCHENHAM, B., MENDES, E., & TRAVASSOS, G. H., 2006. A Systematic Review of Cross- vs. Within-Company Cost Estimation Studies.
- KURNIAWAN, I., ARMAN, A. A., & MARDIYANTO, S., 2017. Development of analogy-based estimation method for software development cost estimation in government agencies. 2017 6th International Conference on Electrical Engineering and Informatics (ICEEI), 1–6.
- LANGSARI, K., & SARNO, R., 2017. Optimizing COCOMO II parameters using particle swarm method. 2017 3rd International Conference on Science in Information Technology (ICSITech), 29–34.
- LANGSARI, KHOLED, SARNO, R., & SHOLIQ., 2018. Optimizing Effort Parameter of

- COCOMO II Using Particle Swarm Optimization Method. *TELKOMNIKA*, 16(5), 2208–2216.
- MAQDUM, A. N., PERDANAKUSUMA, A. R., & PUTRA, W. H. N., 2019. Implementasi Metode COCOMO II untuk Estimasi Biaya Pengembangan Perangkat Lunak di CV . *Profile Image Studio*. 3(6).
- NAGACHARAN, J. G., & MARY, R. R., 2017. Recommendations for Software Pricing Strategies in Different Business Contexts. *International Journal of Research in Engineering, IT and Social Sciences*, 7(3), 29–33.
- PADMAJA, M., & HARITHA, D., 2017. Software Effort Estimation using Meta Heuristic Algorithm. *International Journal of Advanced Research in Computer Science*, 8(5), 196–201.
- PRATAMA, R. Y., SARNO, R., & SHOLIQU., 2017. Optimizing COCOMO II parameters using artificial bee colony method. 2017 11th International Conference on Information Communication Technology and System (ICTS), 125–130.
- QASIM, I., TUFAIL, H., & FATIMA, A., 2018. Cost Estimation Techniques for Software Development: A Systematic Literature Review. *International Conference on Engineering, Computing & Information Technology (ICECIT 2017)*, Icecit 2017, 38–42.
- SABBAGH JAFARI, S. M., & ZIAADDINI, F., 2016. Optimization of software cost estimation using harmony search algorithm. 2016 1st Conference on Swarm Intelligence and Evolutionary Computation (CSIEC), 131–135.
- SARNO, R., SIDABUTAR, J., & SARWOSRI., 2015a. Comparison of different Neural Network architectures for software cost estimation. 2015 International Conference on Computer, Control, Informatics and Its Applications (IC3INA), 68–73.
- SARNO, R., SIDABUTAR, J., & SARWOSRI., 2015b. Improving the accuracy of COCOMO's effort estimation based on neural networks and fuzzy logic model. 2015 International Conference on Information Communication Technology and Systems (ICTS), 197–202.
- SHARMA, B., & PUROHIT, R., 2018. Review of current software estimation techniques. *Communications in Computer and Information Science*, 799, 380–399.
- SINGH, S. P., & KUMAR, A., 2017. Software cost estimation using homeostasis mutation based differential evolution. 2017 11th International Conference on Intelligent Systems and Control (ISCO), 173–181.
- SULIMAN, S. M. A., & KADODA, G., 2017. Factors that influence software project cost and schedule estimation. 2017 Sudan Conference on Computer Science and Information Technology (SCCSIT), 1–9.
- ULLAH, A., WANG, B., SHENG, J., LONG, J., ASIM, M., & RIAZ, F., 2019. A Novel Technique of Software Cost Estimation Using Flower Pollination Algorithm. 2019 International Conference on Intelligent Computing, Automation and Systems (ICICAS), 654–658.
- ULLAH, M., ALI, R., ABDULLAH, AHMAD, M., KHAN, T., & MULK, F. U., 2020. Software Cost Estimation – A Comparative Study of COCOMO-II and Bailey-Basili Models. 2019 International Conference on Advances in the Emerging Computing Technologies (AECT), 1–5.
- WAHONO, R. S., 2007. A Systematic Literature Review of Software Defect Prediction: Research Trends, Datasets, Methods and Frameworks. *Journal of Software Engineering*, 1(1), 1–16.

Halaman ini sengaja dikosongkan