

ESTIMASI PENGUKURAN PERANGKAT LUNAK MENGUNAKAN METODE MATRIK FUNCTION POINT ANALYSIS – STUDI KASUS APLIKASI AKADEMIK SISWA BERBASIS WEBSITE

Rizky Parluka¹, Rahmadany Fahreza Taufiqurrahman², Hafi Ihza Farhana³, Rahmat Dimas Syahputra⁴ dan Fairuz Aldifa⁵

^{1,2,3,4,5} Informatika, Universitas Pembangunan Nasional Veteran Jawa Timur
Jl. Rungkut Madya No.1 Surabaya, Indonesia

Email: rizkyparlika.if@upnjatim.ac.id¹, 20081010089@student.upnjatim.ac.id², 20081010094@student.upnjatim.ac.id³, 20081010113@student.upnjatim.ac.id⁴, 20081010118@student.upnjatim.ac.id⁵

ABSTRAK

Function Point (FP) merupakan metode pengukuran sekaligus satuan ukuran dari sebuah perangkat lunak. FP dapat mengukur perangkat lunak dengan cara mengkuantifikasi (jumlah satuan dalam angka) fungsionalitas pada perangkat lunak yang sudah disediakan untuk pengguna berdasarkan pada logika. Matrik FP dihitung dari 5 komponen yang antara lain jumlah input pemakai (external input-EI), jumlah output kepada pemakai (external inquiry – EQ), jumlah inquiry pemakai (external enquiry – EQ), jumlah file internal yang digunakan (internal logical file – ILF), dan jumlah antarmuka eksternal (external file – EIF). Studi kasus ini bertujuan untuk mempelajari secara seksama cara untuk menghitung FP yang menghasilkan hasil akhir dalam bentuk estimasi biaya pada perangkat lunak. Perangkat lunak yang telah ada yakni Aplikasi Akademik Siswa Berbasis Website (AASBW). Berkas teknis yang tersedia dari perangkat lunak ini adalah source code dan running application. Pengukuran FP ini akan dilakukan dari sudut pandang perangkat lunak, yakni mengukur aplikasi yang sudah jadi. Dari hasil observasi didapatkan bahwa FP untuk AASBW = 152,90 dan estimasi biaya untuk memasarkan aplikasi ini mencapai Rp. 7.645.000, estimasi waktu produksi selama 29 hari, dan estimasi biaya manajemen aplikasi ini sebesar Rp.3.850.000, sehingga hasil dari ketiga faktor estimasi tersebut dijadikan penulis sebagai bahan analisis pada Aplikasi Akademik Siswa Berbasis Website yang menggunakan metode pengukuran Function Point Analysis.

Kata Kunci: Function Point (FP), Software Metrics, Software Engineering (RPL), Estimasi Proyek, pengukuran.

ABSTRACT

Function Point (FP) is a measurement method as well as a unit of measure from a software. FP can measure software by quantifying (number of units in numbers) functionality in software that is already provided to users based on logic. The FP matrix is calculated from 5 components which include the number of user inputs (external input-EI), the number of outputs to the user (external inquiry - EQ), the number of user inquiries (external inquiry - EQ), the number of internal files used (internal logical file - EQ). ILF), and the number of external interfaces (external files – EIF). This case study aims to study carefully how to calculate FP which produces the final result in the form of cost estimation in software. The existing software is the Website-Based Student Academic Application (AASBW). The technical files available from this software are source code and running application. This FP measurement will be carried out from a software point of view, namely measuring the finished application. From the observation, it is found that the FP for AASBW = 152.90 and the estimated cost to market this application is Rp. 7,645.000, the estimated production time is 29 days, and the estimated cost of managing this application is Rp. 3,850,000, so the results of the three estimation factors are used by the authors as analysis material for Website-Based Student Academic Applications using the Function Point Analysis method.

Keywords: Function Point (FP), Software Metrics, Software Engineering (RPL), Project Estimation, measurement

I. PENDAHULUAN

Perangkat lunak yang dikembangkan pada saat ini sudah jauh lebih kompleks dan berkembang dibanding dengan tahun awal mulanya perangkat lunak dibentuk di rentang tahun 1970 yang mana perangkat lunak beroperasi masih menggunakan *processor* tunggal. Perangkat lunak yang berkembang pada saat ini memiliki *Graphical User Interface* (GUI) dan beberapa arsitektur pada *client-server*. Upaya pengembangan perangkat lunak merupakan hal yang penting dalam pengelolaan proyek bagi organisasi tertentu. Akurasi serta upaya yang dapat diandalkan dan biaya yang ditentukan pada proyek perangkat lunak sangat utama dalam perencanaan, pengembangan, serta penjadwalan, terutama pada tahap awal *software development life cycle* [1]. Jadi masalah utama pada pengembangan perangkat lunak ini terletak pada kehandalan serta keakuratan dalam pengelolaan proyek perangkat lunak [2].

Teknik estimasi ialah salah satu dari sekian banyaknya cara untuk menangani secara efektif dalam penentuan jadwal dan biaya dalam sebuah proyek perangkat lunak [3]. Sebuah prediksi ini jika akurat nantinya dapat bermanfaat pada perencanaan suatu proyek perangkat lunak sehingga pengelolannya menjadi lebih baik, hal ini

menjamin kualitas pengembangan pada perangkat lunak. Jika tim pengembang tidak memiliki data yang cukup jelas, pengembang akan menghabiskan banyaknya tenaga dalam mengumpulkan data pengembangan pada perangkat lunak yang mereka kembangkan[4].

Metode analisa FP ini dapat digunakan untuk mengukur estimasi upaya penentuan biaya pemasaran pada perangkat lunak yang dipilih. Hal ini dilakukan dengan cara membagi sistem ke dalam beberapa komponen sehingga dapat dianalisis dan dimengerti. Masalah yang kompleks mungkin akan tampak lebih sederhana setelah dibagi dan dipecah menjadi bagian yang terpisah-pisah kemudian dikelompokkan.

Dengan melakukan perhitungan FP pada perangkat lunak yang pernah dikerjakan maupun dikembangkan sebelumnya, akan memudahkan tim pengembang dalam memprediksi biaya pada proyek perangkat lunak sehingga meminimalisir upaya dalam pengumpulan data dengan skala yang besar. Meskipun penilaian ahli tetap ada yang digunakan dan ada pula juga yang berminat dalam penerapan teknik *machine learnik* dan stastitika untuk memprediksi usaha pada perangkat lunak [5]. Tiga komponen utama dari biaya usaha dominan, biaya perangkat, biaya perjalanan, serta biaya pelatihan pada komponen perangkat lunak [6]. Meskipun kompleksitas estimasi ini terkadang hanya dilakukan oleh seorang ahli berdasarkan estimasinya sendiri, dalam beberapa dekade terakhir ini, FP telah dikembangkan untuk menangani estimasi pada proyek perangkat lunak [7].

