

PENENTUAN EFFORT RATE PADA ESTIMASI EFFORT MENGUNAKAN METODE USE CASE POINT UNTUK PENGEMBANGAN PERANGKAT LUNAK WEBSITE KEPEMERINTAHAN

Wahyu Kurniawan¹⁾ Sholih²⁾ Teguh Sutanto³⁾

Program Studi/Jurusan Sistem Informasi

STMIK STIKOM Surabaya

Jl. Raya Kedung Baruk 98 Surabaya, 60298

Email : 1)sayawahyukurniawan@yahoo.com, 2)sholih@is.its.ac.id, 3)teguh@stikom.edu

Abstract: *Effort estimation using Use Case Point (UCP) method obtained from the multiplication of the UCP with the Effort Rate (ER). Some studies conclude that there is some value of ER used in the calculation of effort estimation. However, the most often used is the ER with value 20 man-hours given by Karner. Disputes about the research that is ER by Karner occurred in 1993, so it is possible that the ER by Karner not suitable when applied in the calculation of effort estimation for the project worked on in 2013 and the years to come.*

This study used 8 data of governance website software development projects. The data is used to calculate actual effort, making use case diagrams, and calculations UCP estimation. Further level of analysis that is performed correlation and regression equations for actual effort and the UCP estimation. Based on the regression equation then performed calculations tangent θ to produce the ER.

From this research produced the ER as big as 5,178 man-hours. The ER is much smaller than the value given by Karner ER. Case is possible for several reasons, among others: 1) Engineering software technology that is growing rapidly. 2) Manufacturing website using components. 3) Source of the internet so complete.

Keywords: *Effort Rate (ER), Effort Estimation, Use Case Point (UCP), Governance Website*

Tingkat kegagalan pada proyek pengembangan perangkat lunak sangat tinggi. Pada tahun 2002 sampai 2010 hanya terdapat 37% proyek teknologi informasi yang sukses (Standish Group, 2011). Salah satu prosentase terbesar yang menyebabkan kegagalan proyek pengembangan perangkat lunak yaitu kurang baiknya perencanaan proyek, yakni mempunyai prosentase sebesar 39% (BULLS, 1998). Pendapat senada juga didapatkan dari survey yang dilakukan oleh KPMG Canada. Salah satu penyebab utama kegagalan pelaksanaan proyek pengembangan perangkat lunak yaitu buruknya perencanaan proyek (KPMG Canada, 1997).

Melihat begitu besarnya kegagalan proyek yang terjadi akibat buruknya

perencanaan proyek, maka perlu dilakukan perbaikan dalam perencanaan proyek. Salah satu cara untuk memperbaiki perencanaan proyek yaitu dengan melakukan estimasi usaha (*effort*) pada proyek pengembangan perangkat lunak menggunakan metode yang tepat. Definisi dari estimasi perangkat lunak yaitu suatu kegiatan melakukan prediksi atau ramalan mengenai keluaran dari sebuah proyek dengan meninjau jadwal, usaha, biaya bahkan hingga ke resiko yang akan ditanggung dalam proyek tersebut (Galorath, 2006). Metode *use case point* (UCP) adalah metode yang mempunyai kemampuan untuk memberikan estimasi *effort* yang diperlukan untuk membuat suatu proyek berdasarkan jumlah dan kompleksitas *use case*

yang dimiliki oleh proyek perangkat lunak tersebut (Karner, 1993). Studi yang telah dilakukan oleh beberapa peneliti membuktikan bahwa metode UCP lebih baik dari perkiraan para ahli, seperti berikut :

1. Perbandingan estimasi *effort* dengan upaya yang sebenarnya menggunakan metode UCP memiliki deviasi sebesar 19%, sementara estimasi para ahli memiliki deviasi sebesar 20% (Anda, 2002).
2. Penelitian lain menunjukkan terjadi deviasi sebesar 6% (Nageswaran, 2001).
3. Pendapat terakhir menunjukkan terjadi deviasi sebesar 9% (Carroll, 2005).

Dari beberapa penelitian di atas menunjukkan bahwa metode UCP merupakan metode yang tepat untuk melakukan estimasi *effort*.

Dalam metode UCP, estimasi *effort* didapatkan dari perkalian antara nilai UCP dengan nilai *Effort Rate* (ER). Dimana estimasi *effort* nantinya akan menghasilkan jumlah pegawai (*man-hours* atau *man-days*) yang dibutuhkan dalam pembuatan proyek perangkat lunak (Muhardin, 2011). Nilai ER pertama kali yaitu 20 *man-hours* (Karner, 1993).

Beberapa penelitian menyimpulkan nilai ER yang digunakan oleh para peneliti memiliki variasi dengan dasar penentuan yang berbeda - beda, seperti uraian berikut :

1. Nilai *effort rate use case point* sebesar 20 *man-hours* dengan menggunakan tiga data proyek pengembangan perangkat lunak (Karner, 1993).
2. Nilai *effort rate use case point* sebesar 20, 24, dan 36 *man-hours* menggunakan dasar kompleksitas proyek dengan mengacu pada *Technical Complexity Factor* (Schneider, 1998).
3. Nilai *effort rate use case point* berkisar antara 15 sampai 30 *man-hours* dengan menggunakan dasar kualitas personil tim dan data historis (Clemmons, 2006).
4. Nilai *effort rate use case point* berkisar antara 4 sampai 35 *man-hours* yang dihitung dari 14 proyek perangkat lunak yang telah selesai (Ochodek, 2011).

Dari beberapa nilai *effort rate* yang tercantum di atas, nilai *effort rate* yang paling sering digunakan untuk perhitungan estimasi *effort* yaitu nilai 20 *man-hours* sesuai temuan dari Karner. Penelitian yang menggunakan nilai *effort rate* dari Karner terdapat pada beberapa penelitian, antara lain yaitu penelitian seperti

berikut ((Nageswaran, 2001), (Damodaran, 2002), (Kasumoto, 2006), (Frohnhoff, 2008), dan (Monteiro dkk, 2008)).

