

REpubLIK INDONESIA
KEMENTERIAN HUKUM DAN HAK ASASI MANUSIA

SURAT PENCATATAN CIPTAAN

Dalam rangka perlindungan ciptaan di bidang ilmu pengetahuan, seni dan sastra berdasarkan Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta, dengan ini menerangkan:

Nomor dan tanggal permohonan : EC00201852556, 3 November 2018

Pencipta

Nama : **Herman Yuliansyah dan Hanifahma Wardhani**
Alamat : Perumahan Graha Nirmala, No. F-17, Jeblog, Tirtonirmolo, Kasihan, Bantul, DI Yogyakarta, Bantul, DI YOGYAKARTA, 55181
Kewarganegaraan : Indonesia

Pemegang Hak Cipta

Nama : **Herman Yuliansyah dan Hanifahma Wardhani**
Alamat : Perumahan Graha Nirmala, No. F-17, Jeblog, Tirtonirmolo, Kasihan, Bantul, DI Yogyakarta, Bantul, DI YOGYAKARTA, 55181
Kewarganegaraan : Indonesia

Jenis Ciptaan : **Program Komputer**
Judul Ciptaan : **Simulasi Penentuan Nilai Effort Rate (ER) Pada Use Case Point (UCP)**

Tanggal dan tempat diumumkan untuk pertama kali : 28 Oktober 2018, di Yogyakarta
di wilayah Indonesia atau di luar wilayah Indonesia

Jangka waktu perlindungan : Berlaku selama 50 (lima puluh) tahun sejak Ciptaan tersebut pertama kali dilakukan Pengumuman.

Nomor pencatatan : 000122921

adalah benar berdasarkan keterangan yang diberikan oleh Pemohon.

Surat Pencatatan Hak Cipta atau produk Hak terkait ini sesuai dengan Pasal 72 Undang-Undang Nomor 28 Tahun 2014 tentang Hak Cipta.

a.n. MENTERI HUKUM DAN HAK ASASI MANUSIA
DIREKTUR JENDERAL KEKAYAAN INTELEKTUAL



Dr. Freddy Harris, S.H., LL.M., ACCS.
NIP. 196611181994031001

Disclaimer:

Dalam hal pemohon memberikan keterangan tidak sesuai dengan surat pernyataan, Menteri berwenang untuk mencabut surat pencatatan permohonan.

USULAN HAK CIPTA
JENIS CIPTAAN PROGRAM KOMPUTER

**Simulasi Penentuan Nilai *Effort Rate* (ER)
Pada *Use Case Point* (UCP)**

Diusulkan Oleh:

Herman Yuliansyah, S.T., M.Eng.

Hanifahma Wardhani, S.Kom.

Diusulkan Kepada:

DIREKTORAT JENDERAL KEKAYAAN INTELEKTUAL

KEMENTERIAN HUKUM DAN HAK ASASI

REPUBLIK INDONESIA

2018

Deskripsi Singkat Sistem

Aplikasi web ini adalah penerapan metode *linear regression* sebagai metode untuk penentuan tingkat upaya proyek pada perhitungan *Use Case Point*. Hasil luaran dari aplikasi ini adalah nilai tingkat upaya proyek (*effort rate*) pengembangan perangkat lunak. Nilai tingkat upaya ini digunakan sebagai acuan pada estimasi pengembangan perangkat lunak dengan perhitungan estimasi perangkat lunak menggunakan metode *use case point*. Pengguna aplikasi akan mengentrikan nama proyek, upaya aktual, biaya aktual, bobot aktor, bobot *use case*, faktor kompleksitas teknis dan faktor lingkungan. Aplikasi ini dapat digunakan sebagai sarana pembelajaran estimasi pengembangan perangkat lunak.

Aplikasi ini dapat diakses mulai tanggal 28 Oktober 2018 di alamat <http://effortrate.bootcampkoding.com/>

Aplikasi web ini dikembangkan dengan bahasa pemrograman PHP, database MySQL dan framework codeigniter serta bahasa pemrograman lainnya yang tidak menjadi bagian dari hak cipta yang diusulkan.