II. METODE PENELITIAN

Metode pengukuran yang penulis gunakan sebagai metode dalam penelitian ini. Pengukuran yang d\dimaksud mengacu pada beberapa estimasi pada biaya pemasaran dan waktu produksi yang akan diuji dengan *Function Point Analysis* dengan mengujinya pada Aplikasi Akademik Siswa Berbasis Website (AASBW).

Jenis pengukuran yang penulis gunakan merupakan prinsip dasar pengukuran menurut Roche [15], hal ini menunjukkan bahwa kegiatan pengukuran ini dapat dikategorikan berdasarkan lima faktor yang diantaranya:



Gambar 1. Metode Penelitian

1.1 Function Point (FP)

Analisa Function Point yang telah dikembangkan oleh Allan J. Albrecht di tahun 1970-an. FP ini dapat digunakan untuk mengukur estimasi kompleksitas dari perangkat lunak [8], estimasi ini meliputi pengukuran waktu, dan biaya selama pengerjaan perangkat lunak. Jumlah dari total FP ini tergantung pada perhitungan yang berbeda (dalam hal format maupun pengolahan logika) jenis. Berikut merupakan 5 jenis komponen FP yang terdiri dari 2 data fungsi dan 3 tipe transaksi [4]:

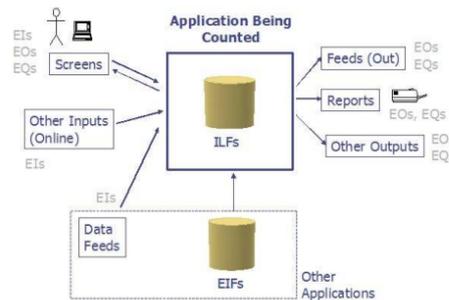
1. *Internal Logical File (ILF)*, merupakan logika data yang disimpan pada saat maintained dalam aplikasi boundary.

2. *External Interface File (EIF)*, merupakan pengelompokan logika pada data yang berada di luar aplikasi namun menyediakan data yang mungkin dapat digunakan pada aplikasi tersebut. EIF *equal* ILF hanya saja ILF yang berada pada aplikasi lain disebut dengan EIF.

3. *External Inputs (EI)*, merupakan proses data EI yang mana kontrol informasi ini datang dari luar aplikasi boundary. EI juga dapat berupa aktifitas dari unit terkecil sebuah aplikasi. Pada implementasinya EI ini berupa field-field yang ada pada aplikasi.

4. *External Outputs (EO)*, merupakan proses dasar yang mana data melintasi boundary dari dalam aplikasi ke luar aplikasi. EO dapat memperbarui ILF. Pada implementasi EO ini berupa laporan yang bisa dicetak.

5. *External Inquiry (EI)*, merupakan sebuah proses dasar yang terdiri dari gabungan dari input-output yang menghasilkan data kembalian (*retrieve*) atau data yang dapat dilihat.



Gambar 2. Lima Komponen pada Function Point

Dengan memahami kelima komponen ini, tim pengelola proyek dapat menentukan apakah tingkat kesulitan pada proyek tersebut dapat dikatakan masuk ke dalam kategori mudah, rata-rata ataupun kompleks. Sehingga tim pengelola proyek dapat lebih mudah untuk mengembangkan proyek yang serupa. Berikut ini merupakan cara menentukan kategori apakah mudah, rerata ataupun kompleks. Jika RET yang diketahui, maka digunakan Tabel 1, sedangkan jika FTR yang diketahui maka Tabel 2 yang akan digunakan [9].

Tabel 1. ILF/EIF Matriks Kompleksitas

RET	Data element Type (DET)		
	1-19	20-50	51+
1	Sederhana	Sederhana	Menengah
2 to 5	Sederhana	Sederhana	Kompleks
6 or more	Sederhana	Kompleks	Kompleks

Tabel 2. ILF/EIF Matriks Kompleksitas

FTR	Data element Type (DET)		
	1-4	5-15	16+
0-1	Sederhana	Sederhana	Menengah
2	Sederhana	Menengah	Kompleks
3 (if EL) or 4 (if EQ, EQ) or more	Menengah	Kompleks	Kompleks

Total dari perhitungan function point ini disebut dengan *unadjusted function point (UFP)* ISO Standard sebuah standar internasional yang digunakan untuk mengukur FP. UFP merupakan hasil perhitungan dari penjumlahan lima komponen yang meliputi ILF + EIF + EI + EO + EQ. Setelah menemukan ke lima komponen tersebut, berikut adalah istilah penting dan definisi yang digunakan dalam menggambarkan kelima komponen tersebut[10]:

1) *User Identifiable*: Istilah ini mengacu pada persyaratan yang ditetapkan pada sebuah proses maupun kelompok data yang disepakati, dan dipahami oleh pengguna dan pengembang pada perangkat lunak.

2) *Control Information*: Istilah ini mengacu pada proses dasar dari aplikasi yang sudah dihitung dan juga secara khusus menyebutkan beberapa pertanyaan yang meliputi data apa yang akan diproses, kapan data akan diproses, atau bagaimana data akan diproses.

3) *Elementary Process*: Unit terkecil dari sebuah aktivitas yang berarti penting bagi pengguna. Elementary Process ini harus serba lengkap dan meninggalkan proses bisnis aplikasi yang diukur sedang dalam keadaan konsisten.

4) *Data Element Type (DET)*: Sebuah elemen data yang dikenal oleh pengguna seperti: Nama Siswa, No absen Siswa, dan Tanggal lahir.

5) *Record Element Type (RET)*: Sebuah subgrup data yang dikenali oleh user seperti: data siswa yang terdiri dari biodata siswa, kehadiran siswa, dan nilai siswa.

6) *File Type Reference (FTR)*: Sebuah ILF yang diakses ataupun dibaca oleh proses elementer yang meliputi EI/EO/EQ ataupun EIF.

Ada beberapa hal yang harus diperhatikan dan mungkin dilakukan dalam proses analisis function point yaitu pengukuran ini dilihat dari sudut pandang subjektifitas pengguna, biaya, *multi-platform*, menganalisa ulang proyek, dan function point yang bekerja dengan baik pada use case [11]. Dengan hal-hal tersebut tim pengelola proyek perangkat lunak dapat lebih mudah untuk mengidentifikasi perangkat lunaknya dengan baik dan lebih akurat. Dalam kehidupan nyata function point ini juga dapat digunakan sebagai dasar untuk mengukur estimasi biaya pemasaran aplikasi, estimasi waktu produksi, dan estimasi biaya manajemen pada aplikasi.

1.2 Menghitung Crude Function Point (CFP)

Dalam perhitungan komponen pada FP, tiap-tiap tipe komponen tersebut diberikan bobot berdasarkan dari kompleksitasnya. Contoh pembobotan ini telah disesuaikan dengan *Function Point International User Group (FPIUG)*. CFP atau yang biasa disebut dengan *Crude Function Point* merupakan bobot nilai yang digunakan untuk menghitung dari komponen-komponen pada FP yang telah dikaitkan dengan perangkat lunak yang telah dipilih[12].

