



---

**STUDI LITERATUR PERBANDINGAN ANTARA METODE *LOC*, *COCOMO*, *FPA* DALAM RANAH *SOFTWARE METRIC***

**Rizky Parluka<sup>1</sup>, Devan Cakra Mudra Wijaya<sup>2</sup>, Heri Khariono<sup>3</sup>, Rifky Akhmad Fernanda<sup>4\*</sup>**

<sup>1234</sup>Informatika, FIK, Universitas Pembangunan Nasional Veteran Jawa, Indonesia

\*email: rifkyakhmad911@gmail.com

**Received: 8 April 2020 Accepted: 20 Juni 2020 Published: 30 Juni 2020**

**Abstrak**

Pengukuran perangkat lunak (*software metric*) adalah tindakan mengevaluasi dan menilai pelaksanaan kerangka kerja atau seperangkat instrumen untuk memantau dan menentukan tingkat produktivitas perangkat lunak berbasis pada pendekatan kuantitatif. Pengukuran perangkat lunak membantu dalam kemajuan program untuk mengesahkan kualitas barang, survei kualitas individu termasuk dalam pembuatan produk, mengevaluasi manfaat dari penggunaan strategi modern dan instrumen, dan sebagai dasar untuk membuat alat pengukur yang memiliki kegunaan yang berbeda. Estimasi proyek perangkat lunak yang akurat ditentukan oleh tingkat manajerial perangkat lunak yang mana telah memperkirakan ukuran perangkat lunak dengan benar. Perkiraan yang tepat dari ukuran merupakan hal penting dalam menghitung perkiraan biaya proyek, usaha, waktu dan durasi karena menyediakan informasi yang diperlukan untuk pengembangan perangkat lunak. Makalah ini memberikan pendalaman materi dari tiga strategi pengukuran perangkat lunak yang paling umum digunakan yaitu *Lines Of Code (LOC)*, *Constructive Cost Model (COCOMO)*, *Function Point Analysis (FPA)* dari beberapa referensi yang penulis gunakan.

**Kata kunci:** *software metric, LOC, FPA, COCOMO*, perangkat lunak

**Abstract**

*Software measurement (Metric software) is the act of evaluating and assessing the implementation of the framework or a set of instruments to monitor and determine the level of productivity of the software based on a quantitative approach. Software measurement helps in the advancement of the program to validate the quality of goods, the individual quality surveys are included in the manufacture of the product, evaluating the benefits of the use of modern strategies and instruments, and as a basis for making tools a gauge that has different uses. Accurate software project estimation is determined by the managerial level of the software to which it has predicted the size of the software correctly. The right estimate of size is important in calculating project costs, effort, time and duration because it provides the information needed for software development. This paper provides the deepening of the material of the three most commonly used software measurement strategies of Lines of Code (LOC), Constructive Cost Model (COCOMO), Function Point Analysis (FPA) from some references whose authors uses.*

**Keywords:** *software metric, LOC, FPA, cocomo, software*

**How to cite (in APA style):** Parluka, R., Wijaya, D. C. M., Khariono, H., & Fernanda, R. A. (2020). Studi literatur perbandingan antara metode LOC, COCOMO, FPA dalam ranah software metric. *Jurnal Pendidikan Informatika dan Sains*, 9(1), 66-74

Copyright © 2020 Rizky Parluka, Devan Cakra Mudra Wijaya, Heri Khariono, Rifky Akhmad Fernanda  
DOI: 10.31571/saintek.v9i1.1697



## **PENDAHULUAN**

Sudah diketahui bahwa gaya hidup dan kerja pada era saat ini secara signifikan didorong oleh perkembangan teknologi informasi (Sharma, Sharma and Sakpal, 2016). Salah satu faktor yang mendukung perkembangan teknologi informasi adalah pengembangan perangkat lunak yang merupakan pendekatan secara sistematis pada rekayasa perangkat lunak dalam membangun dan memelihara sistem perangkat lunak (Langsari and Sarno, 2017). Paradigma pengembangan perangkat lunak yang baru secara terstruktur, modular, berorientasi objek, dan aspek-aspek pendekatan lainnya yang berkaitan digunakan untuk mengakomodasi dan mengeksplorasi aplikasi perangkat lunak sejauh mungkin (Sophsathit, 2017).