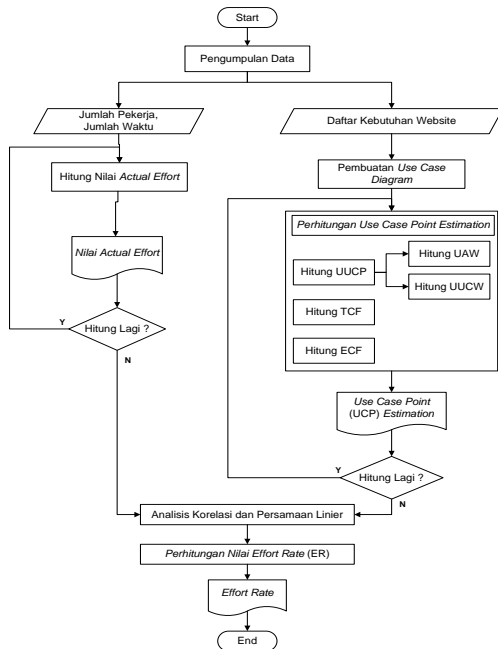
Permasalahan yang timbul yaitu penelitian tentang nilai *effort rate* yang dilakukan oleh Karner hanya menggunakan tiga data proyek pengembangan perangkat lunak dengan menggunakan analisis regresi. Analisis regresi dengan menggunakan tiga data diskrit cenderung tidak akurat. Analisis korelasi antar data untuk membentuk persamaan regresi juga tidak dilakukan. Selain itu, penelitian perhitungan *effort rate* yang dilakukan oleh Karner terjadi pada tahun 1993. Teknologi informasi dalam rentang waktu 1993 sampai 2013 mengalami perkembangan yang cukup pesat, sehingga sangat dimungkinkan nilai *effort rate* yang ditemukan oleh Karner tidak sesuai apabila diaplikasikan dalam perhitungan estimasi *effort* untuk proyek pengembangan perangkat lunak yang dikerjakan pada tahun 2013 dan tahun-tahun mendatang. Berdasarkan penjabaran di atas maka nilai ER yang diusulkan oleh Karner dapat dipertanyakan dan ditinjau ulang.

Penelitian penentuan nilai *effort rate* merupakan proyek penelitian yang dilakukan dengan menggunakan beragam proyek perangkat lunak. Penelitian dibagi meliputi perhitungan nilai ER pada proyek pengembangan perangkat lunak Pemerintahan, Pendidikan, dibidang Bisnis dan proyek pengembangan Website Kepemerintahan. Pada penelitian tugas akhir ini melakukan perhitungan nilai ER untuk proyek pengembangan perangkat lunak website pemerintahan.

Hasil akhir yang diharapkan dari penelitian tugas akhir ini berupa nilai ER pada metode UCP untuk estimasi *effort* proyek pengembangan perangkat lunak website pemerintahan, yang kemudian nilai ER tersebut dapat dijadikan acuan bagi pengembang perangkat lunak untuk melakukan *estimasi effort* dalam proyek pengembangan perangkat lunak website pemerintahan di masa mendatang.

METODE

Tahapan-tahapan pengerjaan penelitian tugas akhir ini akan ditunjukkan melalui gambar 1 seperti berikut:



Gambar 1. Tahapan Penelitian Tugas Akhir

Pengumpulan Data

Pengumpulan data dilakukan dengan berbagai cara, antara lain dengan wawancara, survei atau observasi, dan kuisioner. Pengumpulan data dilakukan dengan tujuan untuk mendapatkan data-data yang dibutuhkan dalam pengerjaan tugas akhir ini. Adapun data yang dibutuhkan pada tugas akhir ini dibagi menjadi 3 data seperti berikut :

1. Jumlah Pekerja dan jumlah waktu kerja: jumlah pekerja dan jumlah waktu kerja digunakan sebagai bahan untuk perhitungan nilai *actual effort*.
2. Daftar kebutuhan website : yaitu daftar yang berisi data *use case* dan *actor* apa saja yang terdapat pada masing – masing website.
3. Nilai *technical factor* dan nilai *environmental factor*

Perhitungan Nilai Actual Effort

Nilai *actual effort* adalah nilai yang dihasilkan dari banyaknya jumlah pegawai dan jumlah waktu yang diperlukan untuk mengerjakan proyek perangkat lunak. Data yang

dibutuhkan yaitu banyaknya jumlah pekerja dan lama waktu pengerjaan yang dibutuhkan oleh pengembang untuk menyelesaikan proyek perangkat lunak website pemerintahan tersebut.

Setelah didapatkan jumlah pekerja dan jumlah waktu pengerjaan proyek perangkat lunak website pemerintahan, maka selanjutnya dapat dihitung nilai *actual effort* untuk proyek pembuatan perangkat lunak website pemerintahan dengan rumus, sebagai berikut :

$$Actual\ Effort = \sum\ Pekerja \times \sum\ Jam\ Kerja \dots (1)$$

Pembuatan Use Case Diagram

Setelah didapatkan daftar kebutuhan pembuatan website, maka dapat dilakukan proses pembuatan *use case diagram*. Daftar kebutuhan tersebut berisi *use case* dan *actor* yang dibutuhkan pada proyek pembuatan perangkat lunak website pemerintahan. Untuk memudahkan proses pengerjaan dan pendokumentasian *use case diagram*, maka digunakan alat bantu atau *tools* yakni *Enterprise Architect 7.5*.

Perhitungan Unadjusted Use Case Point (UUCP)

Untuk mendapatkan nilai UUCP, maka perlu dilakukan pembobotan dan skoring terkait kompleksitas *actor* dan *use case* ditinjau dari *use case diagram* yang telah dibuat pada tahap sebelumnya. Skoring dihitung berdasarkan parameter-parameter yang telah ditentukan.

