

REKAYASA PERANGKAT LUNAK (SOFTWARE ENGINEERING)

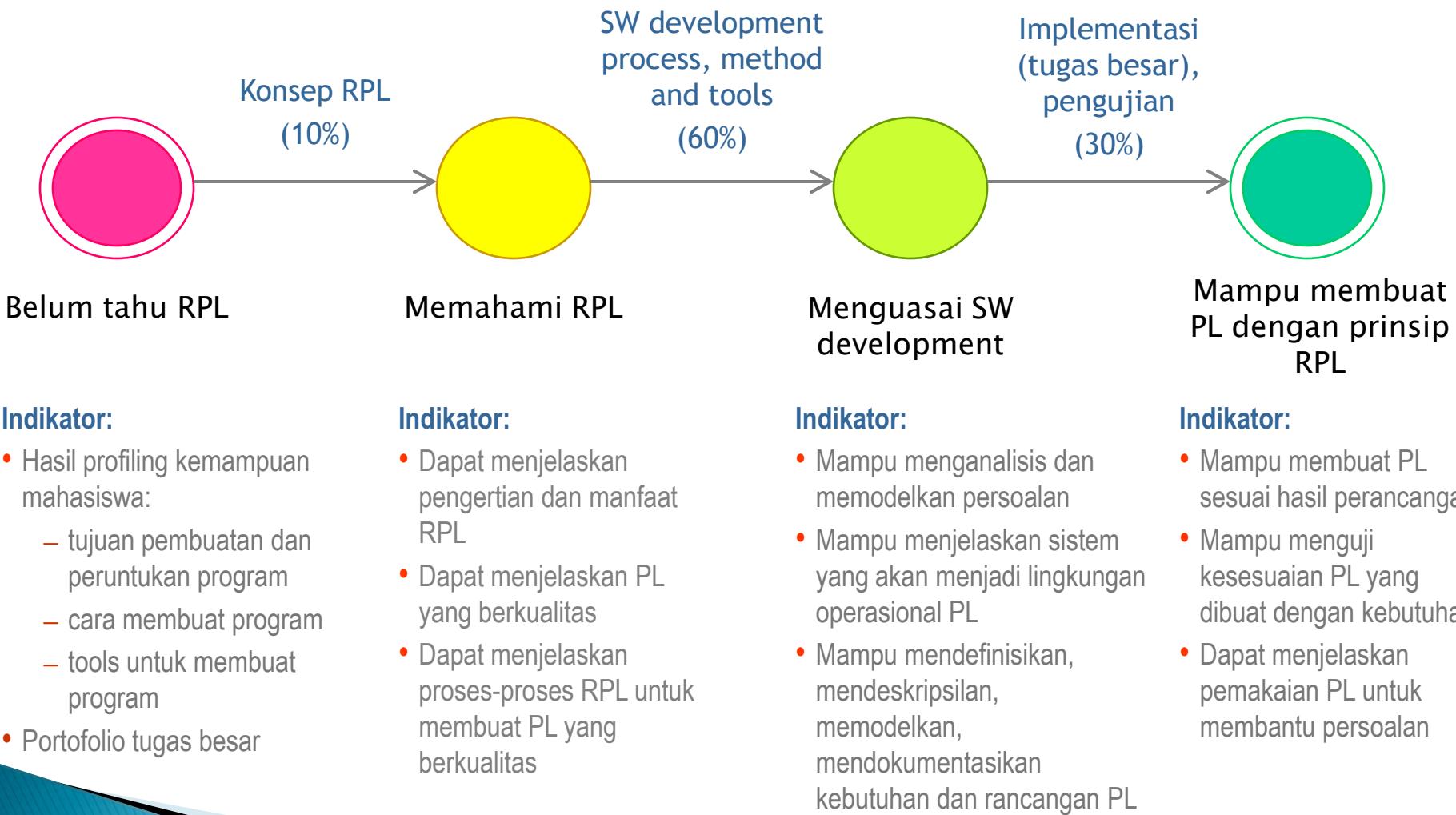
TUJUAN

- ▶ Menyamakan kembali **persepsi** dan **pemahaman** tentang Rekayasa Perangkat Lunak (*Software Engineering*).
- ▶ Memahami kembali **konsep**, **proses**, **metode**, dan **alat bantu** dalam rekayasa perangkat lunak.

MATERI

- ▶ Konsep Rekayasa Perangkat Lunak
- ▶ Software Engineering Process:
 - Software Development
 - Software Documentation
 - Software Project Management
 - Software Quality Assurance

MODEL 4 KEADAAN KOMPETENSI RPL



BAGIAN I

KONSEP REKAYASA PERANGKAT LUNAK

APA YANG DISEBUT PERANGKAT LUNAK?

- ▶ 1) Instruksi (**program komputer**) yang bila dieksekusi dapat menjalankan fungsi tertentu;
- 2) **Struktur data** yang dapat membuat program memanipulasi informasi; dan 3) **Dokumen** yang menjelaskan operasi dan penggunaan program (Roger Pressman).
- ▶ **Program komputer, prosedur, aturan, dan dokumentasi** yang berkaitan dengannya serta **data** yang berkaitan dengan operasi suatu sistem komputer (IEEE).

JENIS-JENIS PERANGKAT LUNAK

- ▶ Perangkat lunak sistem (*system software*)
- ▶ Perangkat lunak waktu nyata (*real-time software*)
- ▶ Perangkat lunak bisnis (*business software*)
- ▶ Perangkat lunak rekayasa dan ilmu pengetahuan (*engineering and scientific software*)
- ▶ *Embedded Software*
- ▶ Perangkat lunak pribadi (*personal software*)
- ▶ Perangkat lunak intelegensia buatan (*artificial intelligent software*)
- ▶ Perangkat lunak lainnya

DOKUMEN PERANGKAT LUNAK

- ▶ Software Project Management Plan (SPMP)
- ▶ Software Requirement Specification (SRS)
- ▶ Software Design Description (SDD)
- ▶ Software Test Plan (STP)
- ▶ Software Test Description (STD)
- ▶ Software Test Result (STR)
- ▶ Software Version
- ▶ User Guide / User Manual

APA YANG DISEBUT RPL?

Proses membuat perangkat lunak dengan menggunakan kaidah-kaidah atau prinsip-prinsip rekayasa sehingga dihasilkan perangkat lunak yang berkualitas.

DEFINISI LAIN (1)

Roger Pressman:

- ▶ Pembentukan dan penggunaan **prinsip rekayasa (*engineering*)** untuk mendapatkan perangkat lunak secara **ekonomis** namun **andal** dan **dapat bekerja secara efisien** pada komputer (dikutip dari Fritz Bauer, 1968).
- ▶ Suatu **disiplin, kaidah** yang mengintegrasikan **proses, metode, dan alat bantu (*tools*)** untuk pembangunan perangkat lunak komputer.

DEFINISI LAIN (2)

IEEE Computer Society:

- 1) The application of a **systematic**, **disciplined**, **quantifiable approach** to the **development**, **operation**, and **maintenance** of software; that is, the application of engineering to software.
- 2) The study of approaches as in (1).

MENGAPA RPL? (1)

- ▶ Perangkat lunak dibuat supaya bisa digunakan (oleh pemakai) untuk **membantu menyelesaikan masalah/ pekerjaan**.
- ▶ Perangkat lunak digunakan oleh pemakai di **lingkungan operasional** dengan **teknologi** tertentu sebagai **bagian dari sistem** yang lebih besar.
- ▶ Perangkat lunak yang dibuat harus **memenuhi apa yang diinginkan** oleh pemakai (bukan yang diinginkan *developer*), **ekonomis** (memberikan keuntungan bagi *developer*), dan **andal** (dapat digunakan dalam jangka waktu yang lama).

MENGAPA RPL? (2)

- ▶ Perangkat lunak sudah diselesaikan dan diserahkan (*delivered*) tetapi tidak pernah digunakan (47%).
- ▶ Pemakai (*user*) sudah membayar untuk perangkat lunak tetapi tidak pernah jadi dan diserahkan (29,7%).
- ▶ Perangkat lunak sudah digunakan tetapi kritis atau ditinggalkan (19%).
- ▶ Perangkat lunak digunakan setelah dilakukan modifikasi (3%).
- ▶ Hanya sebagian kecil perangkat lunak yang dapat digunakan sebagaimana mestinya (2%).

KAPAN RPL ?

- ▶ Ukuran perangkat lunak besar:
 - Cakupan atau ruang lingkup persoalan luas dan dalam.
 - Kompleksitas masalah dan tingkat kekritisan tinggi.
- ▶ Melibatkan banyak personal pelaksana.
- ▶ Waktu dan anggaran pengembangan terbatas.

MANFAAT RPL

- ▶ Perangkat lunak yang dihasilkan **sesuai dengan kebutuhan** yang diinginkan.
- ▶ Perangkat lunak **dapat digunakan dan beroperasi dengan benar** di lingkungan sebenarnya.
- ▶ Perangkat lunak **memberikan manfaat** bagi pemakai yang menggunakannya.
- ▶ **Biaya** yang dikeluarkan untuk membuatnya **rendah (efisien), efektif dan sesuai dengan anggaran** yang telah ditetapkan.
- ▶ **Tepat waktu**, baik saat pembuatan, penyerahan ke pemakai, maupun instalasinya.
- ▶ Setiap tahap pekerjaan **terjamin kualitasnya, terdokumentasi, dan dapat dipertanggungjawabkan kebenarannya** (ada proses verifikasi dan validasi).