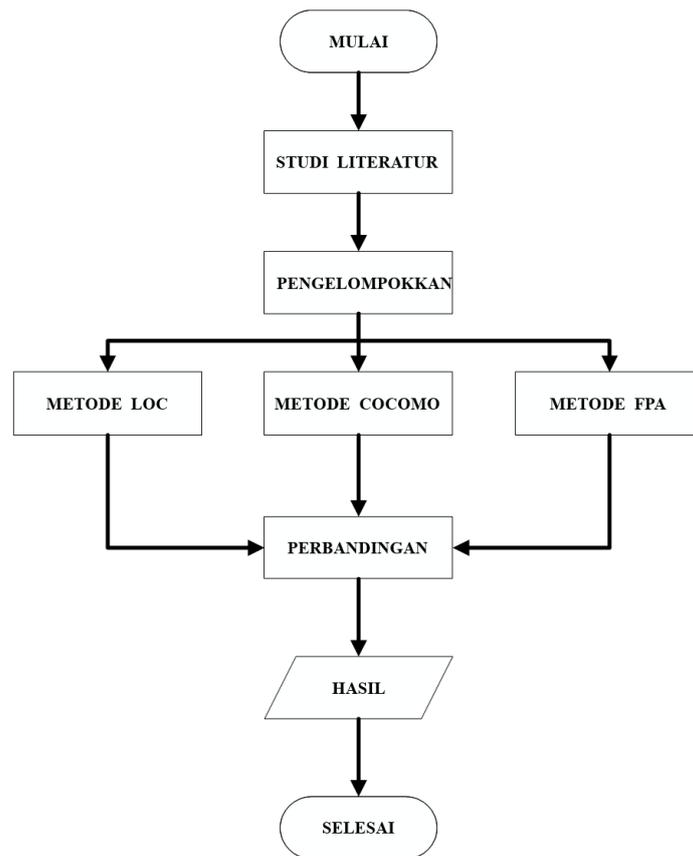
Perusahaan perangkat lunak komputer telah menggunakan model program yang khas untuk melakukan pengukuran. Ada sejumlah model estimasi biaya rekayasa perangkat lunak seperti “Metode analisis upaya, *COCOMO*, *COSYSMO*, analisis titik fungsi, dan banyak model lainnya” (Shad and Bahadar, 2017). Metrik perangkat lunak menawarkan dasar kuantitatif untuk memprediksi proses pengembangan perangkat lunak. Kualitas perangkat lunak harus dicapai untuk memuaskan pelanggan dengan mengurangi biaya perangkat lunak dan meningkatkan tanggung jawab produk perangkat lunak (Rashid, Mahmood and Nisar, 2019). Kualitas dan estimasi biaya perangkat lunak merupakan tantangan utama pada ekspansi perangkat lunak (Langsari and Sarno, 2017).

Keakuratan estimasi sangat penting dalam pembuatan manajemen yang baik untuk perluasan perangkat lunak (Langsari and Sarno, 2017). Pengukuran biasanya dilakukan pada pengembangan perangkat lunak dengan menggunakan metode metrik perangkat lunak (Bukhari, Yahaya and Deraman, 2015). Alasan dilakukan pengukuran yaitu sebagai karakterisasi, evaluasi, prediksi, dan peningkatan perangkat lunak. Metrik telah didefinisikan untuk mengukur kelas pada tingkat desain maupun kode sumber (Chahal and Singh, 2010).