Terdapat dua langkah yang dilakukan untuk menghitung UUCP, antara lain sebagai berikut:

1. Menghitung *Unadjusted Actor Weights* (UAW)
 2. Menghitung *Unadjusted Use Case Weights* (UUCW)
- a). Perhitungan *Unadjusted Actor Weights* (UAW)

Perhitungan UAW dilakukan untuk menghitung jumlah bobot *actor* yang terlibat pada pembuatan proyek perangkat lunak website pemerintahan. Perhitungan bobot *actor* dilakukan dengan cara mengklasifikasikan masing-masing *actor* ke dalam masing-masing bobot yang telah ditentukan. Setelah didapatkan masing-masing bobot pada tiap *actor*, kemudian dilakukan akumulasi dari seluruh nilai bobot

actor, sehingga menghasilkan nilai UAW. Rumus perhitungan UAW yaitu :

$$UAW = \text{Jumlah Actor} \times \text{Bobot Actor} \dots\dots (2)$$

b). Perhitungan *Unadjusted Use Case Weights* (UUCW)

Perhitungan UUCW dilakukan untuk menghitung jumlah bobot *use case* yang dibutuhkan pada pembuatan proyek perangkat lunak website pemerintahan. Setelah didapatkan masing-masing bobot pada tiap *use case*, kemudian dilakukan akumulasi dari seluruh nilai bobot *use case*, sehingga menghasilkan nilai UUCW. Rumus menghitung UUCW yaitu seperti berikut:

$$UUCW = \text{Jumlah Use Case} \times \text{Bobot Use Case} \dots\dots (3)$$

Setelah diketahui nilai UAW dan UUCW, maka selanjutnya dapat dilakukan perhitungan nilai UUCP dengan rumus berikut :

$$UUCP = UAW + UUCW \dots\dots\dots (4)$$

Menghitung *Technical Complexity Factor* (TCF)

Nilai TCF pada penelitian tugas akhir ini adalah nilai dari faktor teknis yang mempengaruhi proyek pembuatan perangkat lunak website pemerintahan. Setelah diketahui masing-masing faktor teknis dan besar bobot masing-masing faktor teknis, kemudian dilakukan pemberian nilai pada masing-masing faktor teknis. Nilai yang diberikan pada setiap faktor tergantung dari seberapa besar pengaruh dari faktor tersebut terhadap pengerjaan proyek pembuatan website perangkat lunak pemerintahan. Nilai 0 berarti tidak mempengaruhi, nilai 3 berarti rata-rata, dan nilai 5 berarti faktor teknis tersebut mempunyai pengaruh yang besar terhadap pengerjaan website pemerintahan tersebut. Hasil perkalian nilai dan bobot pada faktor teknis tersebut kemudian dijumlahkan untuk mendapatkan nilai total *Technical Factor* (TF), yang kemudian nilai tersebut digunakan untuk menghitung *Technical Complexity Factor* (TCF) dengan rumus sebagai berikut :

$$TCF = 0.6 + (0.01 \times TF) \dots\dots (5)$$

Menghitung *Enviromental Complexity Factor* (ECF)

Nilai ECF pada penelitian tugas akhir ini adalah nilai dari faktor lingkungan yang mempengaruhi proyek pembuatan perangkat

lunak website pemerintahan. Setelah diketahui masing-masing faktor lingkungan dan besar bobot masing-masing faktor lingkungan, kemudian dilakukan pemberian nilai pada masing-masing faktor lingkungan. Nilai yang diberikan pada setiap faktor tergantung dari seberapa besar pengaruh dari faktor tersebut terhadap pengerjaan proyek pembuatan website perangkat lunak pemerintahan. Nilai 0 berarti tidak mempengaruhi, nilai 3 berarti rata-rata, dan nilai 5 berarti faktor lingkungan tersebut mempunyai pengaruh yang besar terhadap pengerjaan website pemerintahan tersebut. Hasil perkalian nilai dan bobot pada faktor lingkungan tersebut kemudian dijumlahkan untuk mendapatkan nilai total *Environmental Factor* (EF), yang kemudian nilai tersebut digunakan untuk menghitung *Environmental Complexity Factor* (ECF) dengan rumus sebagai berikut :

$$ECF = 1.4 + (-0.03 \times EF) \dots\dots\dots (6)$$

Setelah diketahui nilai UUCP, nilai TCF, dan nilai ECF, kemudian dapat dilakukan perhitungan nilai UCP dengan rumus sebagai berikut :

$$UCP = UUCP + TCF + ECF \dots\dots\dots (7)$$

Analisis Korelasi dan Persamaan Regresi

Analisis korelasi dilakukan kepada variabel nilai *actual effort* dan nilai UCP menggunakan alat bantu *Statistical Package for the Social Sciences* (SPSS) v.17. Hasil dari analisis korelasi tersebut adalah angka dari 0 sampai 1. Apabila korelasi menunjukkan angka 0 berarti tidak terdapat korelasi antara variabel nilai *actual effort* dan variabel nilai UCP. Semakin mendekati angka 1, maka menunjukkan korelasi yang semakin kuat. Setelah dilakukan analisis korelasi, maka selanjutnya dilakukan pencarian nilai persamaan regresi. Persamaan regresi dilakukan kepada variabel nilai *actual effort* dan variabel nilai UCP. *Output* dari persamaan regresi yaitu menghasilkan suatu persamaan yang nantinya akan digunakan untuk mencari nilai empiris dari *effort rate*.

Perhitungan Nilai *Effort Rate*

Setelah diketahui persamaan regresi, maka selanjutnya dilakukan proses menghitung garis singgung. Langkah yang dilakukan yaitu menentukan nilai variabel x_1 dan x_2 . Kemudian

variabel tersebut dimasukkan kedalam rumus persamaan regresi seperti berikut:

$$y = a + bx \dots\dots\dots (8)$$

Perhitungan nilai y dimasukkan sebanyak 2 kali dengan nilai x yang berbeda, sehingga menghasilkan nilai garis singgung y1 dan y2 yang nantinya akan digunakan sebagai bahan untuk menghitung nilai tangen θ . Setelah didapatkan nilai garis singgung (x1, y1) dan (x2, y2) maka dapat dilakukan perhitungan nilai tangen θ . Rumus untuk menghitung tangen θ adalah sebagai berikut :

$$\text{Tangen } \theta = \frac{\Delta y}{\Delta x} = \frac{y_2 - y_1}{x_2 - x_1} \dots\dots\dots (9)$$

Hasil dari perhitungan nilai tangen θ tersebut merupakan nilai *effort rate* pada penelitian tugas akhir ini.