Pengusul Hak Cipta:

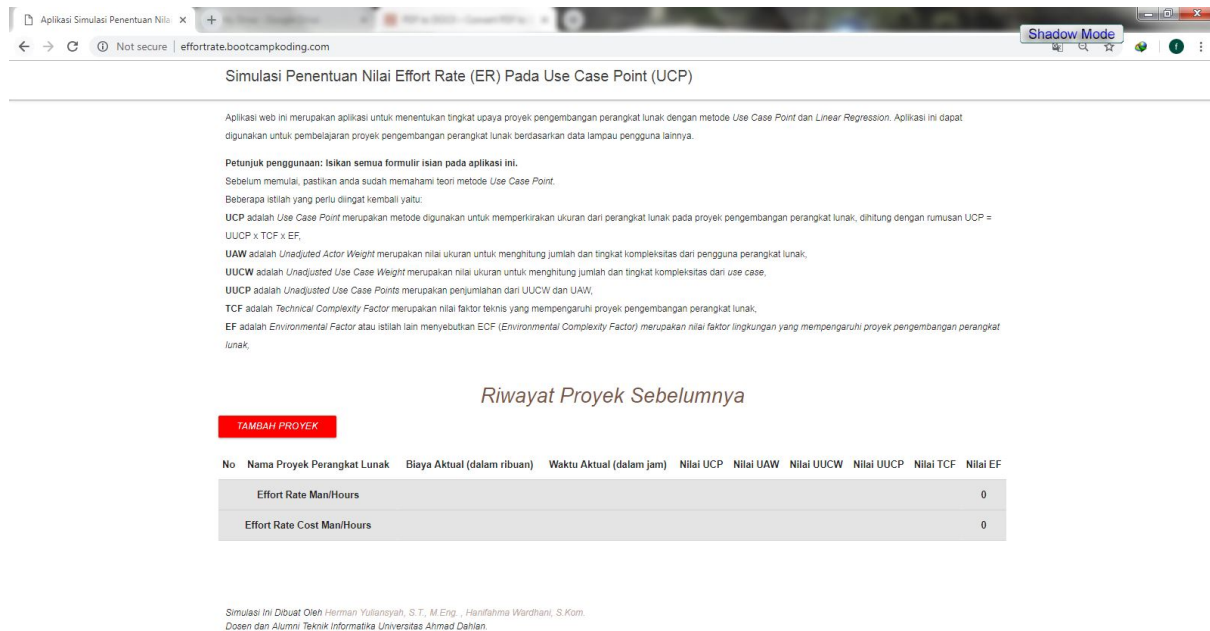
Herman Yuliansyah, S.T., M.Eng. dan Hanifahma Wardhani, S.Kom.

Petunjuk Penggunaan “Simulasi Penentuan Nilai *Effort Rate* (ER) Pada *Use Case Point* (UCP)”

Pada petunjuk ini melingkupi beberapa tahan penggunaan aplikasi yaitu:

1. Mengakses halaman web

Untuk memulai menggunakan aplikasi maka pengguna harus mengakses halaman web yaitu <http://effortrate.bootcampkoding.com/>.



Simulasi Penentuan Nilai *Effort Rate* (ER) Pada *Use Case Point* (UCP)

Aplikasi web ini merupakan aplikasi untuk menentukan tingkat upaya proyek pengembangan perangkat lunak dengan metode *Use Case Point* dan *Linear Regression*. Aplikasi ini dapat digunakan untuk pembelajaran proyek pengembangan perangkat lunak berdasarkan data lampau pengguna lainnya.

Petunjuk penggunaan: Isikan semua formulir isian pada aplikasi ini.

Sebelum memulai, pastikan anda sudah memahami teori metode *Use Case Point*.

Beberapa istilah yang perlu diingat kembali yaitu:

UCP adalah *Use Case Point* merupakan metode digunakan untuk memperkirakan ukuran dari perangkat lunak pada proyek pengembangan perangkat lunak, dihitung dengan rumusan $UCP = UUCP \times TCF \times EF$.

UAW adalah *Unadjusted Actor Weight* merupakan nilai ukuran untuk menghitung jumlah dan tingkat kompleksitas dari pengguna perangkat lunak.

UUCW adalah *Unadjusted Use Case Weight* merupakan nilai ukuran untuk menghitung jumlah dan tingkat kompleksitas dari use case.

UUCP adalah *Unadjusted Use Case Points* merupakan penjumlahan dari UUCW dan UAW.

TCF adalah *Technical Complexity Factor* merupakan nilai faktor teknis yang mempengaruhi proyek pengembangan perangkat lunak.