Pada makalah ini memberikan pendalaman materi dari tiga strategi pengukuran perangkat lunak yang paling umum digunakan yaitu *Lines Of Code* (LOC), *Constructive Cost Model* (COCOMO), *Function Point Analysis* (FPA) dari beberapa referensi yang penulis gunakan. Adapun tujuan dari pengkajian ini yaitu untuk mengetahui lebih dalam tentang aspek - aspek yang dapat mempengaruhi kinerja suatu proyek perangkat lunak. Perbandingan adalah salah satu langkah yang dapat dijadikan solusi untuk memperoleh data secara ringkas dan tepat guna. Perbandingan ini berorientasi pada subjek, objek, serta fungsionalitas terhadap aspek yang ada pada masing-masing metode yang dikaji dengan memprioritaskan keunggulan efektifitas atau efisiensi dalam hal kenyamanan implementasi di suatu proyek.

## **METODE**

Metode penelitian yang digunakan dalam penelitian ini yaitu meliputi langkah-langkah dalam pembahasan metode pengukuran perangkat lunak *LOC*, *COCOMO*, *FPA* dengan tahapan seperti pada Gambar 1.



Gambar 1. *Flowchart* tahapan penelitian

Pada Gambar 1 disajikan 7 tahapan proses pembahasan metode pengukuran perangkat lunak, berikut penjelasan masing-masing proses tersebut.

1. Studi literatur  
Tahapan ini mengatur berbagai hal terkait pendalaman sumber referensi, seperti artikel, makalah, atau laporan yang bersangkutan terhadap permasalahan, mengingat sumber yang akan dimanfaatkan sebagai acuan terhadap kesimpulan.
2. Pengelompokkan  
Tahapan ini menjelaskan tentang proses pengelompokan terhadap berbagai sumber tertulis yang dikaji, sehingga mempermudah untuk mendapatkan data atau informasi terkait beberapa metode yang akan dijadikan perbandingan.
3. Metode LOC  
Tahapan ini menjelaskan tentang penjabaran secara terperinci dari metode pengukuran perangkat lunak LOC (*Line of Code*) berupa rumus, kelebihan, kekurangan, tingkat keakurasian, dan ketepatan pemakaian.
4. Metode COCOMO  
Tahapan ini menjelaskan tentang penjabaran secara terperinci dari metode pengukuran perangkat lunak COCOMO (*Constructive Cost Model*) berupa rumus, kelebihan, kekurangan, tingkat keakurasian, dan ketepatan pemakaian.
5. Metode FPA  
Tahapan ini menjelaskan tentang penjabaran secara terperinci dari metode pengukuran perangkat lunak FPA (*Function Point Analysis*) berupa rumus, kelebihan, kekurangan, tingkat keakurasian, dan ketepatan pemakaian.
6. Perbandingan

Tahapan ini menjelaskan tentang proses perbandingan berdasarkan proses pengelompokan metode pengukuran perangkat lunak dari berbagai sumber tertulis yang dikaji, sehingga mempermudah untuk mendapatkan data atau informasi mengenai beberapa metode yang akan dihasilkan. Pada tahap ini, dilakukan perbandingan dengan 5 aspek yaitu rumus, kelebihan, kekurangan, tingkat keakurasian, dan ketepatan pemakaian. Perbandingan ini berorientasi pada subjek, objek, serta fungsionalitas pada masing-masing metode yang dikaji. Kriteria yang diambil pada perbandingan ini yaitu dengan memprioritaskan keunggulan efektifitas atau efisiensi pada masing-masing metode dalam hal kenyamanan implementasi di suatu proyek.

7. Hasil

Tahapan ini menjelaskan hasil dari studi literatur perbandingan metode pengukuran perangkat lunak (LOC, COCOMO, FPA) yang telah dikelompokkan dan dikaji berdasarkan pengembangan *software metric*.

**HASIL DAN PEMBAHASAN**

Metode LOC, COCOMO, dan FPA pada *software metric* ini dapat dijabarkan berdasarkan studi literatur. Adapun rumus, kelebihan, dan kekurangan dari *software metric* khususnya metode LOC, COCOMO, dan FPA dalam pengukuran perangkat lunak yang secara detail dibahas pada Tabel 1.