SOFTWARE ENGINEERING BODY OF KNOWLEDGE

SWEBOK Knowledge Areas:

- ▶ Software requirements
- ▶ Software design
- ▶ Software construction
- ▶ Software testing
- ▶ Software maintenance
- ▶ Software configuration management
- ▶ Software engineering management
- ▶ Software engineering process
- ▶ Software engineering tools and methods
- ▶ Software quality

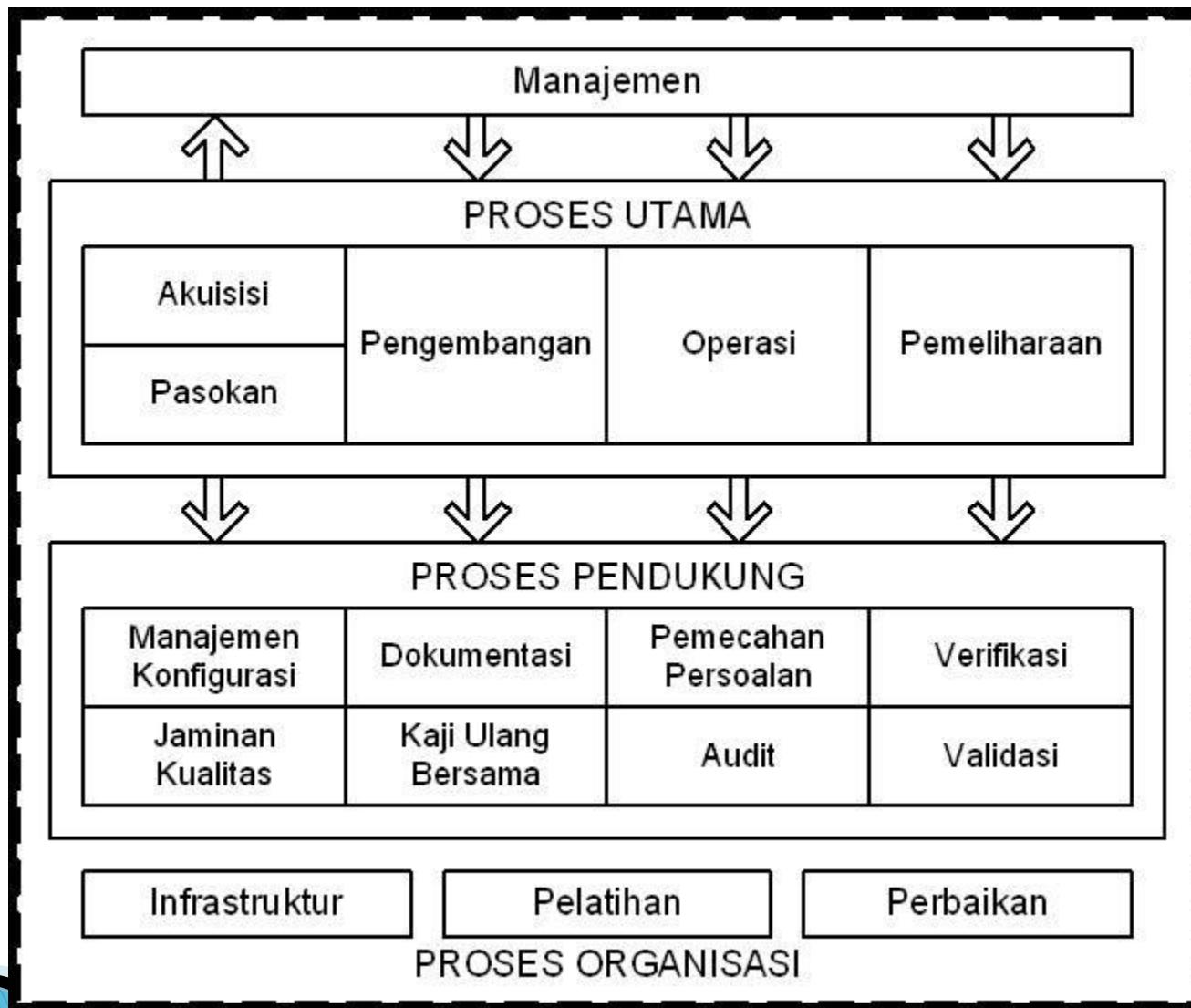
BAGIAN II

PROSES REKAYASA PERANGKAT LUNAK

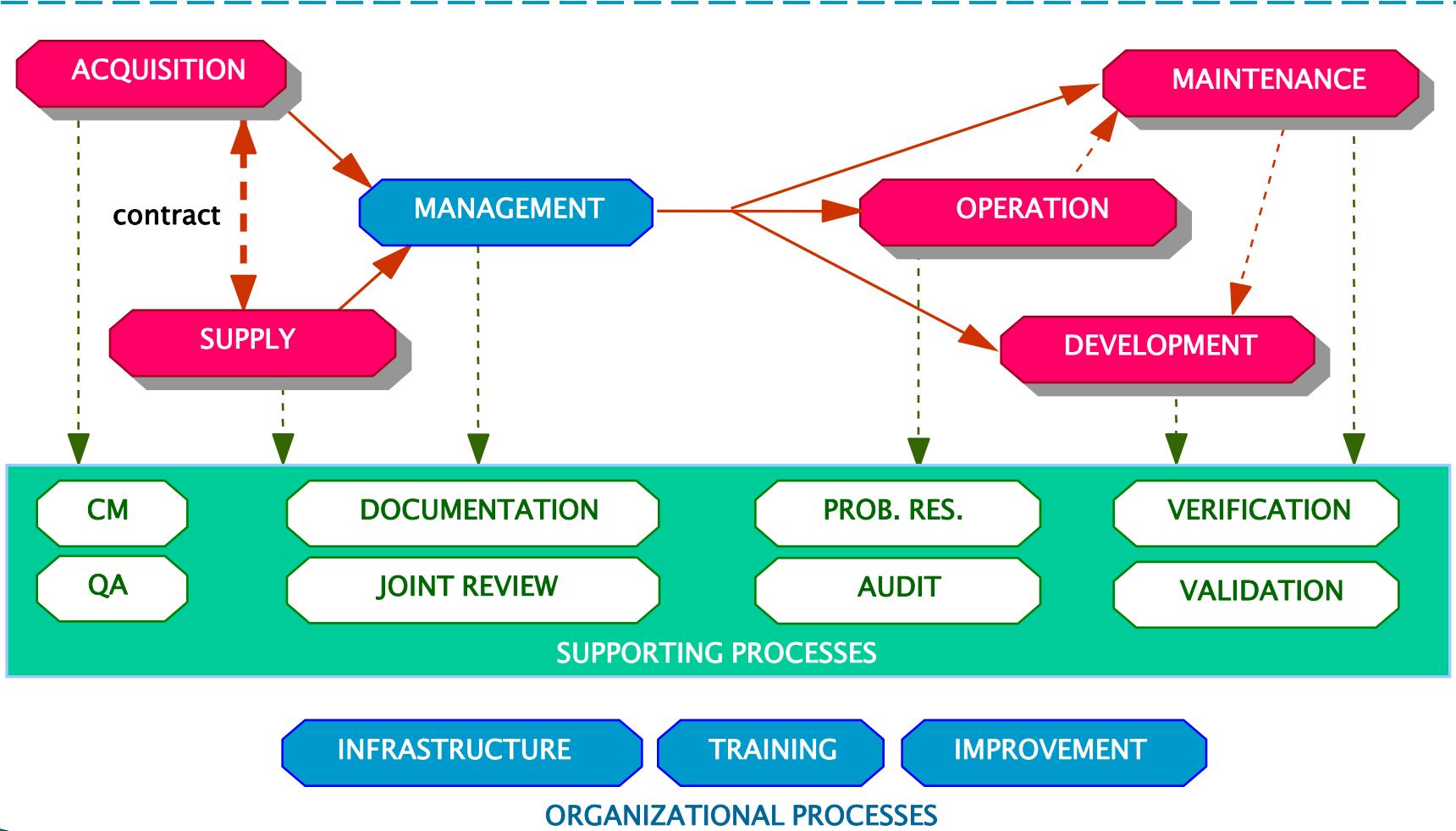
SOFTWARE ENGINEERING PROCESS

- Seri aktivitas yang harus dilaksanakan selama siklus hidup perangkat lunak.
- IEEE/EIA 12207 Standard for Information Technology:
 - Proses utama (*primary processes*)
 - Proses pendukung (*supporting processes*)
 - Proses organisasi (*organizational processes*)