EF adalah *Environmental Factor* atau istilah lain menyebutkan ECF (*Environmental Complexity Factor*) merupakan nilai faktor lingkungan yang mempengaruhi proyek pengembangan perangkat lunak.

Riwayat Proyek Sebelumnya

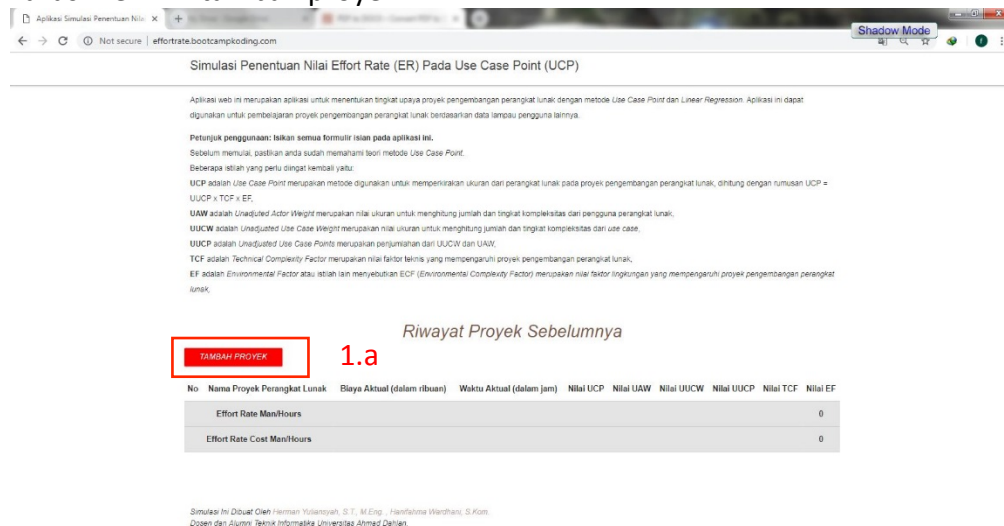
TAMBAH PROYEK

No	Nama Proyek Perangkat Lunak	Biaya Aktual (dalam ribuan)	Waktu Aktual (dalam jam)	Nilai UCP	Nilai UAW	Nilai UUCW	Nilai UUCP	Nilai TCF	Nilai EF
	Effort Rate Man/Hours								0
	Effort Rate Cost Man/Hours								0

Simulasi ini Dibuat Oleh Herman Yuliansyah, S.T., M.Eng., Hanifahma Wardhani, S.Kom.
Dosen dan Alumni Teknik Informatika Universitas Ahmad Dahlan.

2. Menambah Proyek

Setelah pengaksesan untuk memulai menggunakan aplikasi maka pengguna harus memilih tambah proyek



Simulasi Penentuan Nilai *Effort Rate* (ER) Pada *Use Case Point* (UCP)

Aplikasi web ini merupakan aplikasi untuk menentukan tingkat upaya proyek pengembangan perangkat lunak dengan metode *Use Case Point* dan *Linear Regression*. Aplikasi ini dapat digunakan untuk pembelajaran proyek pengembangan perangkat lunak berdasarkan data lampau pengguna lainnya.

Petunjuk penggunaan: Isikan semua formulir isian pada aplikasi ini.

Sebelum memulai, pastikan anda sudah memahami teori metode *Use Case Point*.

Beberapa istilah yang perlu diingat kembali yaitu:

UCP adalah *Use Case Point* merupakan metode digunakan untuk memperkirakan ukuran dari perangkat lunak pada proyek pengembangan perangkat lunak, dihitung dengan rumusan $UCP = UUCP \times TCF \times EF$.

UAW adalah *Unadjusted Actor Weight* merupakan nilai ukuran untuk menghitung jumlah dan tingkat kompleksitas dari pengguna perangkat lunak.

UUCW adalah *Unadjusted Use Case Weight* merupakan nilai ukuran untuk menghitung jumlah dan tingkat kompleksitas dari use case.

UUCP adalah *Unadjusted Use Case Points* merupakan penjumlahan dari UUCW dan UAW.

TCF adalah *Technical Complexity Factor* merupakan nilai faktor teknis yang mempengaruhi proyek pengembangan perangkat lunak.