**Tabel 1. Rumus, kelebihan, dan kekurangan *software metric***

<i>Metode</i>	<i>Rumus</i>	<i>Kelebihan</i>	<i>Kekurangan</i>
<i>LOC</i>	$C = a Lb$  C adalah biaya usaha dalam hitungan individu per bulan. L adalah baris kode sumber. a, b adalah konstanta yang berasal dari proyek - proyek sebelumnya.	Metode ini merupakan metrik yang paling umum dan paling sederhana dan juga memiliki kelebihan tertentu yaitu sebagai penghitung upaya dan durasi proyek perangkat lunak; lalu sebagai penghitung kendala lain yang berkaitan erat dengan pengembangan perangkat lunak; kemudian mudah untuk dihitung secara otomatis; serta dapat menghitung jumlah usaha yang akan diperlukan untuk membangun perangkat lunak. Sumber : (Eric Githua, 2015; Belachew, 2018; Ahmed, Sadath and Nagaria, 2019)	Metode ini memiliki kekurangan yaitu selalu bergantung pada bahasa yang digunakan dalam pengembangan perangkat lunak; lalu tidak memadai untuk menghitung tugas - tugas lain seperti fungsi, kompleksitas, dan efisiensi kode; kemudian tidak dapat dihitung secara akurat ketika perangkat lunak telah selesai; serta estimasi perangkat lunak harus dilakukan saat sebelum konstruksi. Sumber : (Eric Githua, 2015; Belachew, 2018; Ahmed, Sadath and Nagaria, 2019)
<i>COCOMO I</i>	$E = a (KLOC)^b$ E adalah upaya. a,b adalah konstanta yang berasal dari tabel parameter yang biasa disebut <i>COCOMO KLOC</i> .	Metode ini sangat populer karena adanya kemudahan aplikasi dan ketersediaannya, selain itu juga memiliki kelebihan yaitu sebagai alat penghitung dan perkiraan biaya proyek; serta hasilnya fakta dan mudah untuk diinterpretasikan. Sumber :	Metode ini memiliki kekurangan yaitu tidak berguna dalam pengukuran; lalu permintaan data historis yang tidak selalu tersedia, serta dalam proyek, detail data sebelumnya diperlukan. Sumber :

