# Proses Pemodelan Perangkat Lunak Dengan Metode *Opportunistic Approach*

[1]Winny Purbaratri, [2]Moch Fajar Punomo Alam

purbaratri@gmail.com, fajarpurnomoalam@gmail.com
Dosen Tetap STMIK Insan Pembangunan

ABSTRAK

Paper ini akan membahas model proses dalam membangun perangkat lunak. Dengan menggunakan metode tinjauan literatur dari jurnal, buku dan proceeding. Dimana tujuannya untuk mengetahui karakteristik dari model proses Opportunistic Approach. Penelitian ini meluas studi membahas model proses pembuatan perangkat lunak menggunakan metode *Opportunistic Approach*.

Kata Kunci: *Process Model*, Perangkat Lunak, *Software, Software Engineering, Opportunistic.*

## 1. Pendahuluan

### 1.1 Latar Belakang

Tujuan penulisan paper ini adalah membahas secara khusus tentang model proses dari pembuatan perangkat lunak menggunakan *Opportunistic Approach*. Perangkat lunak saat ini sangat berperan dalam kehidupan manusia. Perangkat lunak bahkan memiliki peran yang sangat kritis dalam bidang tertentu dimana kesalahan atau error menjadi hal yang sangat serius. Menentukan model proses dalam pembuatan perangkat lunak tentunya disesuaikan dengan kebutuhan dari konsumen.

### 1.2 Tujuan Penelitian

Tujuan penelitian untuk mengetahui model proses pembuatan perangkat lunak menggunakan *Opportunistic Approach.*

## 2 Landasan Teori

Untuk memperkuat pemahaman dari model proses pembuatan perangkat lunak tulisan yang membahas Opportunistic Approach, tentang metode ini, alangkah baiknya apabila diskusi ini kita mulai dari definisi teori-teori yang digunakan.

### 2.1 Landasan Teori

Software adalah Program komputer dan dokumentasi terkait. Produk perangkat lunak yang dapat dikembangkan untuk pelanggan tertentu atau Produk perangkat lunak yang dapat dikembangkan juga untuk pasar umum.[1]

Proses Model adalah Representasi proses pembuatan perangkat lunak yang disederhanakan, dipresentasikan dari sebuah perspektif yang spesifik. [1] Sebagian besar model proses perangkat lunak didasarkan pada salah satu dari tiga model umum atau paradigma pengembangan perangkat lunak:

1. Pendekatan waterfall
2. Perkembangan Iteratif
3. Component-based software engineering (CBSE)

Untuk model proses pembuatan perangkat lunak yang dibahas pada

paparan ini adalah model proses menggunakan Opportunistic Approach.

*Opportunistic Softwere System Development Model* atau bisa disebut (OSSDM) adalah metode pengembangan perangkat lunak dimana pengembang bergabung bersama untuk menyatukan software yang mereka miliki, beberapa komponen perangkat lunak yang tidak terkait memang tidak dirancang untuk bekerja bersama tetapi bisa menyediakan fungsionalitas yang dibutuhkan dalam sistem baru. [2]

## 3 Metodologi

Studi literature dari review paper, yang dicari dari IEEE Journal, Google Search dan Buku.

## 4. Analisa dan Pembahasan

### 4.1 Karakteristik *Opportunistic Approach*

1. Meski dari kejauhan, pendekatan OSSD Mungkin terlihat seperti paradigma penggunaan ulang perangkat lunak tradisional, ada perbedaan utama. Perangkat lunak dan sistem yang digunakan dalam paradigma penggunaan kembali perangkat lunak yang sudah ada secara sistematis dirancang untuk digunakan kembali dalam konteks tertentu.
2. Itu Proses pembangunan perangkat lunak yang sistematis dan terencana.
3. Metode dan panduannya jelas tentang bagaimana caranya melakukan dan mengelola setiap tugas pembangunan, sehingga menghasilkan tingkat prediktabilitas untuk hasil.
4. Dalam paradigma OSSD, fokusnya adalah pada penggabungan sistem yang tidak pernah dimaksudkan untuk bekerja sama atau bahkan untuk digunakan kembali.
5. Ini tentang mengambil perangkat lunak yang ada sistem dan kinerja minimal coding dengan penekanan pada perekatan mereka bersama-sama-biasanya tanpa interaksi dengan pengembang asli atau akses ke source code untuk mencapai fungsi yang diinginkan baru dengan yang diinginkan melihat.
6. OSSD sangat bergantung pada banyak hal prototyping dan membutuhkan jenis keterampilan, kolaborasi, dan teknik baru.
7. Ini adalah pergeseran besar itu tantangan paradigma proses pengembangan perangkat lunak saat ini.