Tabel 3. Draft Perhitungan CFP

Komponen Sistem Perangkat Lunak	Tingkat Kompleksitas									Total CFP
	Low			Medium			High			
	Jumlah	Bobot	Point	Jumlah	Bobot	Point	Jumlah	Bobot	Point	
	A	B	$C=A * B$	A	B	$C=A * B$	A	B	$C=A * B$	
Input										
Output										
Online Query										
Logic File										
External Interface										
Total CFP										$J=C+F+I$

1.3 Menghitung Relative Complexity Adjustment Factor (RCAF)

RCAF digunakan untuk menghitung bobot kompleksitas dari perangkat lunak berdasarkan 14 karakteristik. Berikut merupakan ke-14 kriteria atau karakteristik yang ditunjukkan pada tabel dibawah ini [11]:

Tabel 4. 14 kriteria pada perangkat lunak

No.	Kriteria	Definisi
1.	<i>Data Communication</i>	Tingkat kebutuhan fungsional pada aspek komunikasi secara langsung antara aplikasi dan processor.
2.	<i>Distributed Data Processing</i>	Tingkat kebutuhan fungsional pada aspek transfer data antara komponen-komponen aplikasi.
3.	<i>Performance</i>	Tingkat response time(waktu tanggap) dan throughput(transfer data efektif, diukur dengan satuan bit/sekon) yg

		perlu dipertimbangkan dalam pengembangan aplikasi
4.	<i>Heavily used configuration</i>	Tingkat kebutuhan fungsional pada aspek setting konfigurasi komputer yang berpengaruh terhadap pengembangan aplikasi.
5.	<i>Transaction rate</i>	Tingkat kecepatan transaksi bisnis yang dapat mempengaruhi pengembangan pada aplikasi.
6.	<i>Online Data Entry</i>	Tingkat kebutuhan fungsional pada aspek penginput-an data secara interaktif
7.	<i>End-user efficienc</i>	Tingkat kemudahan penggunaan (<i>user experience</i>) aplikasi
8.	<i>Online Update</i>	Tingkat kebutuhan ILF(logic pada data) diupdate secara online
9.	<i>Complex Process</i>	Tingkat kesulitan logika proses yang dapat mempengaruhi proses pengembangan aplikasi.
10.	<i>Reusability</i>	Tingkat kebutuhan fungsional pada aspek kode program pada aplikasi yang dirancang dan dikembangkan untuk bisa digunakan pada aplikasi lain.
11.	<i>Installation Ease</i>	Tingkat kemudahan konversi ke sistem baru yang dapat memengarui proses pengembangan aplikasi.
12.	<i>Operational Ease</i>	Tingkat kemudahan aplikasi dalam kebutuhan fungsional pada aspek operasional, seperti startup, backup data, dan proses recovery data.
13.	<i>Multiple Sites</i>	Tingkat kebutuhan fungsional pada aplikasi yang dapat dioperasikan pada lingkungan <i>hardware</i> dan <i>software</i> yang berbeda jenis.
14.	<i>Facilitate Change</i>	Tingkat kemudahan aplikasi untuk memodifikasi logika proses maupun struktur data pada aplikasi.

Penilaian kompleksitas ini memiliki skala antara 0 – 5, hal tersebut ditunjukkan pada tabel dibawah ini [13]:

Tabel 5. Penilaian Kompleksitas

Value	Tingkat Kompleksitas
0.	Tidak Pengaruh
1.	Insidental
2.	Moderat
3.	Rata-rata
4.	Signifikan
5.	Essential

1.3 Menghitung Function Point (FP)

Sedangkan untuk menghitung FP, maka digunakan rumus, sebagai berikut [14]:

$$FP = CFP \times (0,65 + 0,01 \times RCAF)$$

Ket:
FP = *Functional Point*
CFP = *Crude Function Point*
0, 65 dan 0, 01 = **Konstanta yang telah dibuat oleh *Function Point International User Group (IEPUG)***

III. HASIL PEMBAHASAN

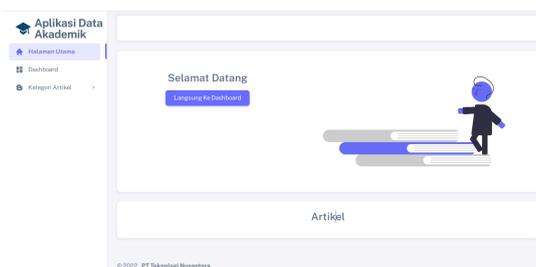
3.1 Prosedur serta Implementasi

3.1.1 Hasil dan Analisa dari Faktor *Formulation*

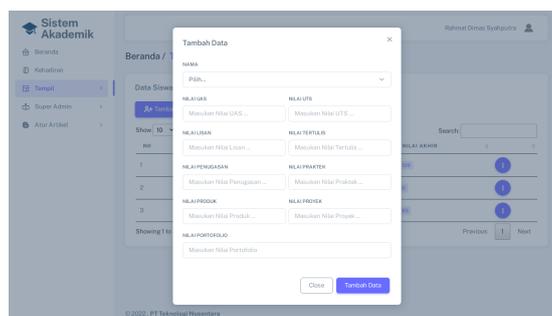
Dalam penelitian estimasi pada kali ini, penulis menggunakan analisis FP dalam pengukuran pada Aplikasi Akademik Siswa Berbasis Website (AASBW). Semisal dalam kasus yang berbeda, penulis akan melakukan penelitian dengan menggunakan metode GQM [16]. *Goal Question Metrics Method (GQM)* merupakan metode yang dimulai dari merancang top-levels goals dari proyek, kemudian dibuatkan beberapa serangkaian pertanyaan untuk tiap goal dimana tiap pertanyaan ini dijawab jika goal telah tercapai, dan kemudian metrik digunakan untuk mengukur hasil dari pertanyaan tersebut [17].

3.1.2 Hasil dan Analisa dari Faktor *Collection*

Dalam penelitian yang dilakukan pada kali ini, penulis menilai sebuah aplikasi pendataan nilai siswa di ruang lingkup sekolah, <https://pemweb-90f9f.web.app/>.



Gambar 3. Halaman Dashboard AASBW



Gambar 4. Halaman Tambah Data AASBW

AASBW adalah aplikasi berbasis website yang dapat membantu pengajar melakukan pendataan nilai, absensi, ekspor nilai dalam bentuk pdf ataupun excel. Aplikasi ini juga dapat membantu siswa untuk melihat ranking dengan standar nilai tertinggi dan dapat membantu juga melihat hasil nilai dari siswa dalam bentuk website pada materi dan jumlah nilai di semester tertentu.