	<p>(Sachan <i>et al.</i>, 2016; Aljohani and Qureshi, 2017; Goyal and Parashar, 2018; Tianpei Xia, Rui Shu, Xipeng Shen, and Tim Menzies, 2020)</p>	<p>(Aljohani and Qureshi, 2017; Goyal and Parashar, 2018; K.Rajeswari and Dr.R.Beena, 2018; KULDEEP SINGH, 2018)</p>	<p>(Aljohani and Qureshi, 2017; K.Rajeswari and Dr. R.Beena, 2018)</p>
<p><i>COCOMO II</i></p>	$PM = 2.94 \prod_{i=1}^{17} EM_i S^{0.91+0.01 \sum_{i=1}^i W_i}$ $E = B + 0.01 \times \sum_{j=1}^5 SF_j$ <p><i>PM</i> adalah upaya per bulan. <i>S</i> adalah upaya dalam ukuran perorangan. <i>Emi</i> adalah pengganda usaha. <i>wi</i> adalah perluasan faktor skala. <i>E</i> adalah skala pada basis eksponen. <i>SF</i> adalah 5 faktor skala. <i>B</i> adalah skala basis eksponen.</p> <p>Sumber :                  (Khairani, 2015; Aljohani and Qureshi, 2017; Suhartoyo and Wijaya, 2017; Tianpei Xia, Rui Shu, Xipeng Shen, and Tim Menzies, 2020)</p>	<p>Metode ini memiliki kelebihan yaitu memberikan estimasi yang dapat ditiru dan objektif; lalu menghasilkan akurasi lebih; kemudian meningkatkan akurasi biaya keseluruhan; selain itu mudah dalam menyediakan informasi yang mendalam; lalu proses kalibrasi jelas dan efektif; serta dapat mengukur usaha, biaya, dan jadwal saat awal proyek.</p> <p>Sumber :                  (Bintiri <i>et al.</i>, 2012; Khairani, 2015; Aljohani and Qureshi, 2017; Thirumalai, Shridhar shan and Ranjith Reynold, 2017; K.Rajeswari and Dr. R.Beena, 2018)</p>	<p>Metode ini memiliki kekurangan yaitu dalam ekspansi perangkat lunak masih tetap menggunakan metode <i>SDLC waterfall</i>; lalu tidak cocok untuk usaha di semua fase <i>SDLC</i>; kemudian dalam proyek, detail data sebelumnya diperlukan; serta parameter pada model <i>COCOMO</i> menjadi kurang relevan dalam rangka pemeliharaan perangkat lunak.</p> <p>Sumber :                  (Bintiri <i>et al.</i>, 2012; Aljohani and Qureshi, 2017; Shad and Bahadar, 2017; K.Rajeswari and Dr. R.Beena, 2018; KULDEEP SINGH, 2018)</p>
<p><i>FPA</i></p>	$FP = Fu \times Wu + Fc \times Wc$ <p><i>Fu</i> adalah jumlah penggunaan. <i>Wu</i> adalah bobot penggunaan. <i>Fc</i> adalah jumlah kelas keseluruhan. <i>Wc</i> adalah bobot kelas. Disesuaikan :</p> $FP \times (0.65 + 0.01 \times \sum FJ)$ <p><i>FP</i> adalah titik fungsi. <math>\sum Fj</math> adalah jumlah 14 faktor yang memiliki rentang nilai (0-5).</p> <p>Sumber :                  (Kembang Hapsari, Jauhari and Wp, 2015; Khairani, 2015; Aljohani and Qureshi, 2017; Rachmat, 2017;</p>	<p>Metode ini memiliki kelebihan yaitu diyakini akurat; lalu dapat mengukur proyek dalam berbagai bahasa pemrograman; kemudian selain itu juga melampaui batasan dari LOC.</p> <p>Sumber :                  (Bintiri <i>et al.</i>, 2012; Aljohani and Qureshi, 2017; Govindharaj and B, 2017;</p>	<p>Metode ini memiliki kekurangan yaitu minimnya hasil <i>dataset</i> penelitian tentang <i>FPA</i>; lalu waktu, kualitas dan pekerjaan secara manual tidak terlalu dipertimbangkan; selain itu juga mekanismenya sulit dilakukan; pengukur harus memiliki pengalaman yang cukup; serta penghitungan dalam <i>FPA</i> melibatkan adanya tingkat subjektivitas.</p> <p>Sumber :                  (Bintiri <i>et al.</i>, 2012; Eric Githua, 2015; Aljohani and</p>

Rizka, Dewi and Wicaksono, K.Rajeswari and Dr.R.Beena, Qureshi, 2017; K.Rajeswari 2019; Rizky Wicaksono, 2018)(Govindharaj and B, and Dr. R.Beena, 2018) Kusumo Kresno Putro and 2017) Aprillia Immanuel, 2019)

Tabel 1 menjelaskan berbagai rumus, kelebihan, dan kekurangan yang ada pada *software metric* khususnya mengenai metode *LOC*, *COCOMO*, dan *FPA* dalam pengukuran perangkat lunak. Berdasarkan tabel tersebut dapat disimpulkan bahwa rumus, kelebihan, dan kekurangan dari masing-masing metode *software metric* dapat ditinjau dari kebutuhan proyek. Metode *LOC* merupakan metrik yang paling umum dan paling sederhana yang memiliki kelebihan yaitu sebagai penghitung upaya dan durasi proyek perangkat lunak serta memiliki kekurangan yaitu selalu bergantung pada bahasa yang digunakan dalam pengembangan perangkat lunak. Metode *COCOMO I* sangat populer karena adanya kemudahan aplikasi dan ketersediaannya, selain itu memiliki kelebihan yaitu sebagai alat penghitung dan perkiraan biaya proyek, serta memiliki kekurangan yaitu tidak berguna dalam pengukuran. Metode *COCOMO II* memiliki kelebihan yaitu memberikan estimasi yang dapat ditiru dan obyektif serta memiliki kekurangan yaitu dalam pengembangan perangkat lunak masih tetap menggunakan metode *SDLC waterfall*. Metode *FPA* memiliki kelebihan yaitu dapat mengukur proyek dalam berbagai bahasa pemrograman serta memiliki kekurangan yaitu mekanismenya sulit dilakukan.