HASIL DAN PEMBAHASAN

Hasil Perhitungan Nilai Actual Effort

Nilai *actual effort* adalah nilai yang dibutuhkan oleh tim pengembang untuk menyelesaikan proyek dari awal proyek sampai selesainya proyek. Keseluruhan hasil perhitungan nilai *actual effort* proyek pengembangan website pemerintahan pada tugas akhir ini dapat dilihat pada tabel 1 seperti berikut:

Tabel 1. Nilai Actual Effort Proyek Pengembangan Website Pemerintahan

Kode Proyek	Jumlah Pekerja	Hari Kerja	Jam Kerja per Hari	Actual Effort
A	4	143	5	2.860
B	4	68	8	2.176
C	4	81	8	2.592
D	6	97	5	3.180
E	4	90	8	2.880
F	3	91	8	2.184
G	3	77	8	1.848
H	6	80	5	2.400

Hasil Perhitungan Unadjusted Use Case Point (UUCP)

1) Menghitung *Unadjusted Actor Weight* (UAW)

Semua *actor* yang terlibat berinteraksi dengan menggunakan web page, maka semua *actor* bersifat *complex*, sehingga setiap *actor* pada masing-masing proyek mempunyai bobot

sebesar 3. Keseluruhan hasil perhitungan nilai UAW pada tugas akhir ini dapat dilihat pada tabel 2 seperti berikut:

Tabel 2. Nilai UAW

	Kode Proyek							
	A	B	C	D	E	F	G	H
\sum Actor	5	4	3	4	4	4	3	4
\sum UAW	15	12	9	12	12	12	9	12

2). Menghitung *Unadjusted Use Case Weight* (UUCW)

Pada penelitian ini tidak terdapat *use case* yang memiliki transaksi lebih dari 7, sehingga tidak terdapat *use case* yang bertipe *complex*. Keseluruhan hasil perhitungan nilai UUCW dari keseluruhan proyek pengembangan website pemerintahan pada tugas akhir ini dapat dilihat pada tabel 3 seperti berikut:

Tabel 3. Nilai UUCW Proyek Pengembangan Website Pemerintahan

Tipe	Bobot	\sum Use Case							
		A	B	C	D	E	F	G	H
Simpl	5	50	33	23	45	31	28	13	34
Medi	10	17	10	14	11	15	15	14	13
\sum UUCW		420	265	255	335	305	290	205	300

Nilai UAW dan UUCP digunakan sebagai bahan menghitung nilai UUCP. Keseluruhan hasil perhitungan nilai UUCP dari keseluruhan proyek pengembangan website pemerintahan pada tugas akhir ini dapat dilihat pada tabel 4 seperti berikut:

Tabel 4. Nilai UUCP Proyek Pengembangan Website Pemerintahan

Variabel	Kode Proyek							
	A	B	C	D	E	F	G	H
UAW	15	12	9	12	12	12	9	12
UUCW	420	265	255	335	305	290	205	300
UUCP	435	277	264	347	317	302	214	312

Hasil Perhitungan Technical Complexity Factor (TCF)

Dalam perhitungan TCF diperlukan nilai dari masing – masing faktor teknis. Nilai dari faktor teknis ini didapatkan dari pihak pengembang proyek perangkat lunak. Kemudian

nilai pada *Technical Factor* (TF) tersebut dikalikan dengan bobot masing-masing faktor, kemudian dijumlah untuk mendapatkan TF, yang kemudian digunakan untuk mendapatkan nilai TCF. Keseluruhan hasil perhitungan nilai TCF proyek pengembangan website pemerintahan pada tugas akhir ini dapat dilihat pada tabel 5 seperti berikut :

Tabel 5. Nilai TCF Proyek Perangkat Lunak Website Pemerintahan

No	Technical Factor	Bobot (B)	Kode Proyek Perangkat Lunak															
			A		B		C		D		E		F		G		H	
			N ₁	BxN	N ₁	BxN	N ₁	BxN	N ₁	BxN	N ₁	BxN	N ₁	BxN	N ₁	BxN	N ₁	BxN
1.	Distributed System Required	2	4	8	5	10	4	8	5	10	4	8	4	8	4	8	5	10
2.	Response Timings Important	1	2	2	4	4	4	4	3	3	4	4	4	4	4	4	3	3
3.	End User Efficiency	1	4	4	3	3	3	3	3	3	3	3	3	4	4	4	3	3
4.	Complex Internal Processing Required	1	1	1	2	2	2	2	2	3	3	2	2	4	4	4	3	3
5.	Reusable Code Must Be A Focus	1	3	3	2	2	2	2	4	4	2	2	2	3	3	3	3	4
6.	Installation easy	0.5	4	2	2	1	2	1	3	1.5	2	1	3	1.5	3	1.5	3	1.5
7.	Usability	0.5	3	1.5	3	1.5	3	1.5	2.5	4	2	5	2.5	4	3	4	2	4
8.	Cross-platform support	2	1	2	4	8	2	4	2	4	2	4	2	4	2	4	2	4
9.	Easy to change	1	3	3	1	1	2	2	3	3	2	2	2	2	2	2	3	3
10.	Highly concurrent	1	3	3	3	3	3	3	2	2	3	3	2	2	2	2	2	2
11.	Custom security	1	4	4	2	2	4	4	4	4	4	4	3	3	3	3	4	4
12.	Dependence on third-part code	1	5	5	3	3	3	3	3	3	3	3	3	3	3	3	3	3
13.	User training	1	5	5	3	2	2	2	2	2	4	4	3	3	4	4	4	4
	TF		41.5		48.5		40.5		44.5		41.5		43.5		44.5		46.5	
	Nilai TCF = 0.9 + (0.01 * TF)		1.015		1.065		1.005		1.045		1.025		1.035		1.045		1.065	

Hasil Perhitungan Environmental Complexity Factor (ECF)

Sama halnya dengan perhitungan TCF, perhitungan *Environmental Complexity Factor* (ECF) juga membutuhkan nilai dari masing-masing faktor dimana nilai tersebut nantinya akan dikalikan dengan bobot pada masing-masing faktor dan akan menghasilkan nilai ECF.