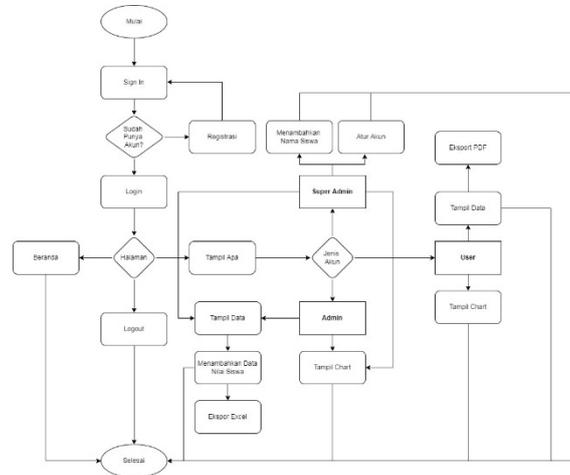
1. Input: Daftar nilai siswa yang sudah diambil dari semester tertentu
2. Output: Data Nilai, Tampilan nilai dalam bentuk diagram, dan ekspor nilai siswa dalam format pdf ataupun excel

Aplikasi ini juga memiliki tiga jenis akun yang memiliki fungsionalitas masing-masing dalam penggunaannya, yang meliputi:

1. *Super Admin* adalah struktur tertinggi atau pengendali penuh berjalanya sistem.
2. *Admin* adalah struktur ke dua setelah *Super Admin*. *Admin* hanya bertugas untuk melakukan CRUD (*Create Read Update Delete*) pada data siswa.

3. *User* adalah struktur ketiga yang memiliki akses untuk melihat detail data siswa dan diagram.

Aplikasi ini digunakan sebagai objek penelitian mengingat fungsionalitasnya yang terbatas dan memiliki ruang lingkup domain yang kecil, sehingga memudahkan penggunaannya sebagai objek studi kasus [18]. Dari hasil analisis terhadap aplikasi yang dapat penulis dapatkan, penulis memetakan ruang lingkup fungsionalitas ke dalam alur proses yang dijabarkan dengan diagram alur (*flowchart*) yang ditunjukkan dalam gambar berikut:



Gambar 4 Diagram Alir Fungsionalitas pada AASBW pada studi kasus

Berdasarkan analisis fungsionalitas yang telah penulis lakukan pada Aplikasi Akademik Siswa berbasis Website diperoleh data berikut dan dapat diturunkan dalam bentuk formulir pembobotan sebagai berikut:

Tabel 6. Pengklasifikasian Fungsionalitas untuk AASBW

No	Fungsionalitas	Type	Level Kompleksitas
1.	Melihat data nilai siswa	View	Sederhana
2.	Memasukkan data nilai siswa	Input	Sederhana
3.	Mengatur data siswa	Input	Menengah
4.	Mengatur akun siswa	Input	Menengah
5.	Mengatur akun admin	Input	Menengah
6.	Mengatur akun super admin	Input	Menengah
7.	Mengatur daftar kehadiran siswa	Input	Menengah
8.	Mengatur tambah artikel	Input	Menengah
9.	Mengatur data artikel	Input	Menengah
10.	Mengatur kategori artikel	Input	Menengah
11.	Mengatur tampil data diagram	Input	Menengah
12.	Mengatur bentuk diagram (batang dan lingkaran)	Input	Menengah
13.	Mengekspor excel	Output	Sederhana
14.	Mengekspor pdf	Output	Sederhana
15.	Melihat tampilan tabel informasi data siswa	View	Sederhana
16.	Melihat tampilan diagram informasi data siswa	View	Sederhana
17.	Melihat Data Informasi kehadiran/absensi siswa	View	Sederhana
18.	Melihat informasi artikel yang tersedia didalam aplikasi	View	Sederhana
19.	Login akun super admin	Query	Kompleksitas
20.	Login akun admin	Query	Kompleksitas
21.	Login akun siswa	Query	Kompleksitas

22.	Pendaftaran siswa online	Interface External	Menengah
23.	Mencari data siswa pada tabel	Search	Kompleksitas
24.	Memunculkan Pagination (Fitur untuk membatasi jumlah maksimum pada tabel)	Search	Kompleksitas
25.	File .pdf untuk data siswa maupun diagram	File	Menengah
26.	File .excel untuk data siswa	File	Menengah
27.	Tabel data nilai siswa	Table	Menengah
28.	Tabel kehadiran siswa	Table	Menengah
29.	Diagram lingkaran untuk nilai siswa	Graphic	Menengah
30.	Diagram batang untuk nilai siswa	Graphic	Menengah

Tabel diatas menunjukkan hasil dari beberapa fungsionalitas pada AASBW untuk menilai tiap-tiap bobot tersebut pada level kompleksitasnya masing-masing, berikut telah penulis rincikan tiap-tiap definisinya dari sudut pandang pengguna sebagai berikut:

1. Hasil menunjukkan pada fungsionalitas dari No.1 dengan tipe *view* pada fitur melihat data nilai siswa (fitur yang dilakukan oleh siswa) menjelaskan bahwa bobot bernilai sederhana dikarenakan pada tampilan tersebut digunakanlah tabel untuk menampung isi dari nilai yang ada dengan menggunakan framework dari css yaitu bootstrap.
2. Hasil menunjukkan pada fungsionalitas dari No. 2 dengan tipe *input* pada fitur memasukkan data nilai siswa (fitur yang dilakukan oleh admin) bahwa bobot bernilai sederhana dikarenakan untuk mengoperasikan fitur ini pengguna hanya perlu menginputkan nilai siswa pada tiap-tiap kolom form yang sudah ada, selanjutnya nilai ini akan ditampilkan outputnya dengan tabel seperti fitur dari No. 1.
3. Hasil menunjukkan pada fungsionalitas dari No. 3 dengan tipe *input* pada fitur mengatur data siswa (fitur yang dilakukan oleh admin) bahwa bobot bernilai menengah dikarenakan fungsionalitas pada fitur ini memiliki empat sub fitur yang meliputi *Create* (Membuat/Memasukkan Nilai Siswa), *Read* (Melihat output nilai siswa yang dimasukkan), *Update* (Merubah nilai siswa yang telah dimasukkan), dan *Delete* (Menghapus data nilai siswa).
4. Hasil menunjukkan pada fungsionalitas dari No. 4 dengan tipe *input* pada fitur mengatur akun siswa (fitur yang dilakukan oleh admin) bernilai menengah dikarenakan untuk login ke dalam AASBW setiap akun yang berhasil registrasi maka admin ataupun superadmin akan memvalidasi akun tersebut dengan tipe akun siswa.
5. Hasil menunjukkan pada fungsionalitas dari No. 5 dengan tipe *input* pada fitur mengatur akun admin (fitur yang dilakukan oleh super admin) bernilai menengah dikarenakan untuk akun admin dibuatkan secara khusus oleh superadmin sebagai pembuat dan validator dari setiap akun yang berhasil registrasi.
6. Hasil menunjukkan pada fungsionalitas dari No. 6 dengan tipe *input* pada fitur mengatur akun superadmin (fitur yang dilakukan oleh pengembang) bernilai menengah dikarenakan untuk mempunyai akun superadmin ini tim pengembang sendiri yang membuat satu akun khusus bertipe superadmin sebagai level paling tinggi untuk mengoperasikan AASBW.
7. Hasil menunjukkan pada fungsionalitas dari No. 7 dengan tipe *input* pada fitur mengatur daftar kehadiran siswa (fitur yang dilakukan oleh admin) bernilai menengah dikarenakan pada fitur ini memiliki tiga jenis kolom pengaturan yang meliputi izin, absen, dan sakit. Ketiga jenis ini dapat dioperasikan dengan konsep CRUD.
8. Hasil menunjukkan pada fungsionalitas dari No. 8 dengan tipe *input* pada fitur menambah artikel (fitur yang dilakukan oleh admin) bernilai menengah dikarenakan untuk menambah artikel diberikan sebuah form penulisan artikel pada AASBW, penambahan artikel ini memiliki banyak fitur yang ada didalamnya sehingga perlu penguasaan untuk menambahkan artikel dalam aplikasi ini,
9. Hasil menunjukkan pada fungsionalitas dari No. 9 dengan tipe *input* pada fitur mengatur data artikel (fitur yang dilakukan oleh admin) bernilai menengah dikarenakan pada fitur ini dapat dioperasikan dengan konsep CRUD pada artikel.
10. Hasil menunjukkan pada fungsionalitas dari No. 10 dengan tipe *input* pada fitur mengatur kategori artikel (fitur yang dilakukan oleh admin) bernilai menengah dikarenakan pada fitur ini akan mengatur tiap-tiap jenis kategori artikel yang telah diinputkan.
11. Hasil menunjukkan pada fungsionalitas dari No. 11 dengan tipe *input* pada fitur mengatur tampil data diagram (fitur yang dilakukan oleh admin) bernilai menengah dikarenakan pada fitur ini admin bisa mengatur data pada diagram sesuai dengan nilai siswa yang telah admin inputkan pada fitur No. 2.