SOFTWARE ENGINEERING PROCESS

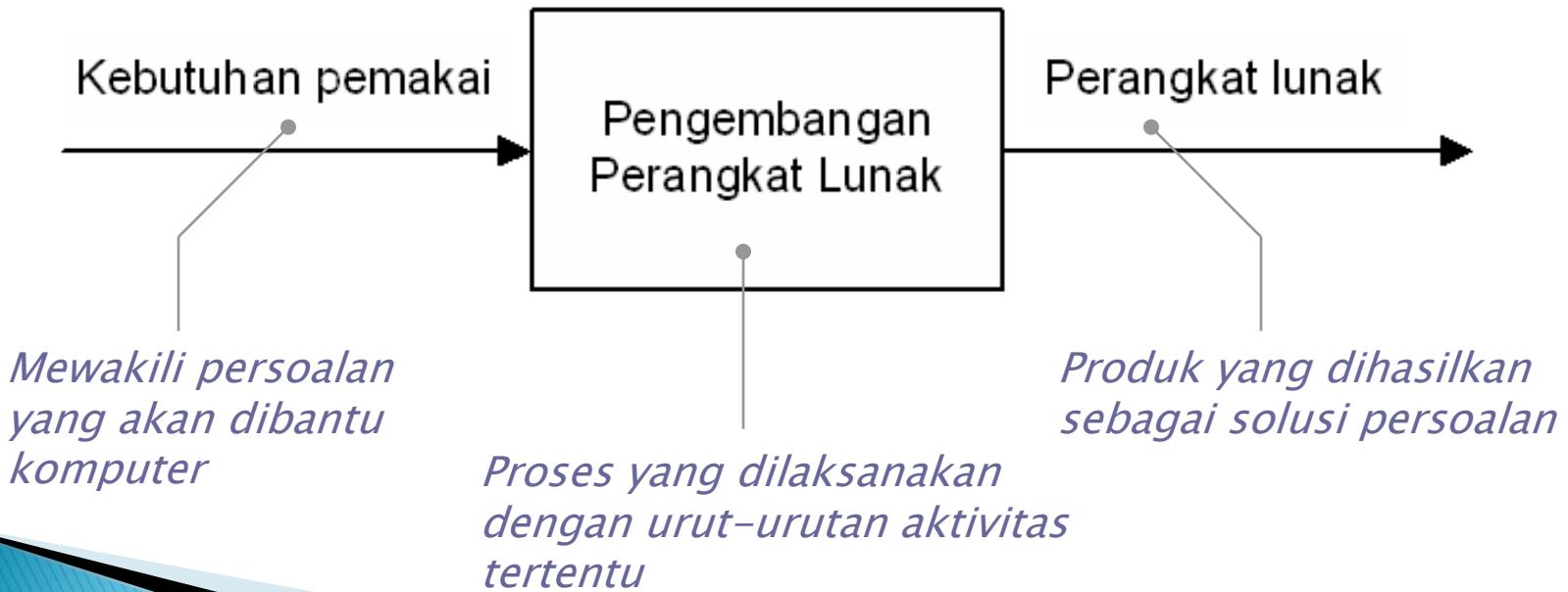


SOFTWARE ENGINEERING PROCESS – IEEE/EIA 12207



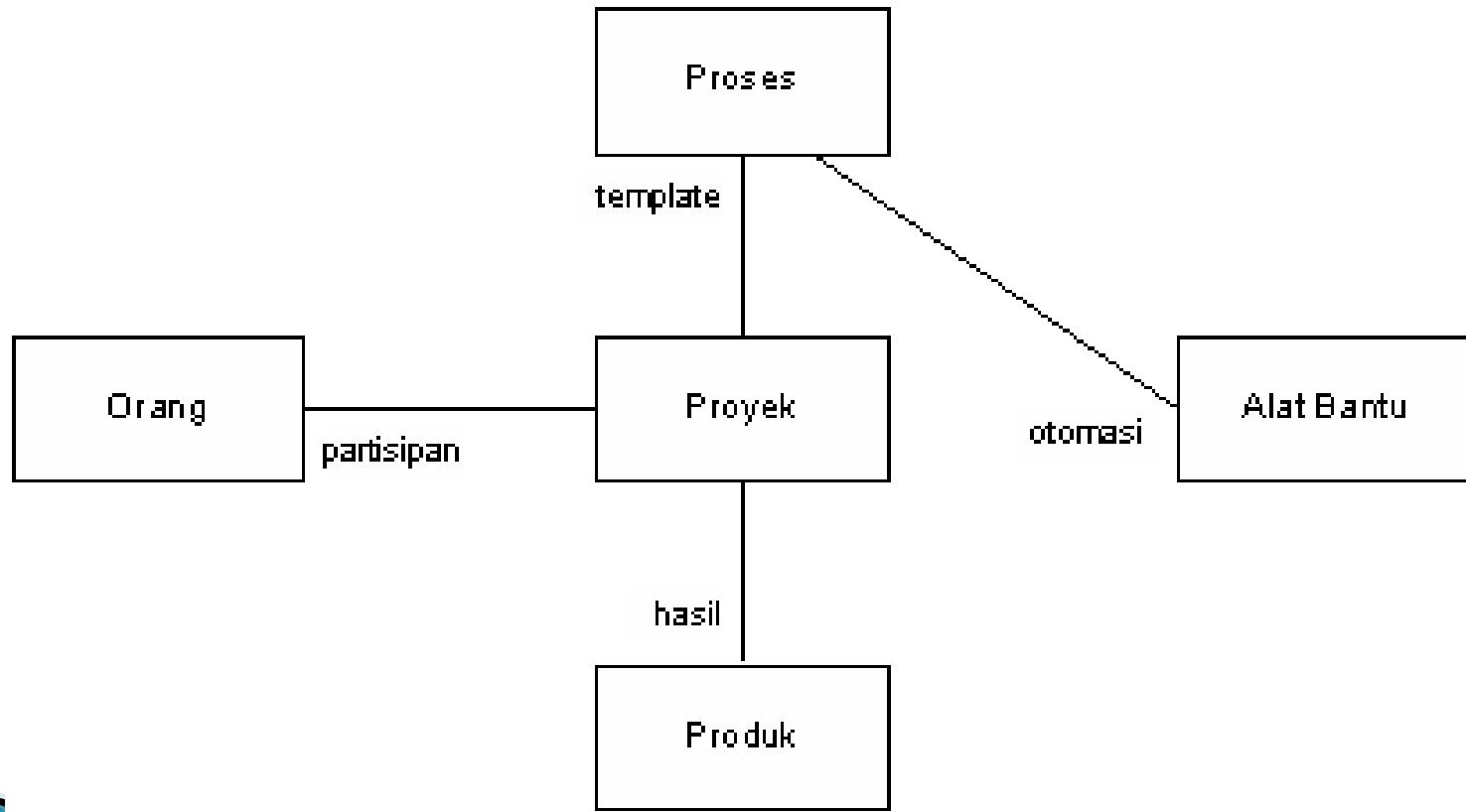
PENGEMBANGAN PERANGKAT LUNAK (1)

- ▶ Proses dimana **persoalan/kebutuhan pemakai** diterjemahkan menjadi **produk perangkat lunak** melalui suatu rangkaian aktivitas tertentu sesuai **model proses, metode, dan alat bantu** yang digunakan.



PENGEMBANGAN PERANGKAT LUNAK (2)

Elemen-elemen proses pengembangan:

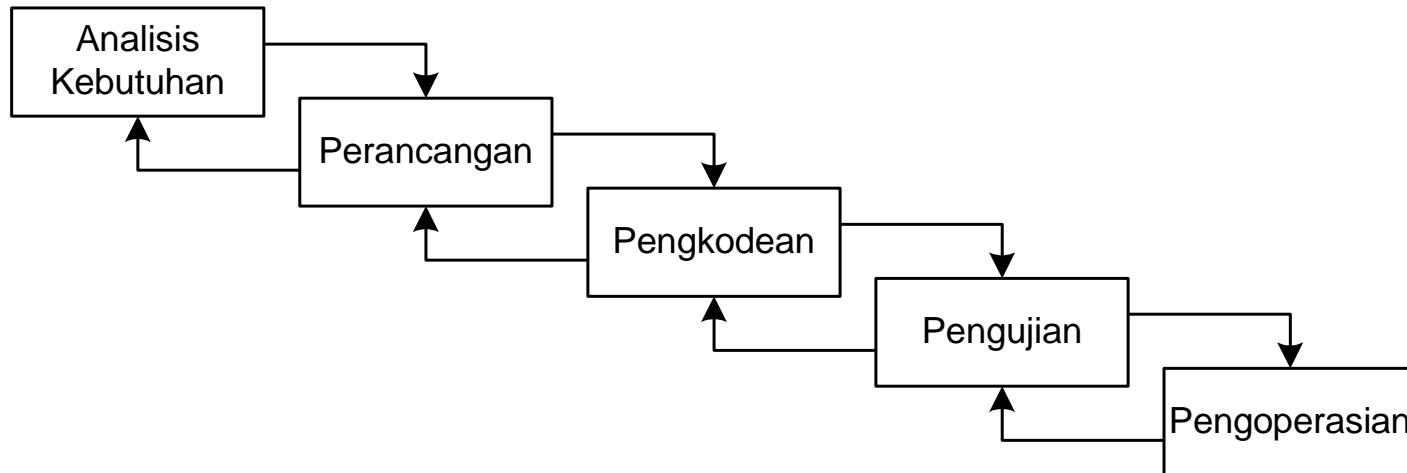


MODEL PROSES PENGEMBANGAN

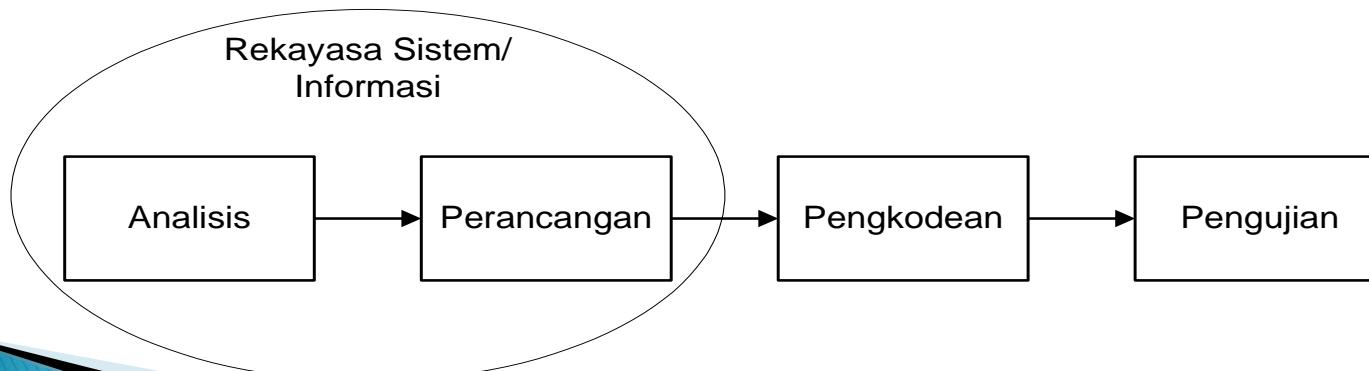
- ▶ **Cara** atau **strategi** bagaimana perangkat lunak **dibuat** sedemikian rupa sehingga produk perangkat lunak tersebut **dapat diwujudkan**.
- ▶ Beberapa model proses pengembangan PL:
 - Waterfall / Linear-Sequential Model
 - Prototyping Model
 - Spiral Model
 - Rational Unified Process (RUP)
 - Extreme Programming (XP)

BEBERAPA MODEL PROSES

Waterfall Model

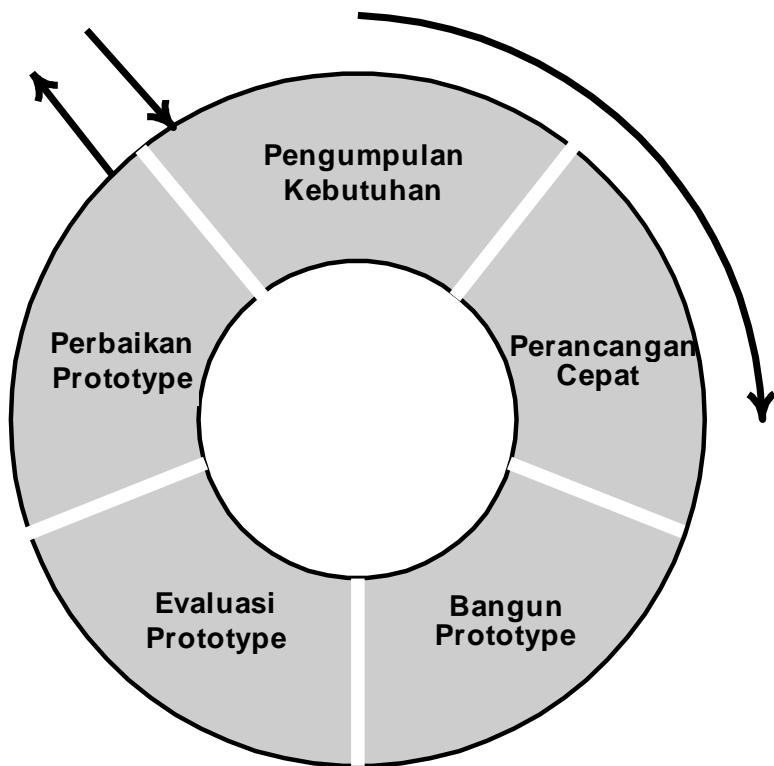


• Linear-Sequential Model

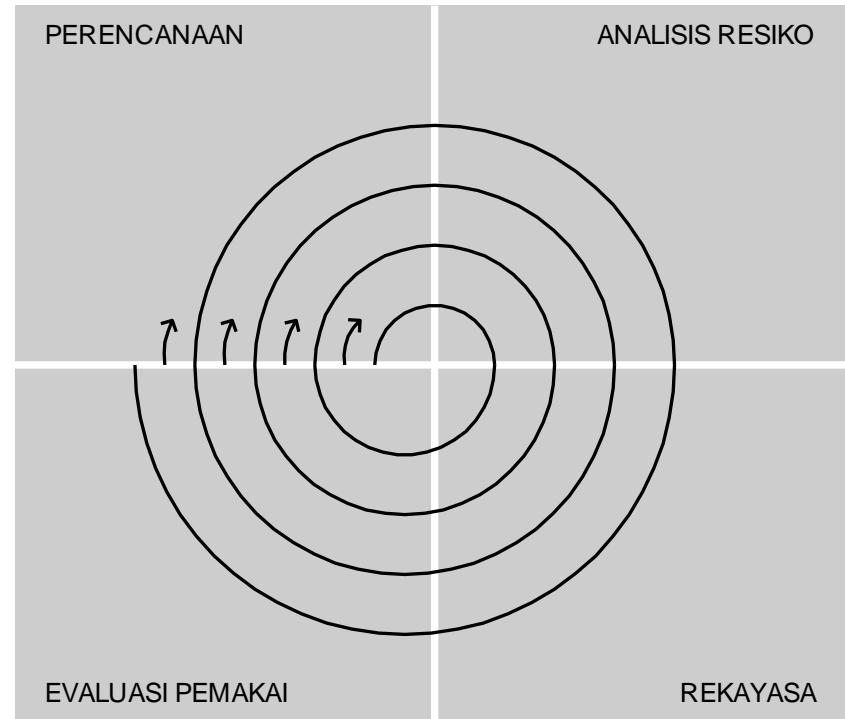


BEBERAPA MODEL PROSES (lanjutan)