Berikut yaitu hasil perhitungan nilai ECF dari keseluruhan proyek pengembangan website pemerintahan pada Tugas Akhir ini dapat dilihat pada tabel 6 seperti berikut:

Tabel 6. Nilai ECF Proyek Perangkat Lunak Website Pemerintahan

No	Environmental Factor	Bobot (B)	Kode Proyek Perangkat Lunak															
			A		B		C		D		E		F		G		H	
			N ₁	BxN	N ₁	BxN	N ₁	BxN	N ₁	BxN	N ₁	BxN	N ₁	BxN	N ₁	BxN	N ₁	BxN
1.	Familiarity with the Project	1.5	3	4.5	3	4.5	3	4.5	5	7.5	3	4.5	5	7.5	5	7.5	5	7.5
2.	Application Experience	0.5	3	1.5	4	2	3	1.5	4	2	3	1.5	5	2.5	5	2.5	4	2
3.	OO Programming Experience	1	1	1	4	4	2	2	2	2	2	3	3	3	3	3	2	2
4.	Load Analytic Capability	0.5	4	2	5	2.5	3	1.5	4	2	3	1.5	5	2.5	5	2.5	4	2
5.	Motivation	1	4	4	5	5	3	3	4	4	3	3	3	3	3	3	4	4
6.	Stable Requirements	2	4	8	3	6	2	4	3	6	1	2	4	8	1	2	4	8
7.	Part Time Staff	-1	3	-3	1	-1	1	-1	4	-4	1	-1	2	-2	2	-2	4	-4
8.	Difficult Programming Language	-1	3	-3	1	-1	1	-1	2	-2	1	-1	1	-1	1	-1	3	-3
	EF		17		15		13.5		17.5		11.5		13.5		17.5		18.5	
	Nilai ECF = 1.1 + (0.03 * EF)		1.039		1.065		1.039		1.075		1.065		1.065		1.075		1.046	

Hasil Perhitungan Use Case Point Estimation

Keseluruhan hasil perhitungan nilai UCP proyek pengembangan website pemerintahan pada tugas akhir ini dapat dilihat pada tabel 7 seperti berikut:

Tabel 7. Nilai UCP Proyek Pengembangan Website Pemerintahan

Variabel	Kode Proyek							
	A	B	C	D	E	F	G	H
UUCP	43	27	26	34	31	30	21	31
	5	7	4	7	7	2	4	2
TCF	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0
	15	55	05	45	25	35	45	65
ECF	0,8	0,6	0,9	0,8	1,0	0,6	0,8	0,8
	9	5	95	75	55	95	75	45
UCP	39	18	26	31	34	21	19	28
	2	9	3	7	2	7	5	0

Hasil Analisis Korelasi dan Persamaan Regresi

Berikut nilai *actual effort* beserta nilai UCP dari keseluruhan proyek pengembangan perangkat lunak website pemerintahan pada tugas akhir ini dapat dilihat pada tabel 8 seperti berikut:

Tabel 8. Nilai Actual Effort dan UCP Estimation

Kode Proyek	Actual Effort	UCP Estimation
A	2.860	392
B	2.176	189
C	2.592	263
D	3.180	317
E	2.880	342
F	2.184	217
G	1.848	195
H	2.400	280

Dari nilai *actual effort* dan UCP tersebut kemudian dimasukkan ke dalam SPSS yang kemudian nilai – nilai yang ada tersebut dilakukan proses analisis korelasi sehingga didapatkan korelasi seperti tabel 9 seperti berikut:

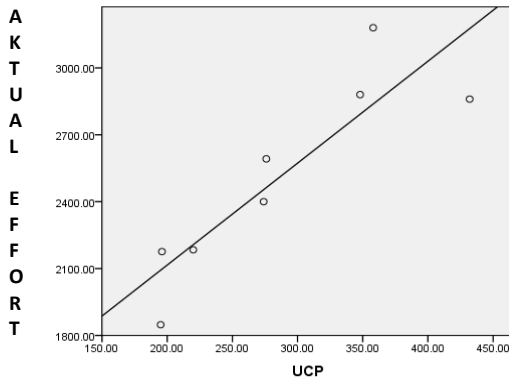
Tabel 9 Korelasi Antara Nilai Actual Effort dan Nilai UCP

		ActualEffort	UCP
ActualEffort	Pearson Correlation	1	.850**
	Sig. (2-tailed)		.008
	N	8	8
UCP	Pearson Correlation	.850**	1
	Sig. (2-tailed)	.008	
	N	8	8

Jika diperoleh nilai Pearson Correlation lebih besar dari 0.5, maka terdapat korelasi

antara kedua variabel tersebut. Berdasarkan interpretasi nilai korelasi pada tabel 2.5 (Usman, 2006). Dari tabel di atas didapatkan nilai Pearson *Correlation* sebesar 0.850, maka dapat disimpulkan nilai *actual effort* dan nilai UCP mempunyai korelasi yang sangat kuat.

Setelah dilakukan analisis korelasi, maka kemudian dicari persamaan regresi menggunakan SPSS. Hasil persamaan regresi dapat dilihat pada gambar 2 seperti berikut :



Gambar 2. Hasil Regresi dan Kurva Nilai *Actual Effort* dengan UCP

Dari hasil regresi pada gambar 2 di atas menunjukkan bahwa terbentuk titik-titik yang membentuk satu garis lurus, sehingga dapat disimpulkan bahwa nilai *actual effort* dan nilai UCP mempunyai persamaan regresi linier. Tahap selanjutnya yaitu dilakukan pencarian persamaan regresi menggunakan *tools* SPSS. Hasil persamaan regresi yang didapatkan ditunjukkan seperti pada tabel 11 berikut :

Tabel 11 Nilai Koefisien Persamaan Regresi

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	1094,360	370,461		2,954	,025
	UCP	5,178	1,310	,850	3,952	,008

Berdasarkan tabel koefisien pada tabel 4.11 dapat dilihat pada kolom B, Constant =1094,360 dan UCP = 5,178. sehingga persamaan regresi adalah seperti berikut: $y = 1094,360 + 5,178x$

Hasil Perhitungan Nilai *Effort Rate* (ER)

Setelah diketahui persamaan regresi, selanjutnya dilakukan perhitungan nilai garis singgung. Langkah pertama yaitu mengambil nilai x_1 dan x_2 dari *scatterplot* pada garis regresi tersebut. Pada perhitungan garis singgung ini

akan dihasilkan nilai (x_1, y_1) dan (x_2, y_2) . Perhitungan garis singgung seperti berikut:

Persamaan regresi : $y = 1094,360 + 5,178x$

Diketahui: $x_1 = 189$

Maka: $y_1 = 1094,360 + 5,178x$
 $= 1094,360 + 5,178 * x_1$
 $= 1094,360 + 5,178 * 189$
 $= 1094,360 + 978,642$
 $= 2073,002$

Diketahui: $x_2 = 392$

Maka: $y_2 = 1094,360 + 5,178x$
 $= 1094,360 + 5,178 * x_2$
 $= 1094,360 + 5,178 * 392$
 $= 1094,360 + 2029,776$
 $= 3124,136$

Setelah mengetahui nilai garis singgung (x_1, y_1) dan (x_2, y_2) maka langkah selanjutnya yaitu mencari nilai ER dengan melakukan perhitungan nilai tangen θ seperti berikut :

Diketahui: $y_1 = 2073,002$

$y_2 = 3124,136$

$x_1 = 189$

$x_2 = 392$

Maka:

$$\text{Tangen } \theta \text{ (ER)} = \frac{y_2 - y_1}{x_2 - x_1}$$

$$\text{Tangen } \theta \text{ (ER)} = \frac{3124,136 - 2073,002}{392 - 189}$$

$$\text{Tangen } \theta \text{ (ER)} = \frac{1051,134}{203}$$

$$\text{Tangen } \theta \text{ (ER)} = 5,178$$

Dari perhitungan di atas telah diketahui nilai *Effort Rate* (ER) proyek pengembangan perangkat lunak pemerintahan sebesar 5,178 *man-hours*.

Diskusi Terhadap Nilai *Effort Rate* (ER) Yang Ditemukan

Diskusi Terhadap ER Penelitian Dengan Nilai ER Karner

Pada penelitian tugas akhir penentuan nilai *effort rate* website pemerintahan ini didapatkan nilai *effort rate* sebesar 5,178 *man-hours*. Nilai ER tersebut jauh lebih kecil dibandingkan dengan nilai ER yang diusulkan oleh Karner yaitu sebesar 20 *man-hours*.

Penelitian yang dilakukan oleh Karner terjadi pada tahun 1993. Teknologi pada rentang waktu 1993 sampai dengan tahun dibuatnya website pemerintahan yang digunakan sebagai bahan penelitian tugas akhir ini telah mengalami peningkatan yang begitu pesat, sehingga apabila nilai ER yang dihasilkan pada penelitian tugas akhir ini mengalami penurunan, maka penyebab yang paling berpotensi mengakibatkan menurunnya nilai ER yaitu faktor teknologi. Beberapa faktor teknologi yang berpotensi menyebabkan penurunan nilai ER pada penelitian tugas akhir ini yaitu sebagai berikut :

a). Teknologi *software engineering*

Semakin berkembangnya teknologi informasi maka semakin berkembang pula *tool* untuk membantu dalam melakukan pengemabangan perangkat lunak website. Jika dahulu *programmer* mengetik baris-baris program dalam *console* atau layar gelap, kini tinggal memilih *software tool* pembuatan pemograman sesuai bahasa pemograman yang dipilih. Selain itu, kini *programmer* dapat mengetik baris program, mendesain tampilan (*interface*), mengkompilasinya, dan kemudian membuat file *executable* (.exe) serta mengkoneksikannya dengan database.

Telah banyak *software tool* yang dapat membantu dalam pengembangan perangkat lunak. Berikut kegiatan-kegiatan dalam pengembangan perangkat lunak yang dapat dimudahkan karena kemajuan teknologi *software engineering*:

➤ Membuat rencana proyek

Terdapat *software* yang digunakan dalam membuat rencana proyek. Salah satu contoh *software* yaitu *Microsoft Project*. Pada *Microsoft Project* telah disediakan fitur-fitur dalam menyusun rencana kegiatan proyek dari awal sampai selesai, beserta peran-peran yang terlibat dalam proyek tersebut.

➤ Analisa kebutuhan

Pada tahap kegiatan analisa kebutuhan, telah terdapat berbagai *software* untuk memudahkan pihak pengembang dalam kegiatannya. Beberapa *software* yang digunakan pada kegiatan analisa kebutuhan yaitu *Ms. Visio*, *Process analyst*, *Enterprise Architecture*, *Rational Rose* dan lain-lain. Pada proses pembuatan analisa

kebutuhan, pihak pengembang hanya perlu melakukan *drag and drop* pada *software* tersebut.

➤ Membuat *database*

Dalam membuat *database*, dewasa ini telah banyak *software* yang digunakan untuk menunjang pengembang dalam membuat *database*. *Software* tersebut antara lain seperti *Power Designer*, *MySql*, *Oracle*, dan lain-lain. Dengan adanya *software* tersebut, tentunya sangat berguna bagi para pengembang dalam membuat *database* sehingga pekerjaan yang mereka lakukan lebih mudah dan lebih cepat.

➤ Kegiatan *coding*

Pembuatan *code* lebih dipermudah dengan adanya bahasa *coding* seperti PHP, ASP. Selain itu telah tersedia *framework* yang digunakan untuk mempermudah pembuatan kode program. *Framework* dapat diartikan sebagai koleksi atau kumpulan potongan-potongan program yang disusun atau diorganisasikan sedemikian rupa, sehingga dapat digunakan untuk membantu membuat aplikasi utuh tanpa harus membuat semua kodenya dari awal. Beberapa contoh *framework* yaitu Code Igtener, Cake PHP, Zend, Symphony, Prado (Farhan, 2011).

b). Website menggunakan komponen

Pada proyek pengembangan perangkat lunak website terdapat berbagai macam fitur dalam menunjang pembuatan website sehingga pembuatan website lebih mudah dan lebih cepat. Penggunaan komponen *software* yang telah ada dapat menguntungkan dari segi siklus waktu pengembang perangkat lunak, karena mampu mengurangi waktu sebesar 70% serta biaya produksi berkurang sampai 84% arena pembangunan komponen berkurang (Proboyekti, 2010).

c). *Source* dari internet

Perkembangan dunia internet juga mempengaruhi jumlah waktu yang digunakan dalam proyek pembuatan website. Akses internet yang mudah dan *source* internet yang lengkap jelas memudahkan para pengembang proyek dalam mencari referensi maupun tutorial dalam pengerjaan website.