12. Hasil menunjukkan pada fungsionalitas dari No. 12 dengan tipe *input* pada fitur mengatur bentuk diagram (fitur yang dilakukan oleh admin) bernilai menengah dikarenakan pada fitur ini admin bisa mengubah bentuk diagram dari inputan data nilai siswa dari diagram batang maupun diagram lingkaran.
13. Hasil menunjukkan pada fungsionalitas dari No. 13 dengan tipe *output* pada fitur mengeksport excel (fitur yang dilakukan oleh siswa) bernilai sederhana karena untuk proses mengeksport file tersebut hanya perlu menekan tombol expor untuk mendownload file .excel dari tabel nilai siswa yang ditampilkan.
14. Hasil menunjukkan pada fungsionalitas dari No. 14 dengan tipe *output* pada fitur mengeksport pdf (fitur yang dilakukan oleh siswa) bernilai sederhana karena untuk proses mengeksport file tersebut hanya perlu menekan tombol expor untuk mendownload file .pdf dari tabel nilai siswa yang ditampilkan.
15. Hasil menunjukkan pada fungsionalitas dari No. 15 dengan tipe *view* pada fitur melihat tampilan tabel informasi data siswa (Fitur yang dapat diakses oleh semua tipe akun) bernilai sederhana karena tampilan pada fitur ini menggunakan tabel untuk menampilkan output dari data nilai siswa yang diinput.
16. Hasil menunjukkan pada fungsionalitas dari No. 16 dengan tipe *view* pada fitur melihat tampilan diagram informasi data siswa (Fitur yang dapat diakses oleh semua tipe akun) bernilai sederhana karena tampilan diagram yang digunakan pada AASBW menggunakan chart js.
17. Hasil menunjukkan pada fungsionalitas dari No. 17 dengan tipe *view* pada fitur melihat data informasi kehadiran/absen siswa (Fitur yang dapat diakses oleh semua tipe akun) bernilai sederhana karena tampilan pada fitur ini akan ditambahkan pada kolom tabel yang menampilkan output dari data nilai siswa.
18. Hasil menunjukkan pada fungsionalitas dari No. 18 dengan *view* pada fitur melihat informasi artikel yang tersedia dalam aplikasi (Fitur yang dapat diakses oleh semua tipe akun) bernilai sederhana karena tampilan pada fitur ini untuk menampilkan artikel ditampilkan pada bagian isi konten dari AASBW.
19. Hasil menunjukkan pada fungsionalitas dari No. 19 dengan tipe *query* pada fitur login akun super admin bernilai kompleksitas karena pada fitur ini jika dilihat dari sisi pengembang mengatur akun untuk login memiliki aturan proses yang rumit untuk menampilkan fitur login maupun registrasi yang ada pada aplikasi AASBW.
20. Hasil menunjukkan pada fungsionalitas dari No. 20 dengan tipe *query* pada fitur login akun admin bernilai kompleksitas karena pada fitur ini jika dilihat dari sisi pengembang mengatur akun untuk login memiliki aturan proses yang rumit untuk menampilkan fitur login maupun registrasi yang ada pada aplikasi AASBW.
21. Hasil menunjukkan pada fungsionalitas dari No. 21 dengan tipe *query* pada fitur login akun siswa bernilai kompleksitas karena pada fitur ini jika dilihat dari sisi pengembang mengatur akun untuk login memiliki aturan proses yang rumit untuk menampilkan fitur login maupun registrasi yang ada pada aplikasi AASBW.
22. Hasil menunjukkan pada fungsionalitas dari No. 22 dengan tipe *interface external* pada fitur pendaftaran siswa online (Fitur yang dapat diakses oleh siswa) bernilai menengah karena untuk pendaftaran siswa online pada AASBW dilakukan pada diluar sistem fungsionalitas dari aplikasi yang dimaksud.
23. Hasil menunjukkan pada fungsionalitas dari No. 23 dengan tipe *search* pada fitur mencari data siswa pada tabel bernilai kompleksitas karena pada fitur ini jika dilihat dari sisi pengembang untuk menampilkan kolom pencarian pada tabel memiliki aturan proses yang rumit untuk menampilkan fitur ini pada sisi klien.
24. Hasil menunjukkan pada fungsionalitas dari No. 24 dengan tipe *search* pada fitur memunculkan paginasi bernilai kompleksitas karena pada fitur ini jika dilihat dari sisi pengembang untuk menampilkan paginasi pada tabel memiliki aturan proses yang rumit untuk menampilkan fitur ini pada sisi klien.
25. Hasil menunjukkan pada fungsionalitas dari No. 25 dengan tipe *file* pada hasil export pdf berupa file bernilai menengah karena memiliki tampilan berupa tabel yang mirip dengan yang ada pada AASBW.
26. Hasil menunjukkan pada fungsionalitas dari No. 26 dengan tipe *file* pada hasil export excel berupa file bernilai menengah karena memiliki tampilan tabel yang dapat diubah nilainya melalui ms office excel/ mirip dengan yang ada pada AASBW.
27. Hasil menunjukkan pada fungsionalitas dari No. 27 dengan tipe *table* pada hasil tabel yang berisi data nilai siswa bernilai menengah karena memiliki fitur yang mendukung fungsionalitas pada tabel ini yaitu kolom searching dan paginasi.
28. Hasil menunjukkan pada fungsionalitas dari No. 28 dengan tipe *table* pada hasil tabel yang berisi data kehadiran siswa bernilai menengah karena memiliki fitur yang mendukung fungsionalitas pada tabel ini yaitu kolom searching dan paginasi.
29. Hasil menunjukkan pada fungsionalitas dari No. 29 dengan tipe *graphic* pada hasil grafik yang berbentuk lingkaran bernilai menengah karena memiliki fitur sorting untuk informasi nilai tertinggi dan pemilik nilai tersebut pada diagram lingkaran.