EF adalah *Environmental Factor* atau istilah lain menyebutkan ECF (*Environmental Complexity Factor*) merupakan nilai faktor lingkungan yang mempengaruhi proyek pengembangan perangkat lunak.

Riwayat Proyek Sebelumnya

TAMBAH PROYEK 1.a

No	Nama Proyek Perangkat Lunak	Biaya Aktual (dalam ribuan)	Waktu Aktual (dalam jam)	Nilai UCP	Nilai UAW	Nilai UUCW	Nilai UUCP	Nilai TCF	Nilai EF
	Effort Rate Man/Hours								0
	Effort Rate Cost Man/Hours								0

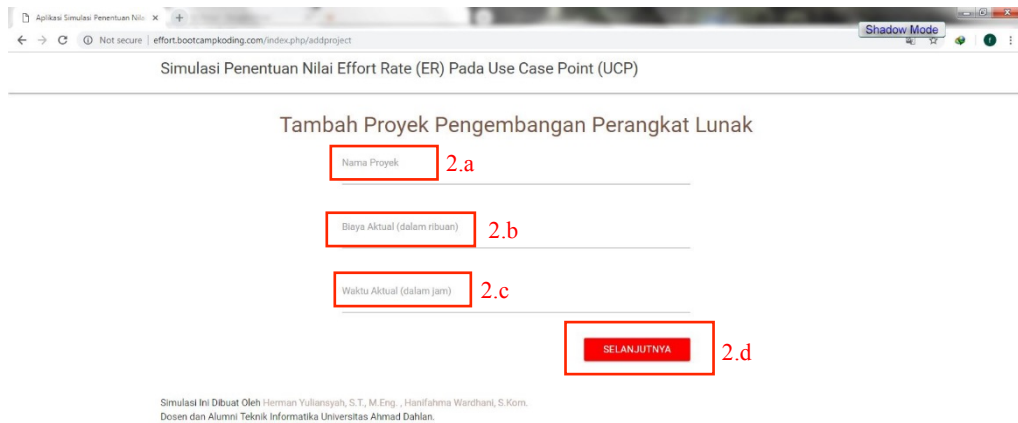
Simulasi ini Dibuat Oleh Herman Yuliansyah, S.T., M.Eng., Hanifahma Wardhani, S.Kom.
Dosen dan Alumni Teknik Informatika Universitas Ahmad Dahlan.

Petunjuk:

1.a : Untuk menambah proyek pilih tombol tambah proyek.

2. Mengisikan identitas proyek

Pengguna dapat mengisikan identitas proyek berupa nama proyek, real pendapatan dan real waktu.



The screenshot shows a web browser window with the URL `effort.bootcampkoding.com/index.php/addproject`. The page title is "Simulasi Penentuan Nilai Effort Rate (ER) Pada Use Case Point (UCP)". The main heading is "Tambah Proyek Pengembangan Perangkat Lunak". There are four input fields: "Nama Proyek" (labeled 2.a), "Biaya Aktual (dalam ribuan)" (labeled 2.b), "Waktu Aktual (dalam jam)" (labeled 2.c), and a red "SELANJUTNYA" button (labeled 2.d). At the bottom, there is a small text block: "Simulasi Ini Dibuat Oleh Herman Yulliansyah, S.T., M.Eng., Hanifahma Wardhani, S.Kom. Dosen dan Alumni Teknik Informatika Universitas Ahmad Dahlan."

Petunjuk:

2.a : Pengguna mengisikan nama proyek

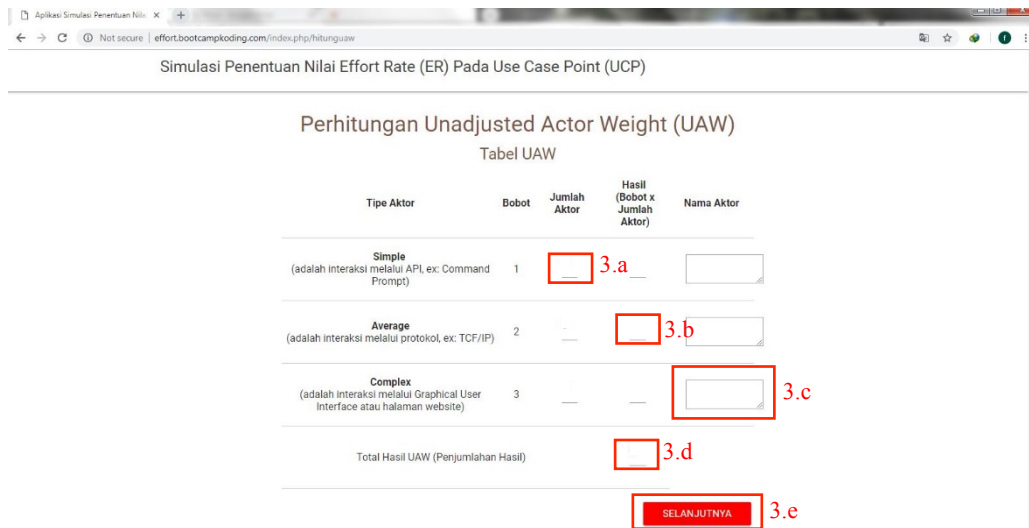
2.b : Pengguna mengisikan real pendapatan

2.c : Pengguna mengisikan real waktu

2.d : Untuk melanjutkan mengisi, pengguna mengklik tombol SELANJUTNYA.

3. Mengisi bobot aktor

Setelah proses mengisi data proyek berhasil dilakukan maka tahap selanjutnya adalah proses mengisi bobot aktor.



The screenshot shows a web browser window with the URL `effort.bootcampkoding.com/index.php/hitunguaw`. The page title is "Simulasi Penentuan Nilai Effort Rate (ER) Pada Use Case Point (UCP)". The main heading is "Perhitungan Unadjusted Actor Weight (UAW)". Below the heading is a table with the caption "Tabel UAW". The table has five columns: "Tipe Aktor", "Bobot", "Jumlah Aktor", "Hasil (Bobot x Jumlah Aktor)", and "Nama Aktor". There are four rows: "Simple" (Bobot 1, Jumlah Aktor input field labeled 3.a), "Average" (Bobot 2, Jumlah Aktor input field labeled 3.b), "Complex" (Bobot 3, Jumlah Aktor input field labeled 3.c), and "Total Hasil UAW (Penjumlahan Hasil)" (Bobot input field labeled 3.d). A red "SELANJUTNYA" button (labeled 3.e) is at the bottom right.

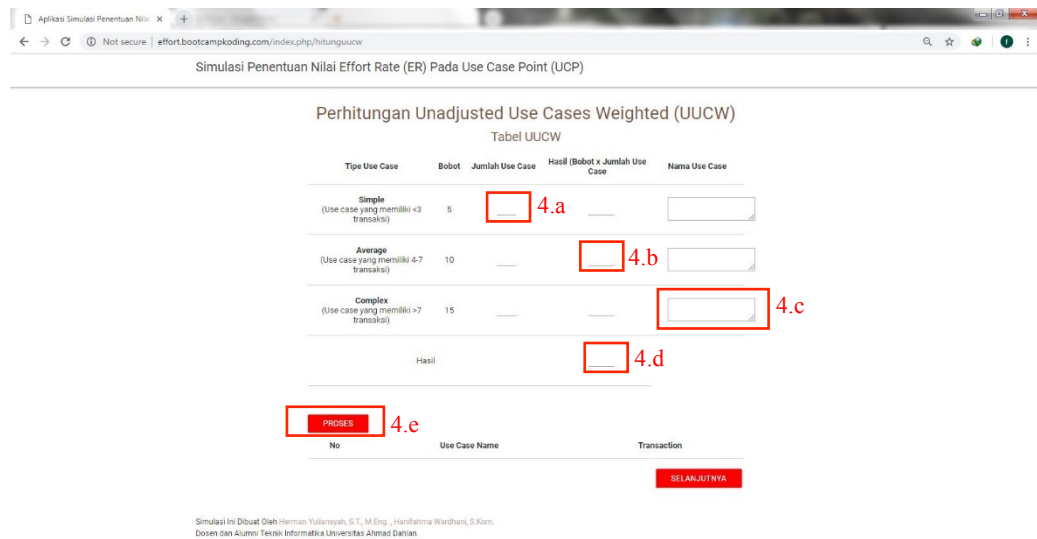
Tipe Aktor	Bobot	Jumlah Aktor	Hasil (Bobot x Jumlah Aktor)	Nama Aktor
Simple (adalah interaksi melalui API, ex: Command Prompt)	1	<input type="text"/>	<input type="text"/>	<input type="text"/>
Average (adalah interaksi melalui protokol, ex: TCF/IP)	2	<input type="text"/>	<input type="text"/>	<input type="text"/>
Complex (adalah interaksi melalui Graphical User Interface atau halaman website)	3	<input type="text"/>	<input type="text"/>	<input type="text"/>
Total Hasil UAW (Penjumlahan Hasil)	<input type="text"/>			