▶ Prototyping Model

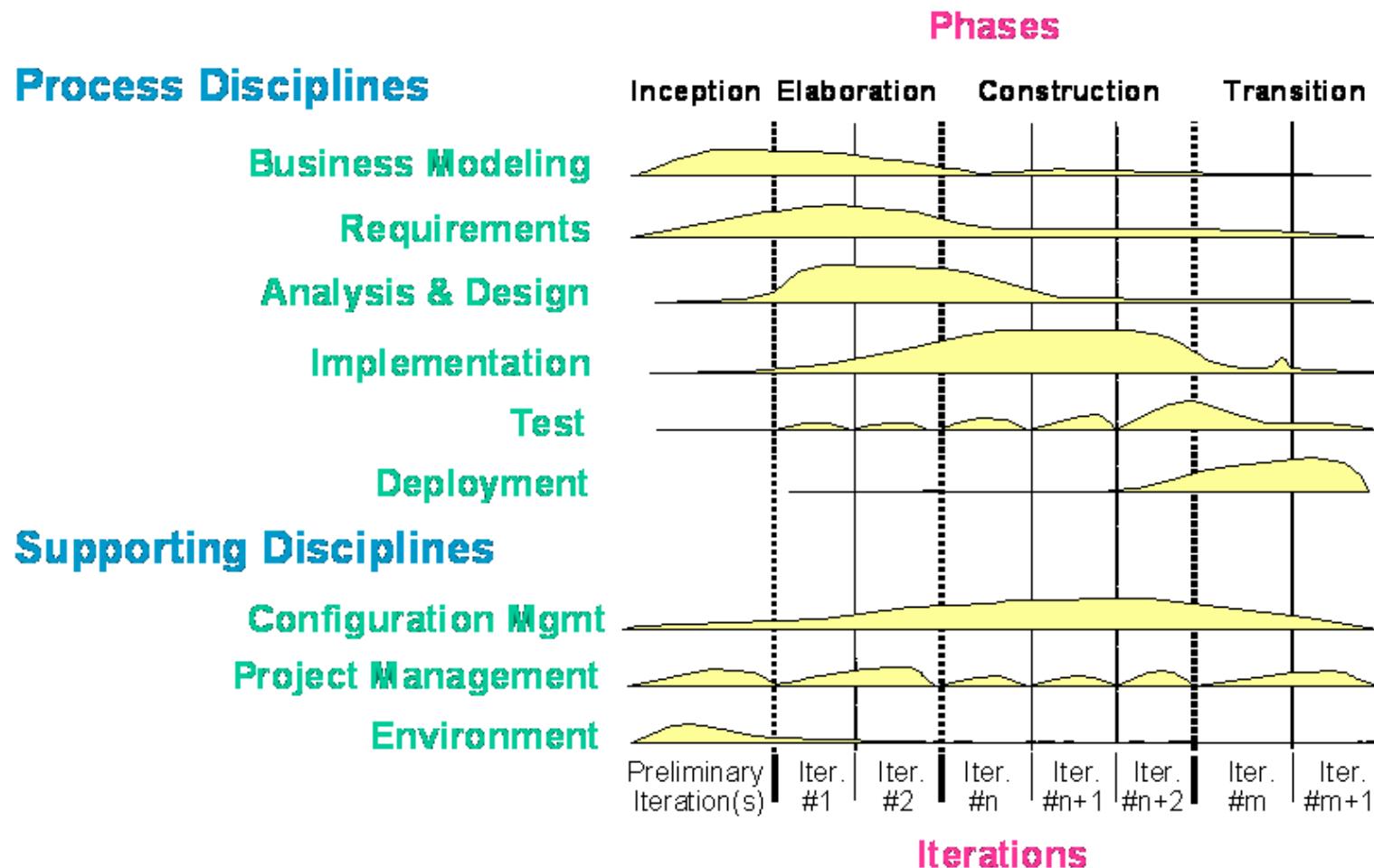


• Spiral Model



BEBERAPA MODEL PROSES (lanjutan)

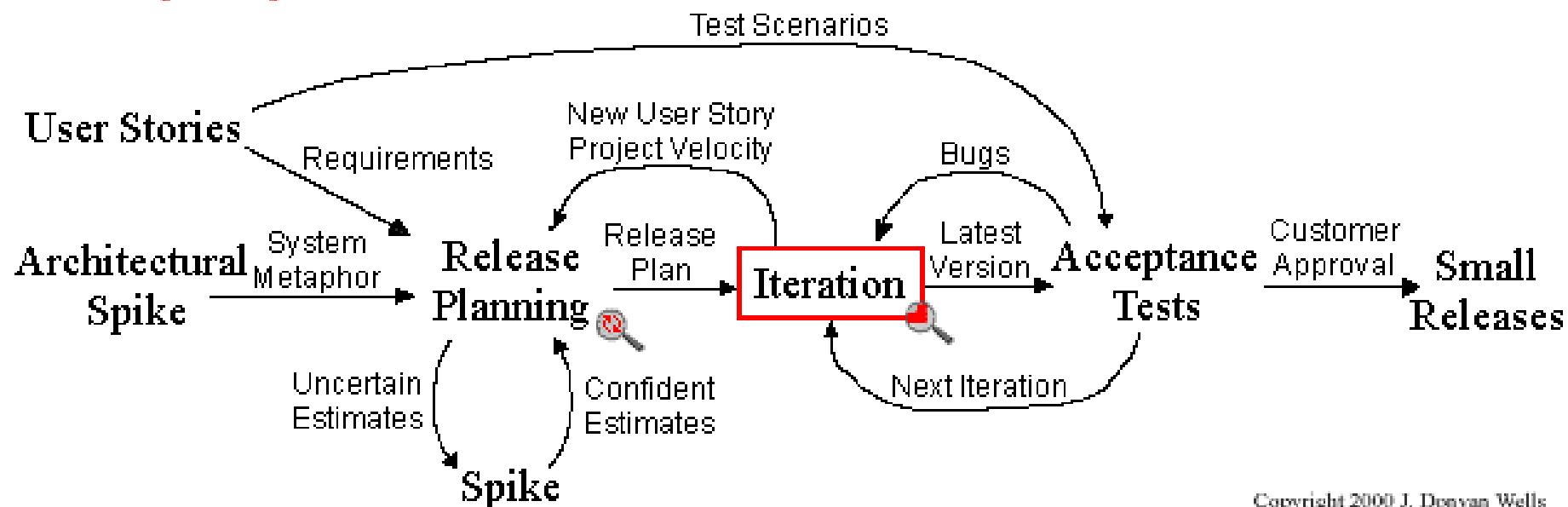
- Rational Unified Process



BEBERAPA MODEL PROSES (lanjutan)



Extreme Programming Project



METODE PENGEMBANGAN

- ▶ Pendekatan, sudut pandang, atau kumpulan aturan yang harus diikuti untuk menyelesaikan tahap-tahap aktivitas pengembangan perangkat lunak.
- ▶ Beberapa metode pengembangan PL:
 - Konvensional atau tradisional (1955)
 - Berorientasi Data (1975)
 - Berorientasi Aliran Data atau Proses (1976)
 - Berorientasi Objek (1980an)

METODE KONVENTSIONAL

- ▶ Sudut pandang pengembangan adalah alur (prosedur) kerja pada sistem fisik organisasi.
- ▶ Prinsip pengembangan:
 - Dokumen apa yang menjadi media data atau informasi
 - Bagaimana dokumen tersebut terbentuk dan mengalir dari satu bagian organisasi ke bagian organisasi yang lain
 - Proses apa saja yang dilakukan terhadap dokumen tersebut
 - Proses mana yang akan dibantu komputer

METODE BERORIENTASI DATA

- ▶ Sudut pandang pengembangan adalah struktur data dari dokumen masukan/keluaran yang digunakan dalam sistem .
- ▶ Prinsip pengembangan:
 - Mengidentifikasi entitas atau item-item yang menjadi objek informasi berikut operasi-operasinya.
 - Menyatakan struktur informasi secara hirarki dengan menggunakan konstruksi *sequence*, *selection* dan *repetition*.
 - Memetakan hirarki struktur informasi menjadi struktur program.

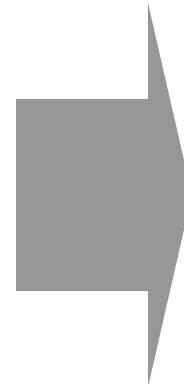
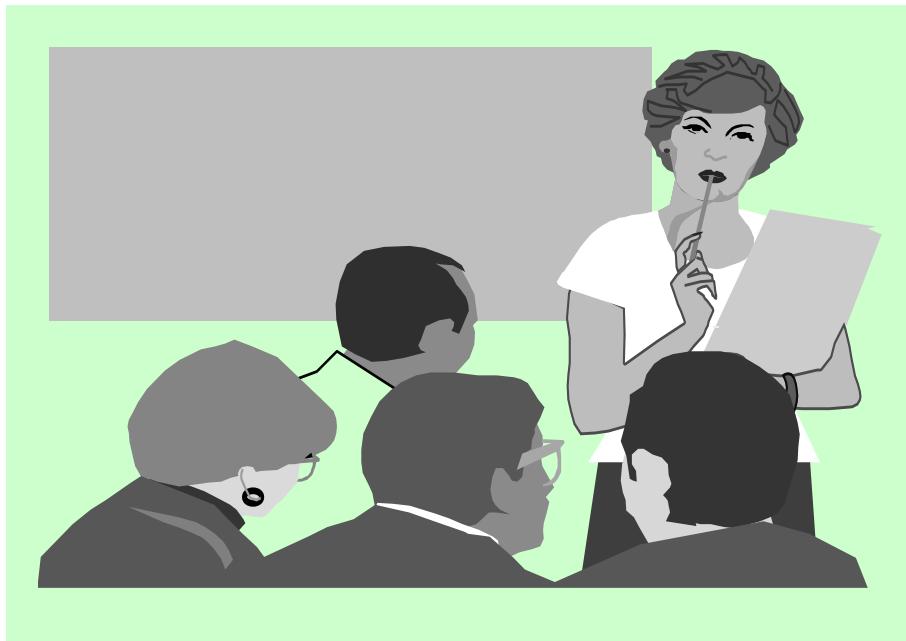
METODE BERORIENTASI FUNGSI

- ▶ Perangkat lunak dianggap sebagai kumpulan fungsi atau proses transformasi data:
 - data masukan
 - proses transformasi
 - data keluaran/hasil transformasi
 - keadaan awal dan akhir
 - perubahan (dari keadaan awal ke akhir)
 - aksi untuk mengubah keadaan

METODE BERORIENTASI OBJEK

- ▶ Dekomposisi persoalan menjadi objek-objek yang berkorrespondensi dengan dunia nyata.