30. Hasil menunjukkan pada fungsionalitas dari No. 30 dengan tipe *graphic* pada hasil grafik yang berbentuk batang bernilai menengah karena memiliki fitur sorting untuk informasi nilai tertinggi dan pemilik nilai tersebut pada diagram tabang.

3.1.3 Hasil dan Analisa dari Faktor *Analysis*

Selanjutnya, pada tahap ini bertujuan untuk menghitung hasil dari pemetaan yang telah dilakukan, penulis telah memasukkan hasil perhitungannya ke dalam tabel perhitungan FP yang sebelumnya telah penulis tunjukkan pada tabel 3. Dari penurunan tabel tersebut didapatkan pemumusan yang berfungsi untuk menghasilkan nilai dari *Crude Function Point* (CFP).

Tabel 7. Perhitungan CFP untuk AASBW

Komponen Sistem Perangkat Lunak	Tingkat Kompleksitas									Total CFP
	Sederhana			Menengah			Kompleks			
	Jumlah	Bobot	Point	Jumlah	Bobot	Point	Jumlah	Bobot	Point	
	A	B	C=A*B	A	B	C=A*B	A	B	C=A*B	
Tipe Input	1	3	3	10	4	40	0	6	0	43
Tipe Output	2	2	4	0	3	0	0	7	0	4
Tipe Query/Search/View	5	4	20	0	5	0	5	7	35	55
Tipe File/Table/Database/Graphic	0	2	0	6	5	30	0	15	0	30
Tipe External Interface	0	6	0	1	7	7	0	10	0	7
Total CFP										139

Berdasarkan perhitungan CFP dari tabel 4 dengan memperoleh nilai total 139 dan hasil yang terbesar terletak pada komponen *query/search/view*. Apabila dilihat dari tingkat kompleksitasnya banyak yang masuk kedalam kategori menengah. Tingkat kompleksitas ini dikur secara subyektif masing-masing sesuai dengan konsep awal dari FP. Subyektifitas ini dilakukan untuk mengamsusikan tiap-tiap komponen pada AASBW telah dihitung oleh penulis (mahasiswa dengan program studi Informatika semester 4 serta dosen pembimbing), sehingga dapat dikategorikan dalam level menengah untuk mengoperasikan perangkat lunak yang telah dikembangkan sendiri oleh penulis.

Nilai kompleksitas ini telah diuji maupun dijalankan oleh 5 orang yang kemudian hasilnya dirata-rata untuk mendapatkan nilai value dari tiap-tiap komponen yang hasilnya dijumlahkan untuk menghasilkan nilai dari *Relative Complexity Adjustment Factor* (RCAF).

Tabel 8. Perhitungan RCAF untuk AASBW

No.	Karakteristik	Bobot						
1.	<i>Data Communication</i>	0	1	2	3	4	5	
2.	<i>Distributed Data Processing</i>	0	1	2	3	4	5	
3.	<i>Performance</i>	0	1	2	3	4	5	
4.	<i>Heavily used configuration</i>	0	1	2	3	4	5	
5.	<i>Transaction rate</i>	0	1	2	3	4	5	

6.	<i>Online Data Entry</i>	0	1	2	3	4	5
7.	<i>End-user efficienc</i>	0	1	2	3	4	5
8.	<i>Online Update</i>	0	1	2	3	4	5
9.	<i>Complex Process</i>	0	1	2	3	4	5
10.	<i>Reusability</i>	0	1	2	3	4	5
11.	<i>Installation Ease</i>	0	1	2	3	4	5
12.	<i>Operational Ease</i>	0	1	2	3	4	5
13.	<i>Multiple Sites</i>	0	1	2	3	4	5
14.	<i>Facilitate Change</i>	0	1	2	3	4	5
Total RCAF		45					

Tabel diatas menunjukkan hasil dari beberapa karakteristik pada AASBW untuk menilai tiap-tiap bobot tersebut pada level karakteristiknya masing-masing, berikut telah penulis rincikan tiap-tiap definisinya sebagai berikut:

1. Hasil menunjukkan pada karakteristik dari No.1 tentang *Data Communication* menjelaskan bahwa bobot bernilai 1 dikarenakan tingkat komunikasi data yang belum saling membutuhkan antara aplikasi dan processor.
2. Hasil menunjukkan pada karakteristik dari No.2 tentang *Distributed Data Processing* menjelaskan bahwa bobot bernilai 4 dikarenakan memiliki keterkaitan penting antara transfer data antara komponen satu dengan komponen lainnya.
3. Hasil menunjukkan pada karakteristik dari No.3 tentang *Performance* menjelaskan bahwa bobot bernilai 4 karena memiliki response time yang baik saat dilakukan percobaan aplikasi.
4. Hasil menunjukkan pada karakteristik dari No.4 tentang *heavily Used Configuration* menjelaskan bahwa bobot bernilai 4 dikarenakan tingkat konfigurasi komputer termasuk baik terhadap pengembangan aplikasi.
5. Hasil menunjukkan pada karakteristik dari No.5 tentang *Transaction Rate* menjelaskan bahwa bobot bernilai 3 dikarenakan tingkat rata-rata kecepatan transaksi termasuk cepat terhadap pengembangan aplikasi.
6. Hasil menunjukkan pada karakteristik dari No.6 tentang *Online Data Entry* menjelaskan bahwa bobot bernilai 4 dikarenakan kebutuhan untuk berbagai penginputan data pada aplikasi sangat beragam dan interaktif.
7. Hasil menunjukkan pada karakteristik dari No.7 tentang *End-user Effiience* menjelaskan bahwa bobot bernilai 4 dikarenakan tingkat yang mudah dan sederhana dalam penggunaan pada aplikasi.
8. Hasil menunjukkan pada karakteristik dari No.8 tentang *Online Update* menjelaskan bahwa bobot bernilai 2 dikarenakan tingkat kebutuhan ILF yang sukar dilakukan update secara online pada aplikasi.
9. Hasil menunjukkan pada karakteristik dari No.9 tentang *Complex Process* menjelaskan bahwa bobot bernilai 4 dikarenakan seberapa signifikan tingkat logika yang digunakan dalam proses pengembangan pada aplikasi.
10. Hasil menunjukkan pada karakteristik dari No.10 tentang *Reusability* menjelaskan bahwa bobot bernilai 3 dikarenakan sukarnya kode program pada aplikasi bisa dikembangkan ataupun digunakan kembali pada aplikasi yang berbeda.
11. Hasil menunjukkan pada karakteristik dari No.11 tentang *Installation Ease* menjelaskan bahwa bobot bernilai 3 dikarenakan jaranganya kode program untuk di kembangkan ataupun digunakan kembali dapat memengaruhi kemudahan untuk menkonversikan sistem lama menjadi baru.
12. Hasil menunjukkan pada karakteristik dari No.12 tentang *Operational Ease* menjelaskan bahwa bobot bernilai 3 dikarenakan pada aplikasi pendataan untuk startup minim digunakan namun pada backup dan proses recovery pada data siswa sangat layak untuk digunakan.
13. Hasil menunjukkan pada karakteristik dari No.13 tentang *Multiple Sites* menjelaskan bahwa bobot bernilai 4 dikarenakan fleksibilitas aplikasi berbasis website untuk dioperasikan maupun dijalankan pada hardware ataupun software yang berbeda-beda.
14. Hasil menunjukkan pada karakteristik dari No.14 tentang *Facilitate Change* menjelaskan bahwa bobot bernilai 3 dikarenakan banyaknya fitur yang meliputi mengatur beberapa jenis akun, mendata nilai dan kehadiran, menampilkan diagram batang maupun lingkaran pada aplikasi dapat mengurangi tingkat kemudahan pada modifikasi logika serta proses ataupun struktur data.