Tingkat keakurasian dan ketepatan pemakaian dari *software metric* khususnya metode *LOC*, *COCOMO*, dan *FPA* dalam pengukuran perangkat lunak secara detail disajikan pada Tabel 2.

**Tabel 2. Akurasi dan pemakaian *software metric***

Metode	Akurasi	Pemakaian
<i>LOC</i>	<i>LOC</i> memiliki tingkat akurasi yang KURANG BAIK, karena ukuran pada metrik memiliki keterbatasan dalam penggunaannya. Sumber : (Khairani, 2015; Bela chew, 2018; Ahmed, Sadath and Nagaria, 2019)	Pemakaian <i>LOC</i> yang tepat yaitu digunakan sebagai penentuan implementasi, upaya, dan durasi proyek perangkat lunak.  Sumber : (Eric Githua, 2015; Belachew, 2018; Ahmed, Sadath and Nagaria, 2019)
<i>COCOMO I</i>	<i>COCOMO I</i> memiliki tingkat akurasi yang BAIK, karena dapat memudahkan dalam upaya menentukan biaya proyek. Sumber : (Aljohani and Qureshi, 2017; Goyal and Parashar, 2018; KULDEEP SINGH, 2018)	Pemakaian <i>COCOMO I</i> yang tepat yaitu digunakan sebagai penentuan biaya proyek, waktu pengembangan dan pemeliharaan usaha.  Sumber : (Sachan <i>et al.</i> , 2016; Aljohani and Qureshi, 2017; K.Rajeswari and Dr. R.Beena, 2018; KULDEEP SINGH, 2018)
<i>COCOMO II</i>	<i>COCOMO II</i> memiliki tingkat akurasi yang SANGAT BAIK, karena pada dasarnya memiliki kemampuan analisis yang lebih sempurna dibandingkan dengan metode yang lain.	Pemakaian <i>COCOMO II</i> yang tepat yaitu digunakan sebagai perkiraan usaha perangkat lunak, pengelolaan sebuah proyek atau penjualan perangkat lunak, serta untuk perkiraan biaya pada tingkat keberhasilan proyek.

	Sumber : (Aljohani and Qureshi, 2017; K.Rajeswari and Dr. R.Beena, 2018; Khoiro and Herlam bang, Admaja Dwi Saputra, 2018)	Sumber : (Fatonah and Afrizal, 2016; Aljohani and Qureshi, 2017; K.Rajeswari and Dr.R.Beena, 2018)
<i>FPA</i>	<i>FPA</i> memiliki tingkat akurasi yang KURANG BAIK, karena hanya terbatas pada fungsionalitas dan juga kompleksitas. Sumber : (Aljohani and Qureshi, 2017; K.Rajeswari and Dr. R.Beena, 2018; Rizky Wicaksono, Kusumo Kresno Putro and Aprillia Immanuel, 2019)	Pemakaian <i>FPA</i> yang tepat yaitu digunakan sebagai penentuan kompleksitas proyek, serta perbandingan hasil dari beberapa lingkungan perangkat lunak yang berbeda. Sumber : (Khairani, 2015; Aljohani and Qureshi, 2017; K.Rajeswari and Dr.R.Beena, 2018)