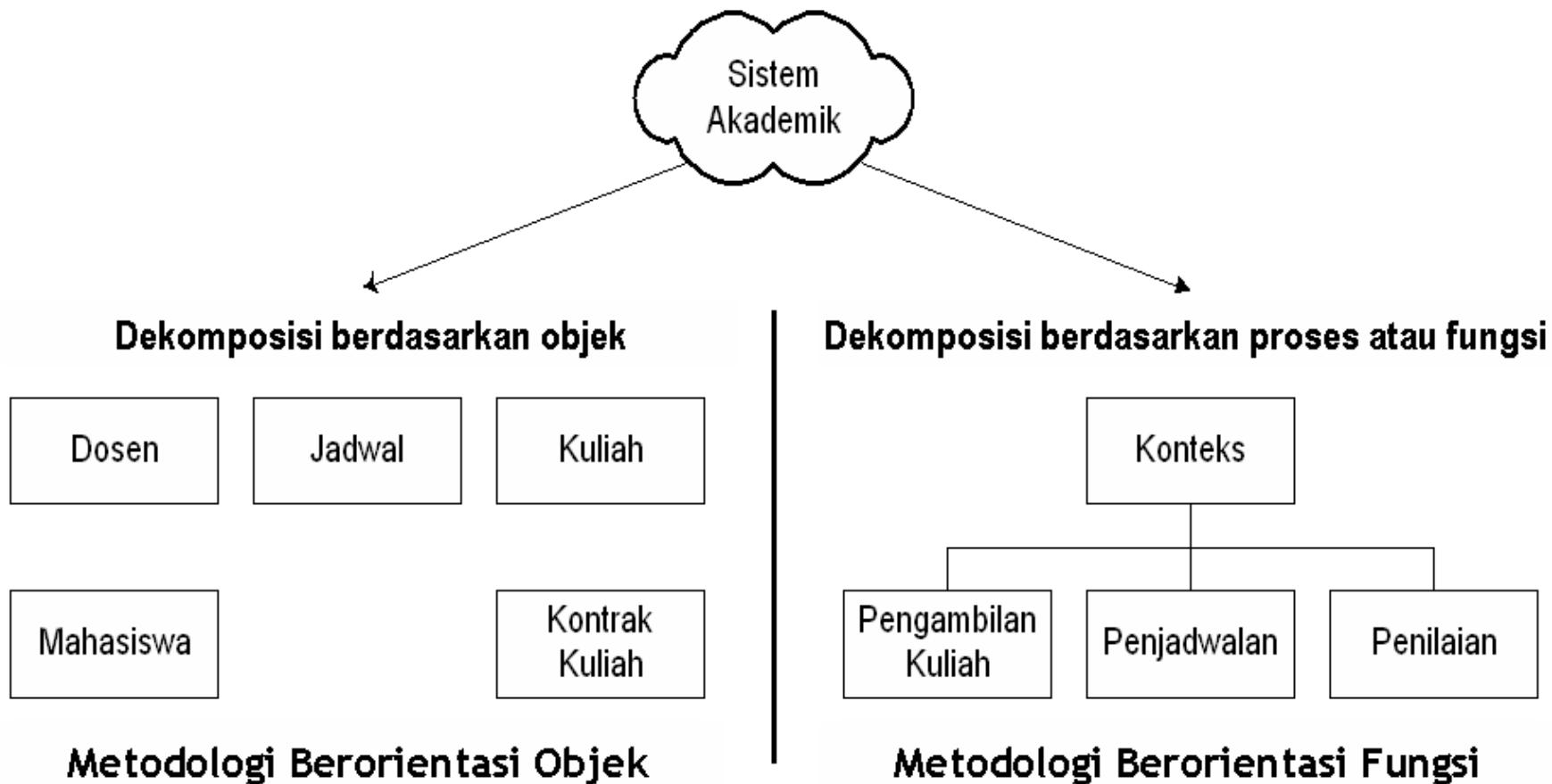
Persoalan: PERKULIAHAN



Objek:

- Dosen
- Mahasiswa
- Kuliah
- Nilai

OBJECT ORIENTED VS FUNCTIONAL



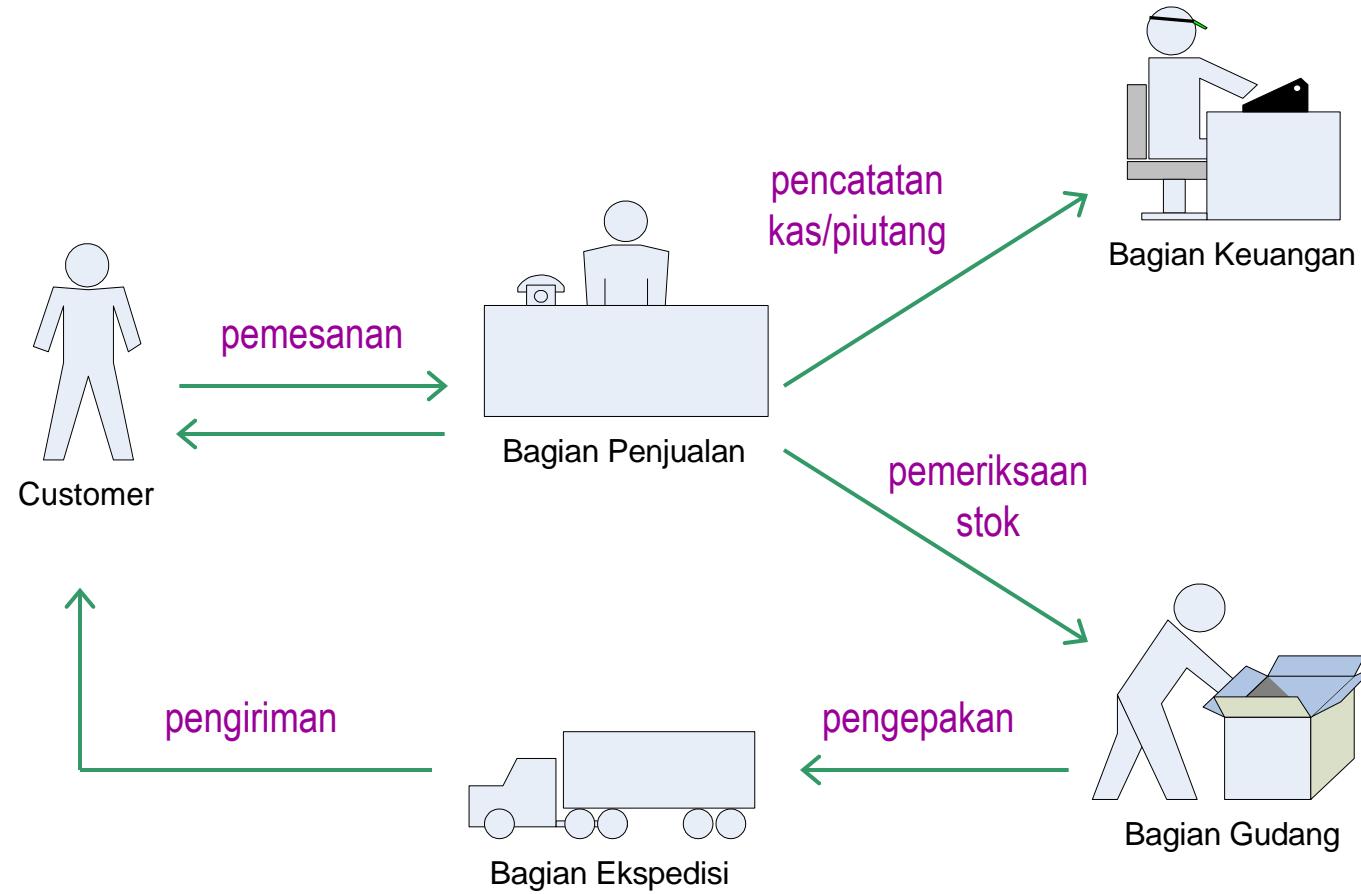
ALAT BANTU

- ▶ Perangkat bantu atau kakas otomatis dan semi-otomatis yang akan digunakan untuk mendukung proses dan metode.
- ▶ Bentuk-bentuk alat bantu pengembangan:
 - **Diagram-diagram** untuk memodelkan hasil setiap tahap pengembangan.
 - **Perangkat lunak** untuk membantu pelaksanaan analisis, perancangan, pembuatan program, atau pengelolaan proyek.
 - **Bahasa pemrograman** untuk penulisan program.

ALAT BANTU PEMODELAN PERSOALAN BISNIS

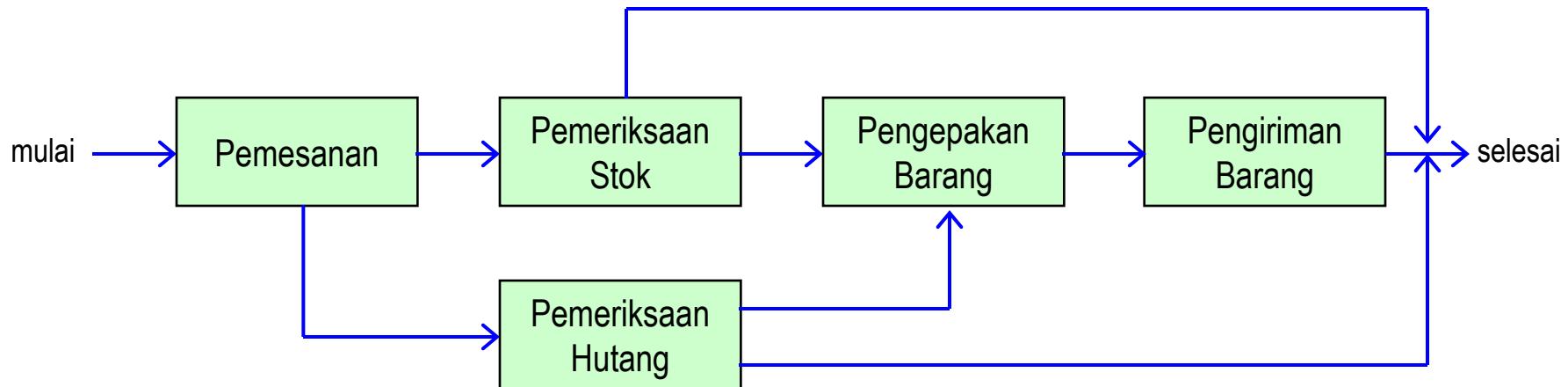
- Workflow Diagram
- Event Model
- Value Chain Diagram
- H – Method
- Business Modeling:
 - Business Use Case Diagram
 - Business Object Model
 - Business Activity Diagram

CONTOH WORKFLOW DIAGRAM



EVENT MODEL

- Prinsip pelaksanaan analisis seperti workflow, akan tetapi fokus ke kejadian (*event*) bukan kronologis pekerjaan.
- Contoh:



VALUE CHAIN



Inbound Logistics

Materials receiving, storing, and distribution to manufacturing premises
Transforming inputs into finished products.