Berdasarkan perhitungan RCAF dengan menggunakan 14 karakteristik dari tabel 8 memperoleh total nilai 45 dan *value* yang paling banyak didapatkan adalah 3 dan 4 dengan menggunakan komponen perhitungan yang meliputi *input*, *output*, *online query*, *logic file* dan *external interface*.

Dalam hal ini menunjukkan bahwa Aplikasi Akademik Berbasis Website yang telah dipilih penulis untuk dijadikan studi kasus masuk kedalam level menengah untuk dioperasikan pada ruang lingkup sekolah.

Menghitung Function Point

Setelah proses kalkulasi perhitungan dari CFP maupun RCAF telah selesai dilakukan, maka tahapan selanjutnya dilakukanlah perhitungan FP, penulis menggunakan pedoman penilaian pada Tabel 7 dan Tabel 8.

$$\begin{aligned}
 FP &= CFP \times (0,65 + 0,01 \times RCAF) \\
 &= 139 \times (0,65 + (0,01 \times 45)) \\
 &= 139 \times (0,65 + 0,45) \\
 &= 139 \times 1,10 \\
 &= 152,90
 \end{aligned}$$

Sehingga didapatkan nilai *Functional Point* dari hasil perhitungan pada Aplikasi Akademik Siswa Berbasis Website adalah **152,90**.

Menghitung Function Point dengan Program Perangkat Lunak

Sebelum melakukan perhitungan FP secara manual, penulis membuat sebuah perangkat lunak untuk membandingkan hasil perhitungan dengan perangkat lunak dan perhitungan secara manual. Penulis menggunakan HTML, CSS, framework, dan Javascript. Berikut tampilan untuk program pengukurannya,

Penilaian Function Poin										
Tabel CFP										
Komponen	Tingkat Kompleksitas									CFP
	Sederhana			Menengah			Kompleksitas			
	Jumlah	Bobot	Poin	Jumlah	Bobot	Poin	Jumlah	Bobot	Poin	
Input (EI)	<input type="text" value="1"/>	3	3	<input type="text" value="10"/>	4	40	<input type="text" value="0"/>	6	0	43
Output (EO)	<input type="text" value="2"/>	2	4	<input type="text" value="0"/>	3	0	<input type="text" value="0"/>	7	0	4
Query/Search/View (EQ)	<input type="text" value="5"/>	4	20	<input type="text" value="0"/>	5	0	<input type="text" value="5"/>	7	35	55
Internal Logical (ILF)	<input type="text" value="0"/>	2	0	<input type="text" value="6"/>	5	30	<input type="text" value="0"/>	15	0	30
External Interface (EIF)	<input type="text" value="0"/>	6	0	<input type="text" value="1"/>	7	7	<input type="text" value="0"/>	10	0	7
Total CFP										139

Gambar 5 Perhitungan CFP dengan Perangkat Lunak

Tabel RCAF		
No.	Karakteristik	Bobot
1.	Data Communication	1
2.	Distributed Data Processing	4
3.	Performance	4
4.	Heavily used configuration	3
5.	Transaction rate	4
6.	Online Data Entry	4
7.	End-user efficiency	3
8.	Online Update	2
9.	Complex Process	4
10.	Reusability	3
11.	Installation Ease	3
12.	Operational Ease	3
13.	Multiple Sites	4
14.	Facilitate Change	3
Total		45

$$FP = CFP * (0.65 + 0.01 * RCAF)$$

$$FP = 139 * (0.65 + 0.01 * 45)$$

$$FP = 139 * (0.65 + (0.01 * 45))$$

$$FP = 139 * (0.65 + 0.45)$$

$$FP = 139 * (1.1)$$

$$FP = 152.9$$

Gambar 6 Perhitungan FP dengan Perangkat Lunak

3.1.4 Hasil dan Analisa dari Faktor *Interpretation*

Metode penggunaan FP dalam hal penggunaannya sebagai salah satu pedoman ataupun acuan untuk mendapatkan nilai metrik atas sesuatu dengan menkonversikan nilai FP ke dalam sebuah metrik yang diinginkan oleh penulis dalam studi kasus terhadap Aplikasi Akademik Siswa berbasis Website.

Penkonversian Penggunaan Function Point

A. Estimasi Biaya Pemasaran

Diasumsikan, setiap *Function Point*, dikenakan biaya Rp. 50.000 dan *Function Point* pada Aplikasi Akademik Siswa Berbasis Website adalah 152, 90 [17]. Sehingga dapat diestimasikan untuk biaya yang diperlukan untuk memasarkan aplikasi yang dimaksud adalah Rp. 50.000 x 152, 90 = Rp. 7.645.000.

B. Estimasi Waktu Produksi

Dalam estimasi ini, diasumsikan proyek ini dikerjakan paling lama 200 jam / 152, 90 FP (1,5 jam/FP) dan dikerjakan oleh 5 pengembang. Masing-masing pengembang mengerjakan 40 jam (200/5) [4].

Sedangkan untuk estimasi waktu produksi, penulis dapat menggunakan perhitungan seperti berikut:

$$1,5 \text{ jam} \times 152, 90 = 230 \text{ jam (pembulatan)}$$

$$= 230/8$$

$$\text{Hasil} = 29$$

Hasil dari angka di atas sama dengan 29 hari bekerja (dengan estimasi 1 hari bekerja selama 8 jam). Jadi dalam satu hari, seorang pengembang memiliki waktu sebanyak 8 jam untuk mengerjakan Aplikasi Akademik Siswa Berbasis Website.

Manajemen Proyek

A. Estimasi Biaya Komponen dalam Manajemen Perangkat Lunak

Untuk memperkirakan biaya yang akan dikeluarkan untuk produksi aplikasi ini, tim pengembang juga dapat melihat kebutuhan fungsional pada aplikasi ini. Dikarenakan aplikasi ini merupakan aplikasi yang dibuat khusus dalam ruang lingkup sekolah. Sedangkan untuk pengerjaan aplikasi ini paling lama dikerjakan dalam waktu 200 jam dan menggunakan pengukuran faktor produktivitas, maka untuk hasil biaya komponen dalam manajemen untuk aplikasi ini ditunjukkan pada tabel 9 [4].