Berbagai literatur bisa didapatkan dari internet. Selain itu *source code* juga dapat dicari di internet, hal tersebut dapat memudahkan pengembang perangkat lunak dalam hal melakukan tahapan kegiatan-kegiatan pengembangan perangkat lunak. Serta adanya referensi dan contoh proyek perangkat lunak yang serupa atau memiliki jenis yang sama dengan perangkat lunak yang sedang dikembangkan sehingga dapat dijadikan rujukan dalam pengembangan perangkat lunak dan dapat mengefisiensi waktu pengerjaan.

Dengan beberapa faktor yang telah dijabarkan di atas, dapat disimpulkan bahwa saat ini *effort* yang dibutuhkan pengembang perangkat lunak website untuk menyelesaikan *project* yang mereka lakukan tidak sebesar *effort* yang dibutuhkan oleh Karner pada tahun 1993 yaitu sebesar 20 *man-hours*.

Perbandingan Nilai Effort Rate Peneliti Dengan Nilai Effort Rate Ochodek

Nilai ER yang dihasilkan Karner pada tahun 1993 sebesar 20 *man-hours*. Namun, pada penelitian yang dilakukan oleh Ochodek pada tahun 2011 dengan menggunakan 14 proyek perangkat lunak, didapatkan hasil ER yang bervariasi, antara nilai 3 sampai dengan 35 *man-hours*. Nilai *effort rate* yang dihasilkan pada penelitian Ochodek dapat dilihat pada gambar 3 sebagai berikut:

Post PF (transactions)	21	35	15	7	10	11	12	12	21	18	14	14	17	4
Post PF (steps)	11	33	10	5	5	5	8	7	10	9	8	8	9	3
Default PF [26]	20	20	20	20	20	20	20	20	20	20	20	20	20	20

Gambar 3 Nilai Effort Rate Penelitian Ochodek

Dengan membandingkan antara nilai ER yang dihasilkan pada penelitian tugas akhir ini yang bernilai sebesar 5,178 *man-hours* dengan nilai ER yang dihasilkan pada penelitian yang dilakukan Ochodek, maka dapat diketahui bahwa pernah dihasilkan nilai ER yang lebih kecil dibanding nilai ER yang ditemukan oleh peneliti pada tugas akhir ini. Sehingga nilai ER yang dihasilkan merupakan nilai ER yang tergolong wajar.

Pengaruh Effort Rate Dengan Estimasi Biaya Proyek Pengembangan Perangkat Lunak

Nilai *Effort Rate* (ER) berbanding lurus dengan nilai estimasi *effort*, sehingga dengan nilai ER yang semakin kecil akan mengakibatkan nilai estimasi *effort* yang semakin kecil pula. Hasil dari perhitungan *effort* menggunakan metode UCP berupa nilai *hours of effort*. Dengan diketahuinya nilai dari estimasi *effort* tersebut, maka dapat dilanjutkan untuk perhitungan selanjutnya, yaitu perhitungan biaya proyek pengembangan perangkat lunak. Terdapat 3 kelompok aktivitas pada pembuatan perangkat lunak. Masing – masing kelompok tersebut mempunyai prosentase nilai *effort* yang berbeda (Shaleh, 2011). Untuk lebih jelasnya, prosentase nilai *effort* tersebut dapat dilihat pada tabel 12 seperti berikut :

Tabel 12. Pembagian Kelompok Aktivitas Pembuatan Proyek

No.	Kelompok Aktivitas	%Effort
1	Software Development	
A	Requirement	7,5%
B	Specifications & Design	17,5%
C	Coding	10,0%
D	Integration Testing	7,0%
	Total	42,0%
2	On Going Activity	
A	Project Management	7,0%
B	Configuration Management	4,0%
C	Documentation	4,0%
D	Acceptance & Deployment	6,0%
	Total	21,0%
3	Quality and Testing	
A	Quality Assurance & Control	12,5%
B	Evaluation and Testing	24,5%
	Total	37,0%

Sumber : Shaleh (2011)

Dari tabel di atas kemudian dapat dihitung berapa nilai *effort* dari masing – masing aktivitas. Apabila sudah diketahui nilai *effort*

dari masing – masing aktivitas, maka selanjutnya bisa dilakukan perhitungan estimasi biaya. Untuk menghitung estimasi biaya diperlukan acuan dalam menentukan standar gaji pada tiap – tiap peran berdasarkan aktivitas. Standar gaji yang sering dipakai mengacu pada Indonesia *Salary Guide* 2011/2012 yang diterbitkan oleh Kelly Service, Inc. Standar gaji proyek pembuatan perangkat lunak dapat dilihat pada tabel 4.13 sebagai berikut :

Tabel 4.13 Penetapan Standar Gaji Proyek Perangkat Lunak

Segmentasi Peran	Posisi dalam Salary Guide	Standar Gaji (per-bln) (Rp)	Standar Gaji (per-jam) (Rp)
Requirement	Systems Analyst	5.000.000	31.250
Specifications & Design	Systems Analyst	5.000.000	31.250
Coding	Software Engineer	3.000.000	18.750
Integration Testing	Software Engineer	3.000.000	18.750
Project Management	Project Manager	10.000.000	62.500
Configuration Management	Solutions Architect Degree	7.000.000	43.750
Documentation	Software Engineer	3.000.000	18.750
Acceptance & Deployment	IT Auditor	4.000.000	25.000
Quality Assurance & Control	Software QA	3.000.000	18.750
Evaluation and Testing	Test Analyst	3.000.000	18.750

Sumber : Kelly Services (2011)

Dari tabel di atas dapat diketahui standar gaji per-jam dari masing-masing peran. Setelah diketahui gaji per-jam maka selanjutnya dapat dilakukan perhitungan biaya pembuatan proyek per-jam. Apabila telah diketahui nilai pembuatan proyek per-jam, kemudian dapat dilakukan perhitungan total biaya proyek dengan cara melakukan perkalian antara *hours of effort* dengan biaya per-jam. Hasil perkalian tersebut merupakan nilai dari estimasi biaya proyek pembuatan perangkat lunak.