***Opportunistic Approach*** membutuhkan sesuatu yang besar yaitu perubahan pola pikir:

a. Dari perancangan dan penulisan kode program asli yang memenuhi persyaratan tertentu, ke dunia di mana beberapa peraturan untuk disain, konstruksi, atau manajemen proses ada.
b. Proses pengembangannya ad hoc dan oportunistik, dengan tim terus menerus mengais dan mencoba solusi kandidat

sambil terus-menerus memperhatikan ketidakcocokan arsitektur, perbedaan dalam bahasa pemrograman, cara yang seragam menggambarkan data, dan sebagainya.

c. Penekanan utama adalah pengembangan teknik cerdas, kreativitas, inovasi, dan cara yang paling imajinatif untuk perekatan bersama potongan perangkat lunak yang tampaknya tidak terkait untuk menyediakan sistem interoperable dan maintainable yang memenuhi kebutuhan pengguna.

d. Ini tentang fabrikasi hibrida sistem oleh pemulungan untuk fungsionalitas dimanapun kita dapat menemukannya untuk "mengelas" bersama-sama menjadi sistem komposit. [2]

### 4.2. Model proses *Opportunistic Approach*

*Gambar 1. Model Proses Opportunistic Approach*

1) Bisa kita lihat pada gambar di atas ada tahap "First Prototype" yang artinya rancangan awal pertama yang langsung di sajikan kepada customer.
2) Lalu bila ada kekurangan langsung dilakukan modifikasi sesuai kebutuhan agar sistem benar benar optimal dan customer merasa terpuaskan lalu dilanjutkan pada tahap.
3) "Think of Idea for Improvement" dilakukan kegiatan ide ide baru apa yang ingin di tambahkan pada sistem yang sedang di rancang, jadi sistem ini terus melakukan iterasi sampai muncul ide ide baru dan terus berkembang demi tercapainya kepuasan dari customer dengan sistem yang sedang di rancang sampai benar benar matang lalu di hentikan proses iterasi tersebut, jadi pada model OSSDM ini sama halnya dengan model prototyping & XP model yang banyak melakukan komunikasi (feedback). [3]

## 5. Kesimpulan

1. Model ***Opportunistic Approach*** ini muncul ketika adanya peningkatan permintaan pengguna dan biaya yang mendorong industri perangkat lunak. Model ini membuat komposisi perangkat lunak baru dari perangkat lunak yang sudah tersedia termasuk sistem lainnya.
2. Model ini tidak tediri dari rangkaian-rangkain proses yang jelas dan bertahap. Jadi tidak ada tahapan yang jelas pada model ini. Dimana kita menggunakan pendekatan kemungkinan - kemungkinan yang dalam proses pembuatan software.
3. Namun disini ada 3 aspek utama dalam pembuatan software :
   1) Prototyping
   2) Modify Until Satisfy
   3) Improvement Idea

   Rekayasa cerdas, kreativitas, inovasi, dan cara-cara imajinatif untuk menemukan dan menempelkan software untuk menyediakan sistem interoperable dan di maintenance yang memenuhi kebutuhan pengguna.Mudahnya model ini

memerlukan improvisasi dalam pembuatan software.[4]

**Daftar pustaka:**


1. Sommerville, Ian (2007) Software Engineering Eighth Edition, England: Addison-Wesley Publishers.
2. Ncube, C., et al., "Opportunistic Software System Development: Making Systems from What's Available", IEEE Software, November/December 2008, pp. 38-41.
3. Lando (2017). Software Process And Engineering Model. Diambil 13 September 2017, dari https://blogs.unpas.ac.id/landorinov/2014/10/17/software-and-process-enginering-model/
4. Aditya,F (2013). ***Rational Unified Process & Opportunistic Approach*.** Diambil 14 September 2017, dari http://if-unsika-2011110.blogspot.co.id/2013/04/rational-unified-process.html