Tabel 9. Produktivitas untuk AASBW

No	Peran	Hari	Biaya/Hari	Jumlah
1.	Pengelola Proyek	22	Rp. 50,000	Rp. 1,100,000
2.	System Analyst	22	Rp. 45,000	Rp. 990,000
3.	Pengembang Spesialis	22	Rp. 55,000	Rp. 1,210,000
4.	Keamanan	15	Rp. 30,000	Rp. 450,000
5.	Tester	4	Rp. 25,000	Rp. 100,000
Total				Rp. 3,850,000

Teknik estimasi ini baik digunakan untuk merencanakan alokasi tim pengembang proyek perangkat lunak dan juga dapat digunakan untuk memperkirakan berapa jam yang dibutuhkan oleh masing-masing peran yang diwakili langsung oleh penulis selaku tim pengembang dari Aplikasi Akademik Siswa Berbasis Website yang berjumlah sebanyak lima orang.

3.1.5 Hasil dan Analisa dari Faktor *Feedback*

Dalam halnya penyampaian *feedback*, atau umpan balik untuk studi kasus ini, diberikan kepada pihak pengembang aplikasi selaku penulis yang meliputi: 4 orang mahasiswa dengan program studi informatika serta dosen pembimbing sebagai tim pengembang maupun pengelola seluruh sistem dari Aplikasi Akademik Siswa berbasis Website yang dijadikan sebagai objek studi kasus.

IV. KESIMPULAN

Berdasarkan analisis studi kasus yang sudah penulis lakukan pada tulisan ini, penulis menyimpulkan bahwa *Function Point* dapat digunakan untuk menilai ataupun mengukur aplikasi yang dikembangkan, dalam implementasinya, sebagai salah satu cara untuk mengukur sebuah perangkat lunak. *Function Point* ini membutuhkan penilaian dalam estimasi waktu, biaya pemasaran, maupun biaya manajemen pada perangkat lunak yang telah dikembangkan oleh penulis sehingga melibatkan penilaian yang sangat subjektif. Oleh karena itu untuk mengurangi faktor subjektifitas dalam analisis studi kasus ini, penulis menggunakan metode perhitungan FP yang mengacu pada [edoman dari *Function Point International User Group* (FPIUG)]. Dalam pengukuran estimasi pada Aplikasi Akademik Siswa Berbasis Website didapatkan nilai FP sebesar 152,90. Nilai ini kemudian dapat penulis konversikan sehingga dapat diturunkan untuk beberapa fungsi pengukuran, seperti estimasi waktu, biaya, maupun biaya manajemen yang dibutuhkan untuk mengukur perangkat lunak tersebut.

DAFTAR PUSTAKA

- [1] Sun-Jen Huang, Chieh-Yi Lin, NanHsing Chiu, "Fuzzy Decision Tree Approach for Embedding Risk Assessment Information into Software Cost Estimation Model", *Journal of Information Science and Engineering*, pp. 297-313, 2006.
- [2] Vishal Sharma, Harsh Kumar Verma, "Optimized Fuzzy Logic based Framework for Effort Estimation in software Development", *International Journal of Computer Science*, Pp 30- 38, 2010.
- [3] Jyoti Mahajan, D., Dhruve, K., "REBEE – Reusability Based Effort Estimation Technique Using Dynamic Neural Network", *Global Journa Computer Science Technology*, 11, (7), pp. 55–60, 2011.
- [4] Fityan Aula Juyuspan, Anita Hidayati. "Estimasi Pengelolaan Suatu Proyek dalam Pengembangan Perangkat Lunak menggunakan Analisa Function Point", *Jurnal Simantec*, Vol.5 , No. 2 Juni 2016

- [5] Deng, J.D., Purvis, M.K., Purvis, M.A., "Software Effort Estimation: Harmonizing Algorithms and Domain Knowledge in An Integrated Data Mining Approach" *Inf. Sci. Discuss. Pap. Ser.*, (5), pp. 1–13, 2009.
- [6] Mittal, H., Bhatia, P., " A Comparative Study of Conventional Effort Estimation and Fuzzy Effort Estimation Based On Triangular Fuzzy Numbers", *International Journal Computer Science Security.*, (4), pp. 36–47, 2002.
- [7] Benton, A., Bradly, M, The International Function Point User Group (IFPUG) in *Function Point Counting Practices Manual – release 4.1'*. (SA), 1999.
- [8] Schatzberg, D.R., *Total Quality Management for Maintenance Process Improvement*, Journal Software Maintenance. R
- [9] Alvin Alexander, *How I estimate Software Development Project*, 2014
- [10] K. E. Wiegers and J. Beatty, *Software Requirements*, 3rd ed. Washington: Microsoft Press, 2013.
- [11] Winangsari Pradani, "Kajian Metode Perhitungan Metrik *Function-Point* dan Penerapannya pada Dua Perangkat Lunak yang Dipilih", *Jurnal AL-AZHAR INDONESIA SERI SAINS DAN TEKNOLOGI*, Vol. 2, No.1, Maret 2013.
- [12] Sandhea, Annisa Azzar, Maulidiyah, Sri, & Sari, Renny Dewi. (2020). Estimasi Biaya Perangkat Lunak Pada Aplikasi SIBIMA Universitas XYZ dengan Menggunakan Metode Function Point. *JURIKOM (Jurnal Riset Komputer)*, 7(1), 41–49
- [13] Cah, "The Function Point Method", Pearson Education Limited, Retrievable from www.cs.nott.ac.uk, 2004
- [14] J. Kh, S. Iskandar, K. Badak, T. Sereal, K. Bogor, and J. Barat, "PENGUKURAN KUALITAS PERANGKAT LUNAK SISTEM E-LEARNING Program Studi Teknik Informatika , Fakultas Teknik , Universitas Ibn Khaldun Bogor Program Studi Teknik Elektro , Fakultas Teknik , Universitas Ibn Khaldun Bogor Pencapaian kualitas perangkat lunak dap," *Pros. SNATIF*, vol. 4, pp. 769–776, 2017.
- [15] Pressman, Roger S. 2010. "SOFTWARE ENGINEERING- A Practitioner's Approach - Seventh Edition", Penerbit McGraw-Hill Companies, New York: 889 hlm
- [16] Akbar, M.; Sukmana, H.T.; Khairani, D., "Models and software measurement using Goal/Question/Metric method and CMS Matrix parameter (Case study discussion forum)," in *Cyber and IT Service Management (CITSM)*, 2014 International Conference on , vol., no., pp.34-38, 3-6 Nov. 2014 doi: 10.1109/CITSM.2014.7042171
- [17] Dewi Khairani, "Studi Kasus Pengukuran Sistem Informasi menggunakan Function Point (FP)", *JURNAL TEKNIK INFORMATIKA VOL. 8 NO. 2 OKTOBER 2015*.