Operations

Storing and distributing products

Outbound Logistics

Promotions and sales force

Marketing and Sales

Service to maintain or enhance product value

Service

Support of entire value chain, e.g. general management planning, financing, accounting, legal services, government affairs, and QM

Corporate Infrastructure

Recruiting, hiring, training, and development

Human Resources Man.

Improving product and manufacturing process

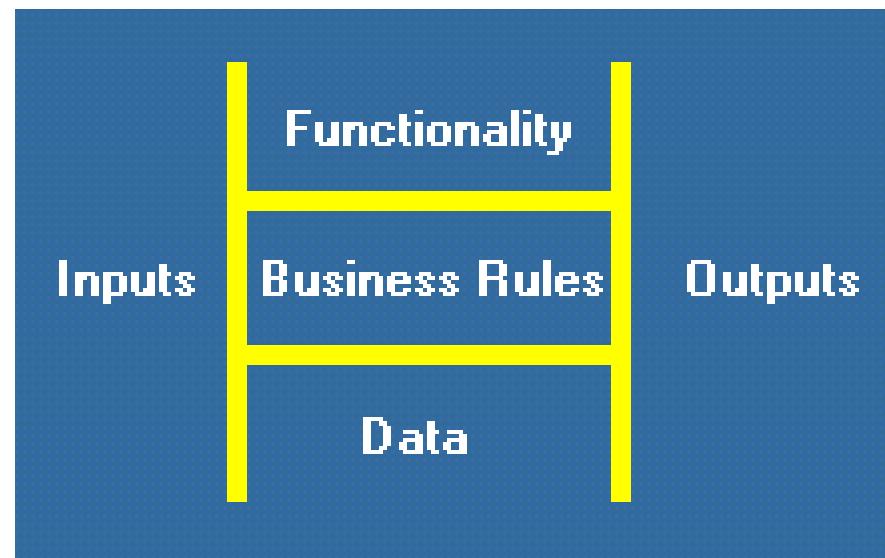
Technology Development

Purchasing input

Procurement

H - METHOD

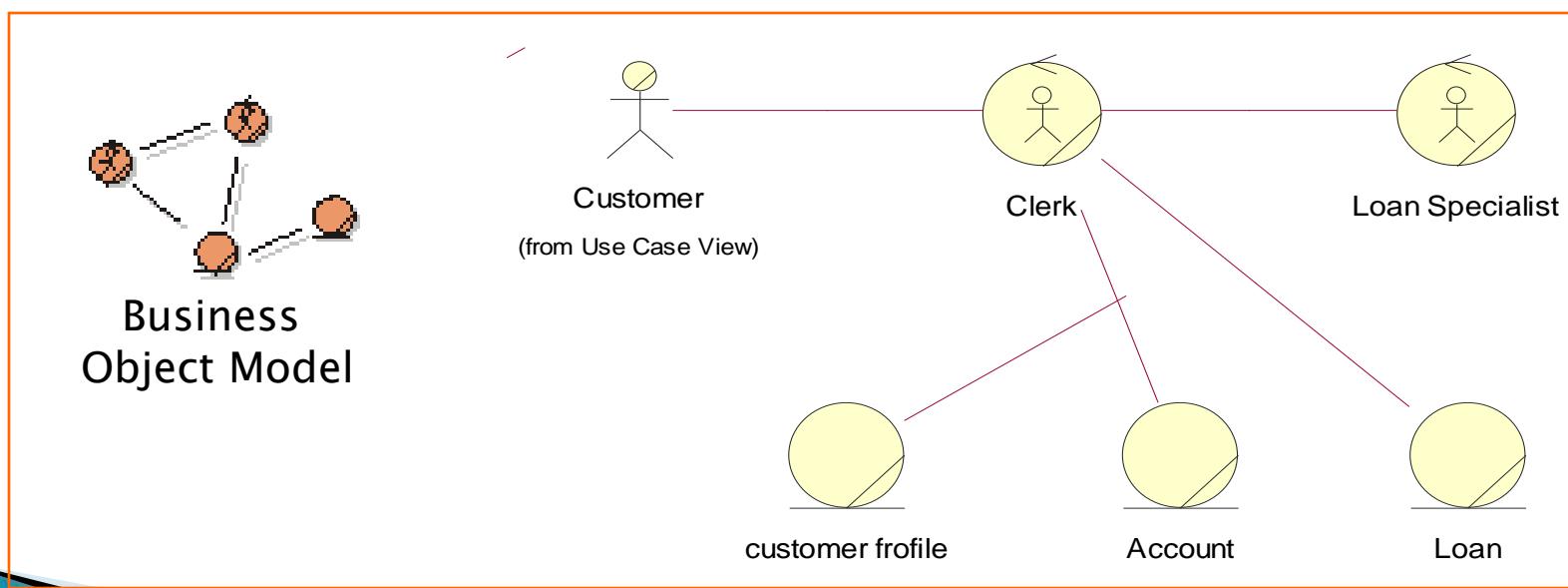
- **Inputs.** What the person needs to do their job.
- **Outputs.** What the person produces.
- **Functionality.** What the person does.
- **Business Rules.** What rules govern the way the person works.
- **Data.** The people places and things the person needs to keep track of.



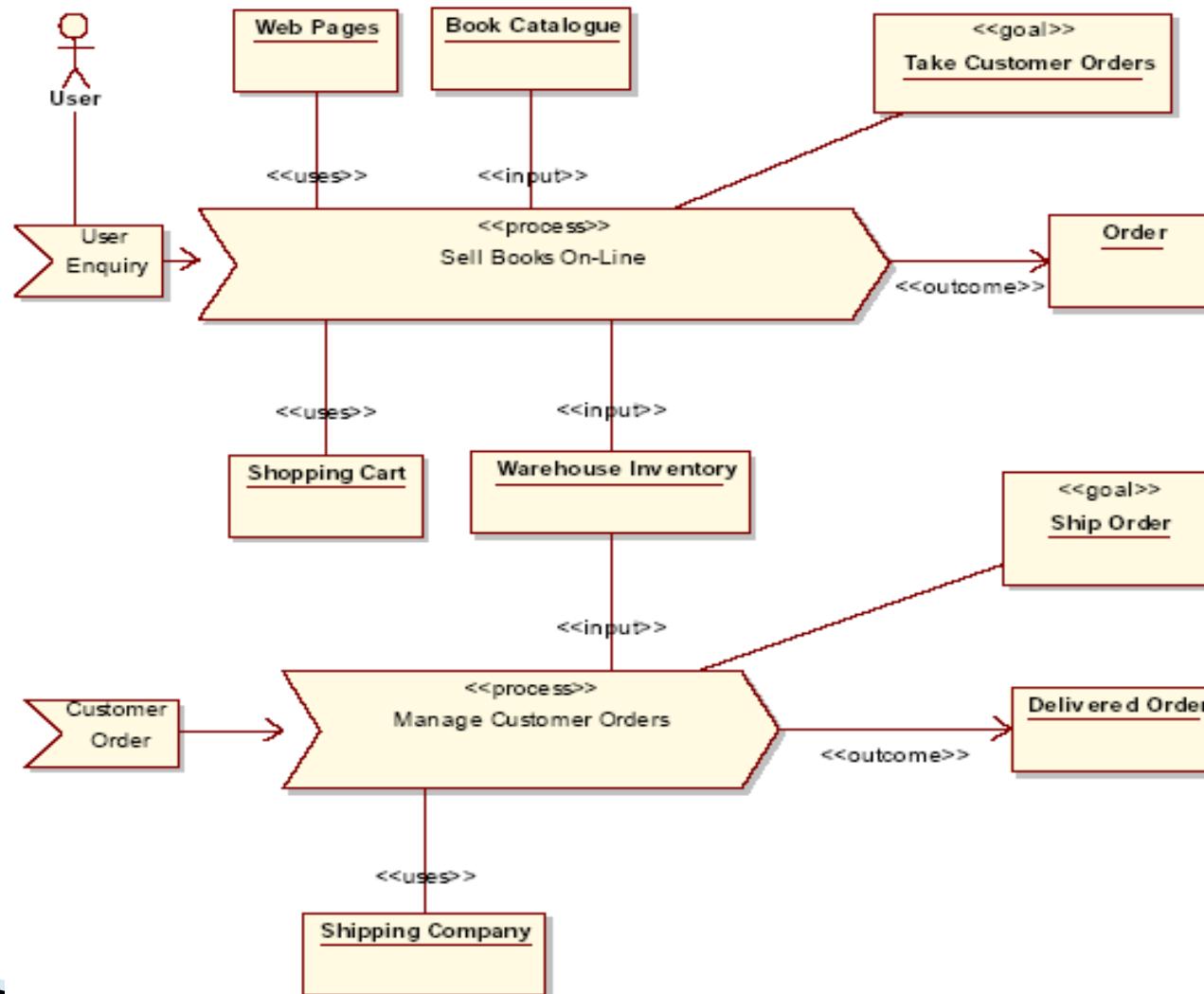
CONTOH H - METHOD

Inputs <ul style="list-style-type: none">• Orders• Customers• Credit Rating• Stock Levels	Functionality <ul style="list-style-type: none">• Check credit rating• Check stock availability• Reserve stock• Backorder stock• Prepare Packing Slip• Advise Accounts of order value• Confirm to customer	Outputs <ul style="list-style-type: none">• Packing Slips• Credit advice• Order to Accounts• Confirmation to customer
	Business Rules <ul style="list-style-type: none">• Don't process if over credit limit• Check with client before backorder	
	Data <ul style="list-style-type: none">• Clients• Order• Backorders• Packing Slips• Stock• Credit Limits• Order Estimated Value	

CONTOH BUSINESS MODELING (1)



CONTOH BUSINESS MODELING (2)



ALAT BANTU PENGEMBANGAN PERANGKAT LUNAK

	Tahap Pengembangan	Alat Bantu
Konvensional	Analisis	Flowmap
	Perancangan	System Flowchart, Program Flowchart
	Implementasi	Bahasa pemrograman
Fungsi	Tahap Pengembangan	Alat Bantu
	Analisis	DFD, DD, p-spec, ERD
	Perancangan	Structure chart, pseudo-code
	Implementasi	Bahasa pemrograman prosedural
Objek	Tahap Pengembangan	Alat Bantu
	Analisis dan Perancangan	UML
	Implementasi/Pemrograman	Bahasa pemrograman objek

AKTIVITAS PENGEMBANGAN (1)

- ▶ Analisis dan pemodelan **persoalan**
 - Analisis (pemahaman) persoalan
 - Pemodelan persoalan
- ▶ Analisis dan perancangan **sistem** (jika sistem belum ada/belum dibuat)
- ▶ Pembuatan **perangkat lunak**:
 - Analisis Kebutuhan
 - Perancangan
 - Konstruksi / Implementasi / *Coding*
 - Pengujian
 - *Deployment*

AKTIVITAS PENGEMBANGAN (2)

Layer	Aktivitas	Pelaksana
Persoalan	Analisis Persoalan	Subject Matter Analyst, System Analyst
	Pemodelan Persoalan	
Sistem	Analisis Sistem	System Analyst, DBA, Network Engineer
	Perancangan Sistem	
Perangkat Lunak	Analisis Kebutuhan	System Analyst, Software Architect
	Perancangan	
	Implementasi	Programmer
	Pengujian	Software Tester
	<i>Deployment</i>	Deployment Team

PEMAHAMAN DAN PEMODELAN PERSOALAN

- ▶ **Mempelajari** dan **memahami** persoalan yang akan dibuat perangkat lunaknya sesuai domain atau ranah persoalannya.
- ▶ **Memodelkan** persoalan (yang sudah dipahami) dengan menggunakan **alat bantu pemodelan** tertentu.
- ▶ **Menentukan solusi** (dan teknologi) untuk menyelesaikan persoalan.