Petunjuk:

- 3.a : Untuk mengisi banyaknya aktor yang berbobot simple, average dan complex.
- 3.b : Hasil dari perkalian antara bobot dan nilai.
- 3.c : Pengisian nama aktor.
- 3.d : Total hasil UAW
- 3.e : Untuk melanjutkan mengisi, pengguna mengklik tombol SELANJUTNYA

4. Mengisi bobot *use case*

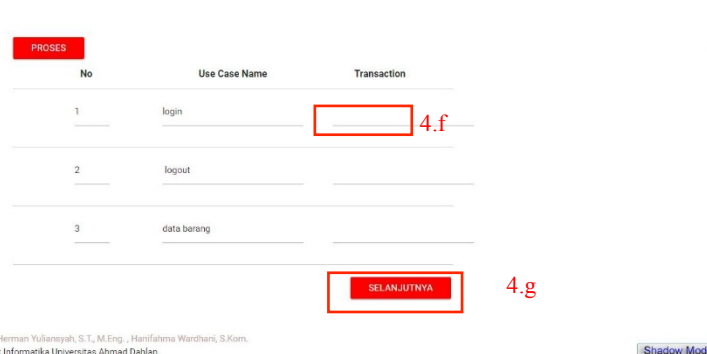
Setelah proses mengisi bobot aktor selesai maka langkah selanjutnya adalah mengisi bobot *use case*.



Petunjuk:

- 4.a : Untuk mengisi banyaknya *use case* yang berbobot simple, average dan complex.
- 4.b : Hasil dari perkalian antara bobot *use case* dan nilai
- 4.c : Pengisian nama *use case*.
- 4.d : Total penjumlahan dari perkalian antar bobot dan jumlah
- 4.e : Untuk mengisi transaksi yang terjadi pada masing – masing *use case*.

Selanjutnya ketika tombol PROSES diklik, maka akan muncul kolom transaksi yang akan diisi.



Petunjuk:

4.e : Mengisi transaksi yang dilakukan tiap *use case*

4.f : Untuk melanjutkan mengisi, pengguna mengklik tombol SELANJUTNYA

5. Mengisi *Technical Complexity Factor*

Setelah proses mengisi bobot *use case* selesai maka langkah selanjutnya adalah mengisi bobot *Technical Complexity Factor*. Terdapat 13 pernyataan yang ada dalam TCF dan bobot TCF diisi antara skala 0 hingga 5, sesuai dengan pernyataan yang ada.

Simulasi Penentuan Nilai Effort Rate (ER) Pada Use Case Point (UCP)

Technical Complexity Factor

Tabel TCF (isi dengan range 0-5)

No	Technical Factor	Bobot	Nilai	Hasil	Reason
T1	Distributed System	2	Choose your option	5.a	
T2	Application performance objectives, in either response or throughput	1	Choose your option	5.b	
T3	End-user efficiency (on-line)	1	Choose your option		5.c
T4	Complex internal processing	1	Choose your option		
T5	Reusability, the code must be able to reuse in other applications	1	Choose your option		
T6	Easy to install	0.5	Choose your option		
T7	Operational ease, usability	0.5	Choose your option		
T8	Portability	2	Choose your option		
T9	Changeability	1	Choose your option		
T10	Concurrency	1	Choose your option		
T11	Special security features	1	Choose your option		
T12	Provide direct access for third parties	1	Choose your option		
T13	Special user training facilities	1	Choose your option		
Hasil				5.d	
Rumus (0.6 + (hasil/100))				5.e	
				5.f	

Simulasi ini Dibuat Oleh Herman Yuliansyah, S.T., M.Eng., Hafifahma Wardhani, S.Kom.
Dosen dan Alumni Teknik Informatika Universitas Ahmad Dahlan.

Shadow Mode

Petunjuk:

5.a : Untuk mengisikan nilai TCF dengan skala antara 0 – 5.