Tabel 2 menjelaskan tentang berbagai tingkat keakurasian dan ketepatan pemakaian pada *software metric* khususnya mengenai metode *LOC*, *COCOMO*, dan *FPA* dalam pengukuran perangkat lunak. Berdasarkan tabel tersebut dapat disimpulkan bahwa tingkat keakurasian dan ketepatan pemakaian dari *software metric* dapat ditinjau dari adanya perbedaan fungsionalitas. *LOC* memiliki tingkat akurasi yang KURANG BAIK, karena ukuran pada metrik memiliki keterbatasan dalam penggunaannya, selain itu ketepatan pemakaian lebih cocok digunakan untuk penentuan implementasi, upaya, dan durasi proyek. *COCOMO I* memiliki tingkat akurasi yang BAIK, karena dapat memudahkan dalam upaya menentukan biaya proyek, selain itu ketepatan pemakaian lebih cocok digunakan untuk penentuan biaya proyek, waktu pengembangan dan pemeliharaan usaha. *COCOMO II* memiliki tingkat akurasi yang SANGAT BAIK, karena pada dasarnya memiliki kemampuan analisis yang lebih sempurna dibandingkan dengan metode yang lain, selain itu ketepatan pemakaian lebih cocok digunakan untuk perkiraan usaha perangkat lunak, pengelolaan sebuah proyek atau penjualan perangkat lunak, serta untuk perkiraan biaya pada tingkat keberhasilan proyek. *FPA* memiliki tingkat akurasi yang KURANG BAIK, karena hanya terbatas pada fungsionalitas dan juga kompleksitas, selain itu ketepatan pemakaian lebih cocok digunakan untuk penentuan kompleksitas proyek, serta perbandingan hasil dari beberapa lingkungan perangkat lunak yang berbeda. Berdasarkan data yang telah diperoleh, target yang tepat guna dari suatu metode tergantung pada implementasinya, jadi perlu diketahui bahwa perencanaan dalam penggunaan metode estimasi itu sendiri harus berdasarkan lingkungan proyek.

## SIMPULAN

Pengukuran perangkat lunak (*software metric*) adalah tindakan mengevaluasi dan menilai pelaksanaan kerangka kerja atau seperangkat instrumen untuk memantau dan menentukan tingkat produktivitas perangkat lunak berbasis pada pendekatan kuantitatif. Prediksi akurasi pada pengembangan perangkat lunak adalah masalah penting untuk membuat keputusan manajemen yang baik dan secara akurat menentukan berapa banyak usaha dan waktu proyek yang dibutuhkan untuk manajer proyek serta analisis sistem dan pengembang. Pemilihan metode estimasi mana yang harus digunakan itu harus berdasarkan lingkungan proyek. Metode *LOC* merupakan metrik yang paling umum dan paling sederhana sebagai penghitung upaya dan durasi proyek perangkat lunak yang memiliki tingkat akurasi yang KURANG BAIK, karena ukuran pada metrik memiliki keterbatasan dalam penggunaannya. Metode *COCOMO I* sangat populer karena adanya kemudahan aplikasi dan

ketersediaannya sebagai alat penghitung dan perkiraan biaya proyek yang memiliki tingkat akurasi yang BAIK, karena dapat memudahkan dalam upaya menentukan biaya proyek. Metode COCOMO II dapat memberikan estimasi yang dapat ditiru dan obyektif serta memiliki tingkat akurasi yang SANGAT BAIK, karena pada dasarnya mempunyai kemampuan analisis yang lebih sempurna dibandingkan dengan metode yang lain. Metode FPA dapat mengukur proyek dalam berbagai bahasa pemrograman serta memiliki tingkat akurasi yang KURANG BAIK, karena hanya terbatas pada fungsionalitas dan juga kompleksitas.