DOMAIN PERSOALAN

- ▶ Konteks atau **ruang lingkup kajian** persoalan pada dunia nyata:
 - Bisnis
 - Matematika
 - Grafik, citra
 - Artificial Intelligence (AI)
 - Komunikasi data
 - Multimedia
 - Sistem operasi
 - dll.
- ▶ Pemahaman persoalan untuk domain tertentu dapat melibatkan disiplin ilmu diluar Informatika.

UNTUK DIINGAT

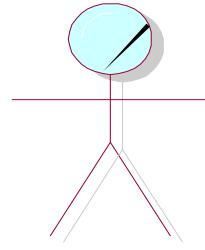
- ▶ Perangkat lunak dibuat supaya bisa digunakan (oleh pemakai) untuk **membantu menyelesaikan masalah/ pekerjaan.**
- ▶ Oleh karena itu, hal pertama yang harus dilakukan saat membuat perangkat lunak adalah **memahami masalah/ pekerjaan** yang akan dibantu penyelesaiannya.

The first step in solving any problem is to understand it. We often propose solutions to problems that we do not understand and then are surprised when the solutions fail to have the anticipated effect.

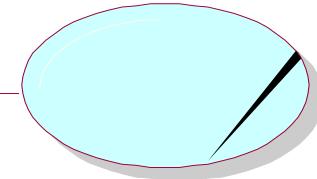
CONTOH KASUS

- ▶ Membuat perangkat lunak untuk membantu mengolah data promosi objek wisata dan penawaran investasi potensi wisata.
- ▶ Domain persoalan: Bisnis.