5.b : Hasil dari perkalian antara bobot TCF dan nilai yang diisikan

5.c : Pengisian alasan memberikan nilai bobot pada pernyataan yang ada.

5.d : Hasil dari perhitungan TCF.

5.e : Hasil akhir nilai TCF

5.f : Untuk melanjutkan ke proses berikutnya, pengguna mengklik tombol NEXT.

6. Mengisi *Environmental Factor*

Setelah proses mengisi bobot *Technical Complexity Factor* selesai maka langkah selanjutnya adalah mengisi bobot *Environmental Factor*. Terdapat 8 pernyataan yang ada dalam EF berhubungan dengan lingkungan kerja tim dan bobot EF diisi antara skala 0 hingga 5, sesuai dengan pernyataan yang ada.

No	Environmental Factor	Bobot	Nilai	Hasil	Reason
E1	Familiar with Objectory	1.5	Choose your option		
E2	Part time workers	0.5	Choose your option		
E3	Analyst capability	1	Choose your option		
E4	Application experience	0.5	Choose your option		
E5	Object oriented experience	1	Choose your option		
E6	Motivation	2	Choose your option		
E7	Difficult programming language	-1	Choose your option		
E8	Stable requirements	-1	Choose your option		
Hasil					
Rumus (1.4 - (0.03 * hasil))					

Simulasi Ini Dibuat Oleh Herman Yuliansyah, S.T., M.Eng., Hanifahma Wardhani, S.Kom.
Dosen dan Alumni Teknik Informatika Universitas Ahmad Dahlan.

Petunjuk:

6.a : Untuk mengisi nilai EF dengan skala antara 0 – 5.

6.b : Hasil dari perkalian antara bobot EF dan nilai yang diisikan

6.c : Pengisian alasan memberikan nilai bobot pada pernyataan yang ada.

6.d : Total hasil dari perkalian bobot EF dengan nilai yang diisikan.

6.e : Hasil akhir nilai EF

6.f : Untuk melanjutkan ke proses berikutnya, pengguna mengklik tombol NEXT.

7. Hasil Perhitungan proyek

Setelah proses mengisi bobot *environmental factor* selesai maka halaman selanjutnya adalah perhitungan satu proyek yang telah diisikan meliputi total TCF, EF, bobot aktor, bobot *use case*, dan perhitungan *use case point*.

Calculations From Other Tabs		
Technical Complexity Factor	TCF	1.3
Environmental Factor	EF	0.725
Unadjusted Actor Weight	UAW	6
Unadjusted Use Cases Weighted	UUCW	70
Unadjusted Use Case Points	UUCP= (UAW + UUCW)	76
Calculation of Use Case Points		
Use Case Points	UCP = UUCP * TCF * EF	71.63

Simulasi Ini Dibuat Oleh Herman Yuliansyah, S.T., M.Eng., Hanifahma Wardhani, S.Kom.
Dosen dan Alumni Teknik Informatika Universitas Ahmad Dahlan.

Petunjuk:

7.a : Untuk melanjutkan ke proses berikutnya, pengguna mengklik tombol NEXT.

8. Hasil akhir yang menunjukkan besarnya *Effort Rate* yang dihasilkan dari seluruh proyek yang telah diisikan.

Simulasi Penentuan Nilai Effort Rate (ER) Pada Use Case Point (UCP)

History Rekap

No	Project	Real Cost (x 1000)	Real Waktu	UCP	UAW	UUCW	UUCP	TCF	EF
1	proyek 1	Rp 2,00	456.25	71.63	6	70	76	1.3	0.725
ER Man/Hours									0
ER Cost Man/Hours									0.005

TAMBAH PROYEK

8.a

8.b

Simulasi Ini Dibuat Oleh Herman Yuliansyah, S.T., M.Eng., Hanifahma Wardhani, S.Kom.
Dosen dan Alumni Teknik Informatika Universitas Ahmad Dahlan.

Petunjuk:

8.a : Hasil akhir *Effort Rate* dari semua proyek yang telah diisikan

8.b : Untuk kembali mengisikan proyek baru.