## REFERENSI

- Ahmed, S., Sadath, L. & Nagaria, J. (2019). Software testing and lines of codes-a study on software engineering design patterns. In *2019 International Conference on Automation, Computational and Technology Management (ICACTM)*, 389–394. doi: 10.1109/ICACTM.2019.8776688.
- Aljohani, M. & Qureshi, M. R. (2017). Comparative study of software estimation techniques. *IJSEA*, 8, 39-53.
- Bukhari, Z., Yahaya, J. & Deraman, A. (2015). Software metric selection methods: A review. In *2015 International Conference on Electrical Engineering and Informatics (ICEEI)*, 433–438. doi: 10.1109/ICEEI.2015.7352540.
- Chahal, K. & Singh, H. (2010). Exploring design level class cohesion metrics. *JSEA*, 3, 384–390. doi: 10.4236/jsea.2010.34043.
- Dhanuja, B., & Sathiyaa, G. (2017, April). Software metric LOC data analysis for symmetric key cryptographic technique. In *2017 Innovations in Power and Advanced Computing Technologies (i-PACT)* (pp. 1-5). IEEE.
- Goyal, S. & Parashar, A. (2018). Machine learning application to improve COCOMO model using neural networks. *International Journal of Information Technology and Computer Science*. doi: 10.5815/ijites.2018.03.05.
- Hapsari, R. K., & WP, M. J. H. (2015). Estimasi Kualitas Perangkat Lunak Berdasarkan Pengukuran Kompleksitas Menggunakan Metrik Function Oriented. *Semin. Nas. Sains dan Teknol. Terap*, 3, 425-434.
- Khairani, D. (2015). Studi kasus pengukuran sistem informasi menggunakan function point (FP). *JURNAL TEKNIK INFORMATIKA*, 8. doi: 10.15408/jti.v8i2.2442.
- Khoiro, M. N., Herlambang, A. D., & Saputra, M. C. (2018). Evaluasi biaya pengembangan perangkat lunak dengan menggunakan metode cocomo II (Studi Kasus: PT DOT Indonesia).” *JPTIHK*, 2(10), 1–10.
- Kuldeep Singh, U. D. (2018). Analysis of cost size and effort estimation models for software system. *International Journal of Advances in Electronics and Computer Science*, 5(7), 1-5.
- Langsari, K., & Sarno, R. (2017, September). Optimizing effort and time parameters of COCOMO II estimation using fuzzy multi-objective PSO. In *2017 4th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, 1-6. IEEE.
- Rajeswari, K., & Beena, D. R. (2018). A critique on software cost estimation. *International Journal of Pure and Applied Mathematics*, 118(20), 3851-3862.
- Rachmat, N., & Saparudinn, S. (2017, November). Estimasi ukuran perangkat lunak menggunakan function point analysis-studi kasus aplikasi pengujian dan pembelajaran berbasis web. In *Annual Research Seminar (ARS)*, 3(1), 57-60.
- Rashid, J., Mahmood, T., & Nisar, M. W. (2019). A study on software metrics and its impact on software quality. arxiv preprint arXiv:1905.12922.
- Rizka, C. L. D., Dewi, F. S. & Wicaksono, S. R. (2019). Pengukuran dan kualitas perangkat lunak website ‘linkedin’ menggunakan metode function point analysis. *JISKA (Jurnal Informatika Sunan Kalijaga)*. doi: 10.14421/JISKA.2018.32-02.
- Rizky Wicaksono, S., Kusumo Kresno Putro, P. & Aprillia Immanuel, G. (2019). Implementasi function point analysis untuk pengukuran kualitas situs (Studi Kasus: Alibaba.com). *Jurnal Informatika: Jurnal Pengembangan IT*.
- Shad, M. J. & Bahadar, F. (2017). Analyzing cost estimation model to optimize COCOMO II for enterprise level software. in Shad, M. J. and Bahadar, F. (eds.). *Pakistan: Global Society of Scientific Research and Researchers*, pp. 168–179. Available at: <http://ijcjournal.org/>.
- Sharma, M., Sharma, T. & Sakpal, M. (2016). An analytic study and implementation of constructive system engineering cost estimation models. *International Journal of Computer Applications*, 135, 4–7. doi:

10.5120/ijca2016908260.

Sophatsathit, N. (2017). Complexity measure of software composition framework. *JSEA*, 10(4), 1–14. doi: 10.4236/jsea.2017.104019.

Suhartoyo, H. & Wijaya, T. A. (2017). Rancangan estimasi biaya dengan teknik COCOMO II dan neuro fuzzy studi kasus: sistem informasi rumah sakit. 1–11.

Tianpei Xia, Rui Shu, Xipeng Shen, & Tim Menzies, F. (2020). Sequential model optimization for software process control.”*IEEE*, 1–14.