Tujuan dari perhitungan nilai *effort rate* yaitu untuk mendapatkan nilai estimasi *effort* yang akurat. Dengan nilai *effort* yang akurat, maka selanjutnya akan dilanjutkan untuk perhitungan estimasi biaya. Dengan nilai estimasi *effort* yang akurat, maka akan didapatkan nilai estimasi biaya yang akurat pula.

KESIMPULAN

Kesimpulan yang dapat diambil dari pengerjaan tugas akhir ini yakni sebagai berikut:

1. Penelitian tugas akhir ini mempunyai keterbatasan sumber data tidak berasal dari dokumen SKPL dari pihak pengembang, maka untuk penelitian mendatang, diharapkan data yang digunakan untuk penelitian penentuan nilai *effort rate* berupa dokumen SKPL dari pihak pengembang perangkat lunak.
2. Korelasi antara nilai *actual effort* dengan nilai UCP pada penelitian tugas akhir ini yaitu sebesar 0.850, maka dapat disimpulkan bahwa nilai *actual effort* dengan nilai UCP mempunyai korelasi yang sangat kuat.
3. Dengan diketahuinya nilai *actual effort* dan nilai UCP masing-masing proyek pengembangan perangkat lunak website pemerintahan, maka didapatkan persamaan linier yaitu: $y = 1094,360 + 5,178x$, sehingga didapatkan nilai empiris *Effort Rate* (ER) yaitu sebesar 5,178 *man-hours*. Nilai ER tersebut lebih kecil dibandingkan oleh nilai ER yang dikemukakan oleh Karner sebesar 20 *man-hours*. Hal ini dimungkinkan karena beberapa faktor berikut :
 - a. Teknologi *software engineering*
 - b. Website menggunakan komponen
 - c. *Source* dari internet

SARAN

Beberapa hal yang diharapkan dapat dikembangkan di masa mendatang adalah sebagai berikut:

1. Perlu dilakukan analisis lebih mendalam untuk menentukan alasan penyebab nilai *Effort Rate* (ER) menjadi lebih kecil dibandingkan dengan nilai ER yang diusulkan Karner sebesar 20 *man-hours*.
2. Perlu dilakukan penelitian kembali untuk menentukan prosentase masing-masing aktivitas proyek pengembangan

perangkat lunak website seperti yang dikemukakan oleh Kassem Shaleh pada tahun 2011.

RUJUKAN

- Bull Survey. 1998. *Failure Causes* <http://www.itcortex.com/Stat_Failure_Cause.htm#surveys> Diakses pada tanggal 20 Februari 2013.
- Carroll, Edward R. 2005. *Estimating Software Based on Use Case Points*. Object-Oriented, Programming, Systems, Languages, and Object Oriented Programming Systems Languages and Applications (OOPSLA) Conference, San Diego, CA, pp.257–265.
- Clemmons, Roy K. 2006. *Project Estimation With Use Case Point*. Diversified Technical Services, Inc.
- Cohn, Mike. 2005. *Estimating With Use Case Point*. <<http://www.mountaingoatsoftware.com/articles/estimating-with-use-case-points>> diakses pada tanggal 14 April 2013.
- Damodaran, Mel. 2002. *Estimation Using Use Case Points*. Computer Science Program, University of Houston-Victoria.
- Farhan, Muhamad. 2011. *Pemahaman Framework*. <<http://agiptek.com/index.php/php/98-sedikit-pemahaman-tentang-apa-itu-framework.html>> diakses pada tanggal 18 Juni 2013.
- Frohnhoff, S. 2008. *Revised Use Case Point Method-Effort Estimation in Development Projects for Business Applications*. Offenbach.
- Galorath, D. 2006. *Software Sizing, Estimation and Risk Management*. Auerbach.
- Karner, Gustav. 1993. *Resource Estimation for Objectory Projects*. Objective System SF AB.
- Kelly Service, Inc. 2011. *Indonesia Salary Guide 2011/2012*.
- KPMG Canada. 1997. *Failure Causes* <http://www.itcortex.com/Stat_Failure_Cause.htm#surveys> diakses pada tanggal 20 Februari 2013.
- Monteiro, C., Silva, D., dan Marques, A. 2008. *Applying The Use Case Points Effort Estimation Technique To Avionics Systems*, IEEE.
- Muhardin, Endy. 2011. *Estimasi Proyek Software*. <<http://software.endy.muhardin.com/manajemen/estimasi-proyek-software/>> diakses pada 7 Maret 2013.
- Nageswaran, Suresh. 2001. *Test Effort Estimation Using Use Case Points*. <www.cognizant.com/cogcommunity/presentations/Test_Effort_Estimation.pdf>.
- Ochodek, M. *Simplifying Effort Estimation Based on Use Case Points*, Sciencedirect 2011.
- Rizky, Soetam. 2008. *Konsep Dasar Rekayasa Perangkat Lunak Software Reengineering*. Prestasi Pustaka.
- Saleh, K. 2011. *Effort and Cost Allocation in Medium to Large Software Development Projects*. *Intenational Journal of Computers* (1), 74-79.
- Schneider, G. and Winters, J. 1998. *Applying Use Cases – A Practical Guide*. Addison-Wesley.
- Standish Group, 2011. *Chaos Reports*. <www.standishgroup.com/news> diakses pada tanggal 3 September 2012.
- Tokey, Steve. 2004. *Return of Software: Maximizing the Return on Your Software Investment*, Prentice Hall.
- Umi Proboyekti, 2010. *Software Process Model*. <<http://lecturer.ukdw.ac.id/othie/softwareprocess.pdf>> diakses pada tanggal 18 Juni 2013.
- Usman, Husaini. 2006. *Pengantar Statistika*. Bumi Aksara.
- Wigrantoro, Mas. 2003. *Pemanfaatan Teknologi Informasi Dalam Penerapan Good Governance di Indonesia*. Seminar Paradigma Good Governance di Era Informasi yang Kompetitif, Demokratis dan Transparan, Universitas Gunadarma.