Kode Program

Kode Program Untuk Nama File Template.php

```
<?php
defined('BASEPATH') or exit('No direct script access allowed');

class Templete extends CI_Controller{

    public function __construct(){
        parent::__construct();
        $this->load->library(array('parser'));
    }

    public function index(){
        $data['all'] = $this->Modelku-> baca_semua();
        $data['content']='beranda';
        $this->parser->parse('master',$data);
    }

    public function addproject(){
        $data['content']='addproject';
        $this->parser->parse('master',$data);
    }

    public function hitunguaw(){
        $nama_proyek = $this->input->post("nama_proyek");
        $real_penghasilan = $this->input->post("real_pendapatan");
        $real_waktu = $this->input->post("real_waktu");
        $this->Modelku->tambah_proyek($nama_proyek,$real_penghasilan,$real_waktu);
        $data['proyek'] = $this->Modelku->baca_data('proyek');
        $data['content']='hitunguaw';
        $this->parser->parse('master',$data);
    }

    public function hitunguucw(){
        $hasil_simpel = $this->input->post("hasil_simpel");
        $hasil_medium = $this->input->post("hasil_medium");
        $hasil_complex = $this->input->post("hasil_complex");
        $id_proyek = $this->input->post("id_proyek");
        $hasil_uaw = $this->input->post("hasil_uaw");
        $this->Modelku->hitung_uaw($id_proyek,
$hasil_simpel,$hasil_medium,$hasil_complex,$hasil_uaw);
        $data['proyek'] = $this->Modelku->baca_data('proyek');
        $data['content']='hitunguucw';
        $this->parser->parse('master',$data);
    }

    public function hitungtcf(){
        $hasil = $this->input->post("hasil");
        $nameuc = $this->input->post("nameuc");
        $transaction = $this->input->post("transaction");
        $id_proyek = $this->input->post("id_proyek");
        $this->Modelku->hitung_uucw($id_proyek, $nameuc,$transaction,$hasil);
        $data['proyek'] = $this->Modelku->baca_data('proyek');
        $data['content']='hitungtcf';
        $this->parser->parse('master',$data);
    }
}
```

```

public function hitungef(){
    $id_proyek = $this->input->post("id_proyek");
    $t1 = $this->input->post("t1");
    $t2 = $this->input->post("t2");
    $t3 = $this->input->post("t3");
    $t4 = $this->input->post("t4");
    $t5 = $this->input->post("t5");
    $t6 = $this->input->post("t6");
    $t7 = $this->input->post("t7");
    $t8 = $this->input->post("t8");
    $t9 = $this->input->post("t9");
    $t10 = $this->input->post("t10");
    $t11 = $this->input->post("t11");
    $t12 = $this->input->post("t12");
    $t13 = $this->input->post("t13");
    $hasil_tcf = $this->input->post("hasil_tcf");
    $rumus = $this->input->post("rms");
    $this->Modelku->hitung_tcf($id_proyek,
$t1,$t2,$t3,$t4,$t5,$t6,$t7,$t8,$t9,$t10,$t11,$t12,$t13,$hasil_tcf,$rumus);
    $data['proyek'] = $this->Modelku->baca_data('proyek');
    $data['content']='hitungef';
    $this->parser->parse('master',$data);
}

public function hasil(){
    $id_proyek = $this->input->post("id_proyek");
    $e1 = $this->input->post("e1");
    $e2 = $this->input->post("e2");
    $e3 = $this->input->post("e3");
    $e4 = $this->input->post("e4");
    $e5 = $this->input->post("e5");
    $e6 = $this->input->post("e6");
    $e7 = $this->input->post("e7");
    $e8 = $this->input->post("e8");
    $hasil_ef = $this->input->post("hasil_ef");
    $rumus_ef = $this->input->post("rms_ef");
    $ef= $this->Modelku-
>hitung_ef($id_proyek,$e1,$e2,$e3,$e4,$e5,$e6,$e7,$e8,$hasil_ef,$rumus_ef);
    $data['ef']= $this->Modelku->baca_ef($id_proyek);
    $data['tcf']= $this->Modelku->baca_tcf($id_proyek);
    $data['uaw']= $this->Modelku->baca_uaw($id_proyek);
    $data['uucw']= $this->Modelku->baca_uucw($id_proyek);
    $data['proyek'] = $this->Modelku->baca_data('proyek');
    $data['content']='hasil';
    $this->parser->parse('master',$data);
}

public function hasilakhir(){
    $data['all'] = $this->Modelku->baca_semua();
    $data['content']='hasilakhir';
    $this->parser->parse('master',$data);
}
}??>

```