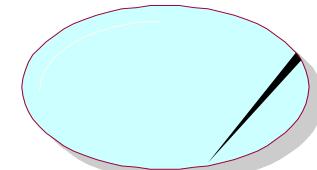
PEMODELAN PERSOALAN – BUSINESS USE CASE DIAGRAM



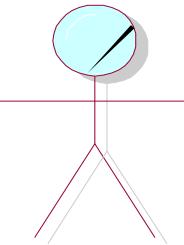
Wisatawan



Promosi Objek Wisata



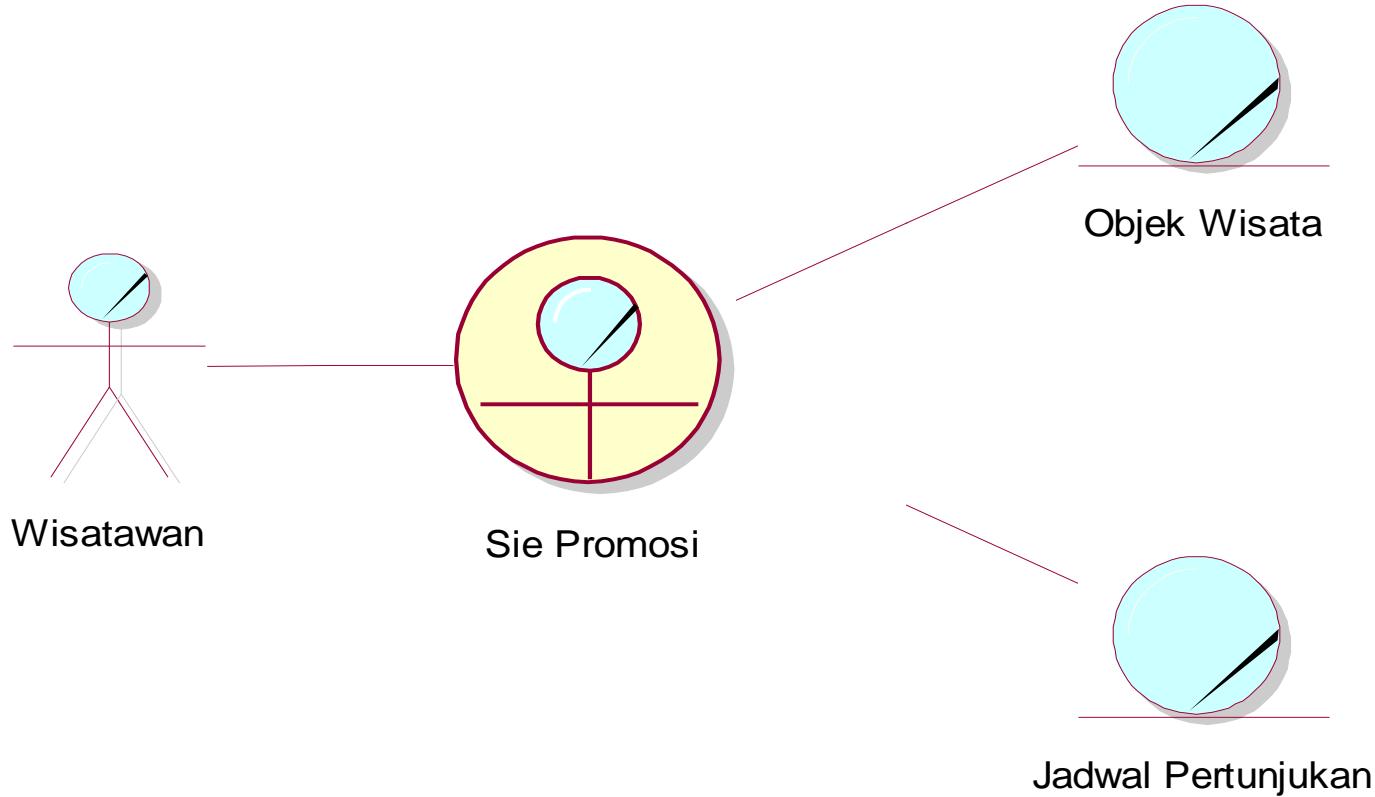
Pengelolaan Potensi Wisata



Calon Investor

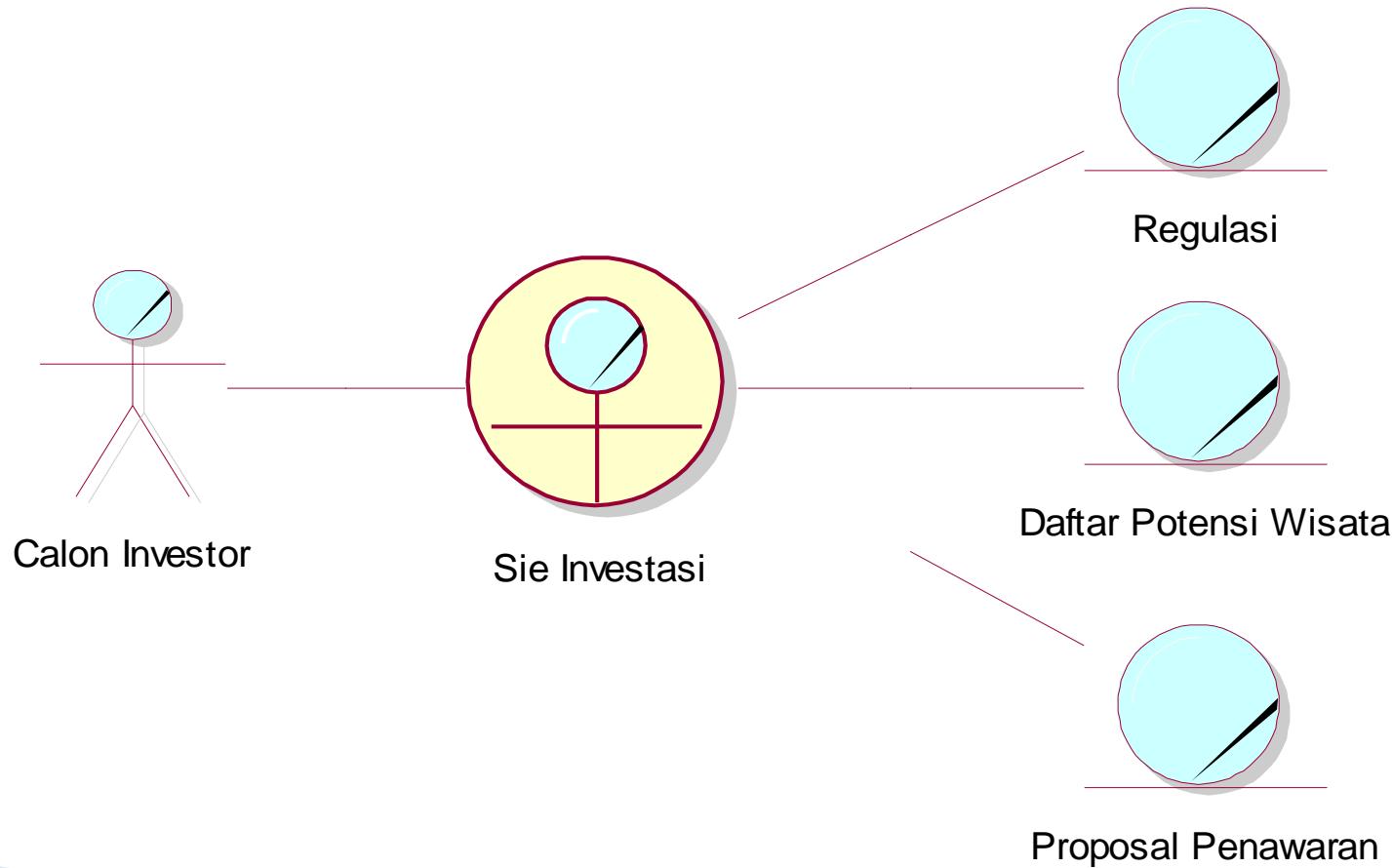
PEMODELAN PERSOALAN – BUSINESS OBJECT MODEL

1. Promosi Objek Wisata

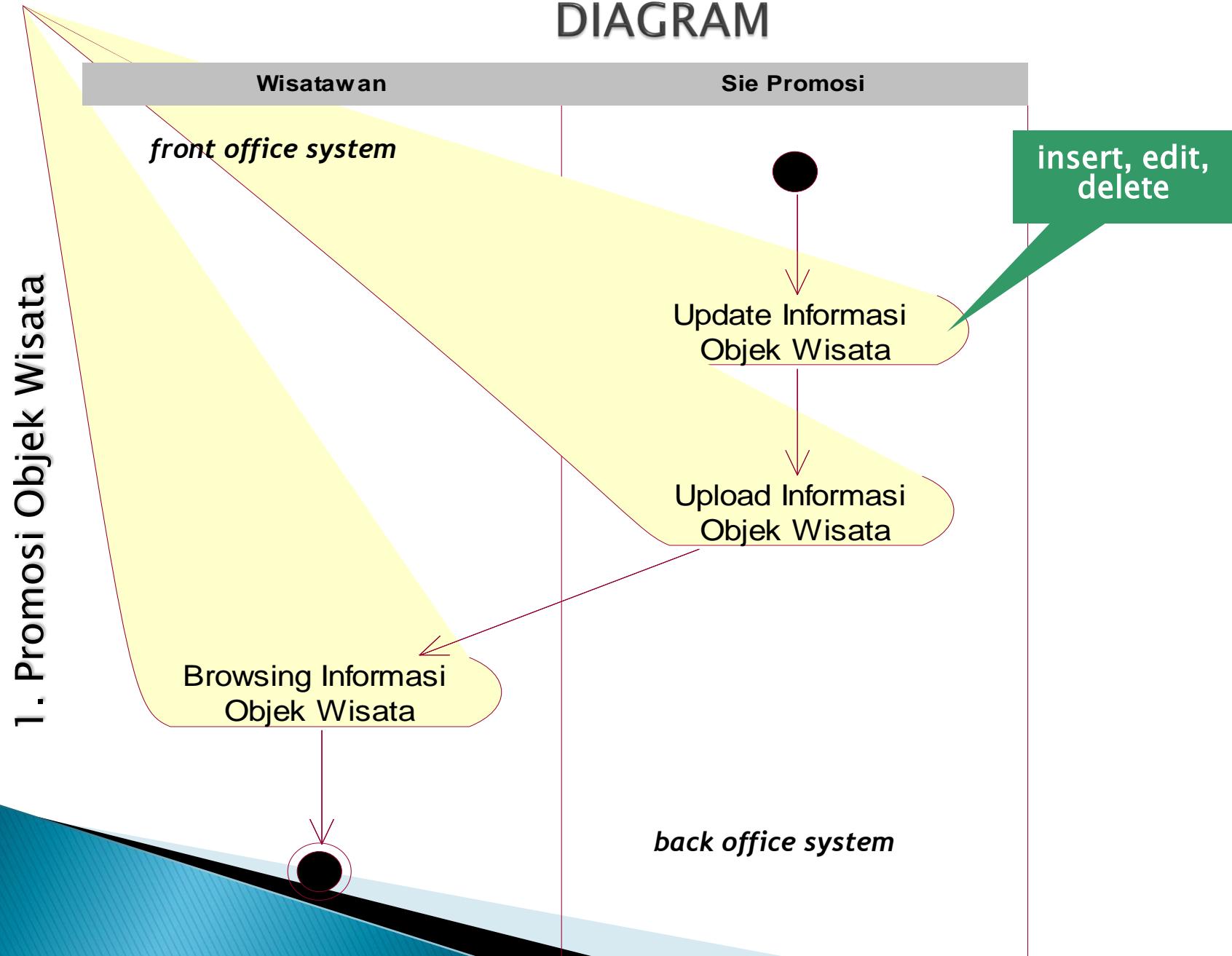


PEMODELAN PERSOALAN – BUSINESS OBJECT MODEL

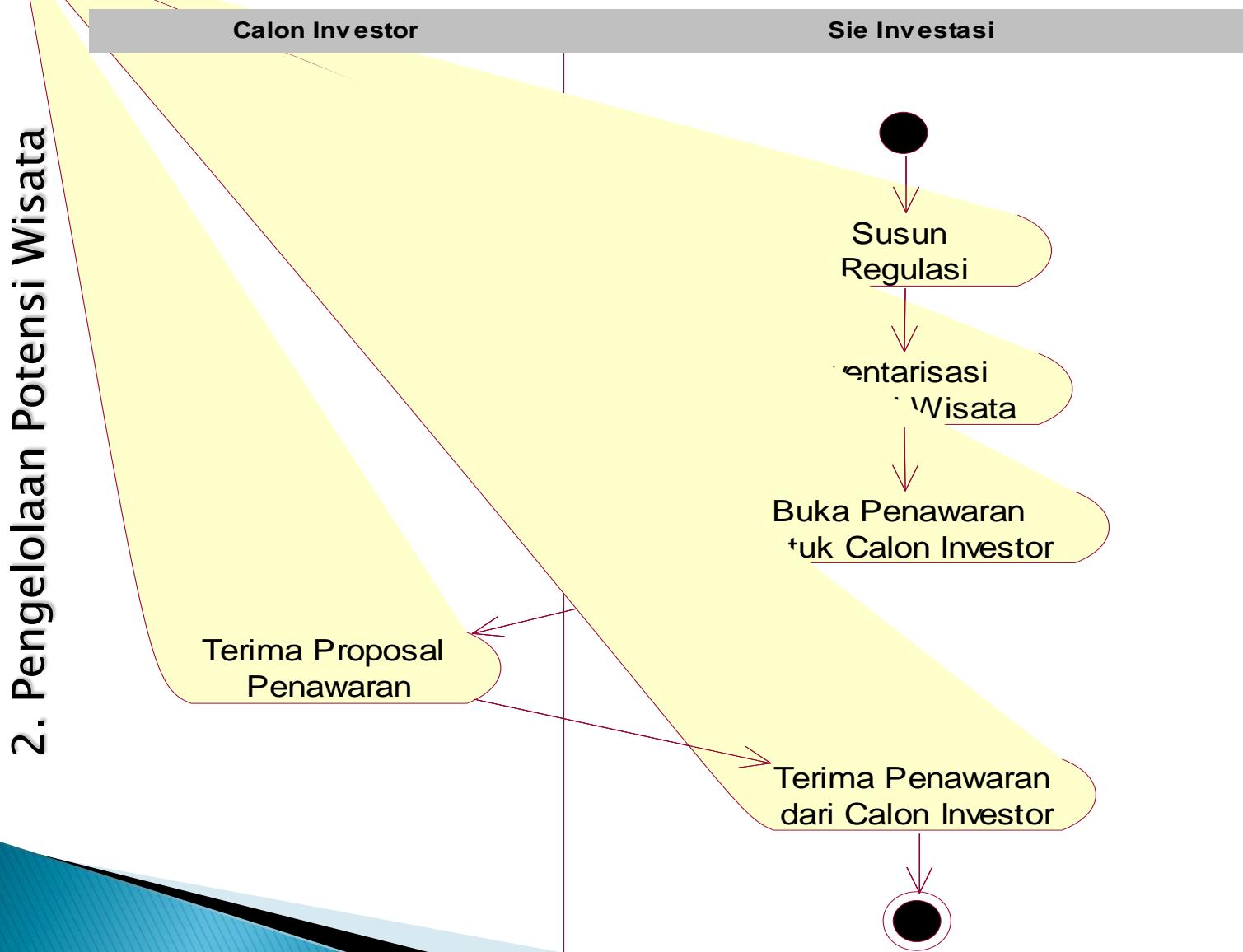
2. Pengelolaan Potensi Wisata



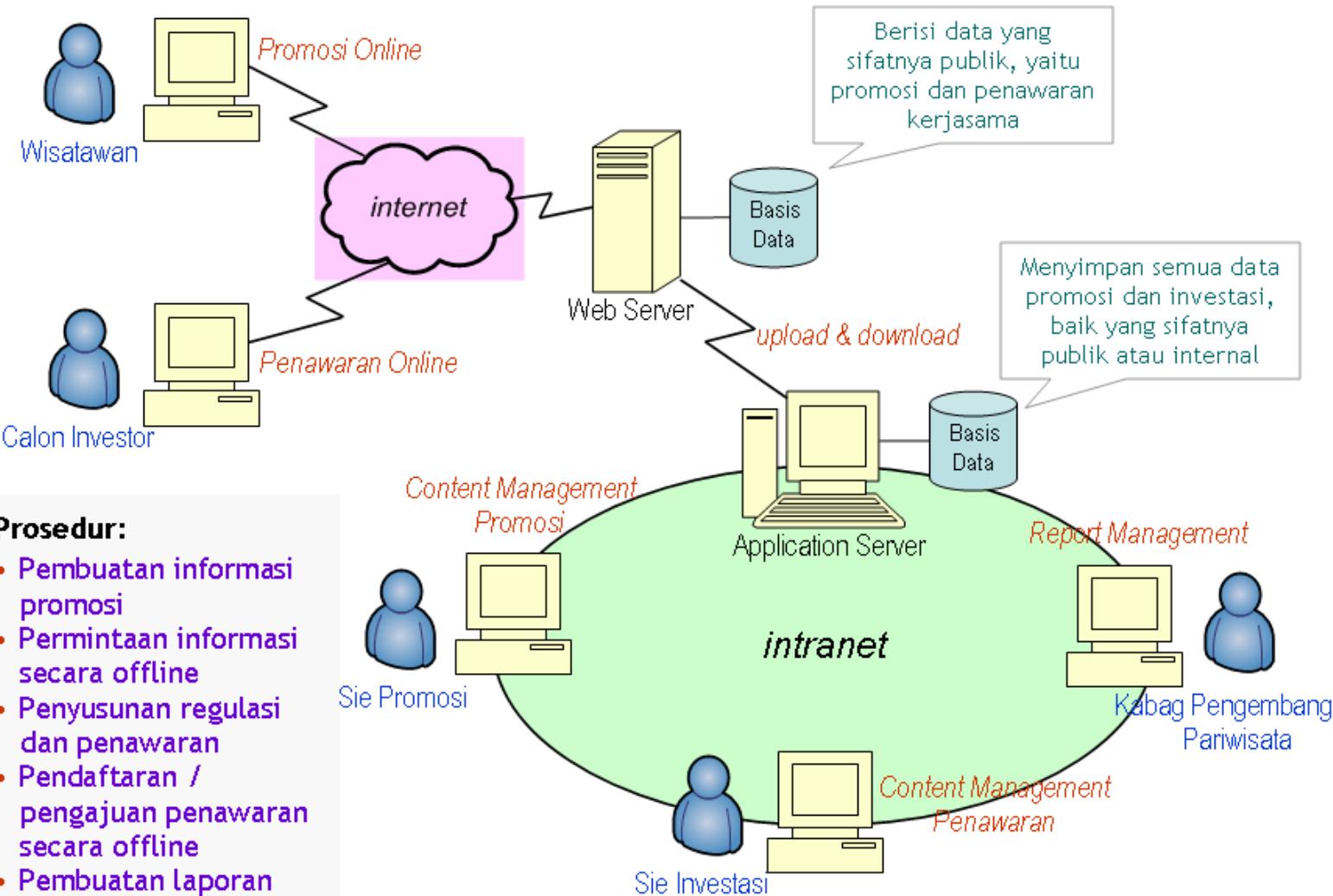
PEMODELAN PERSOALAN – BUSINESS ACTIVITY DIAGRAM



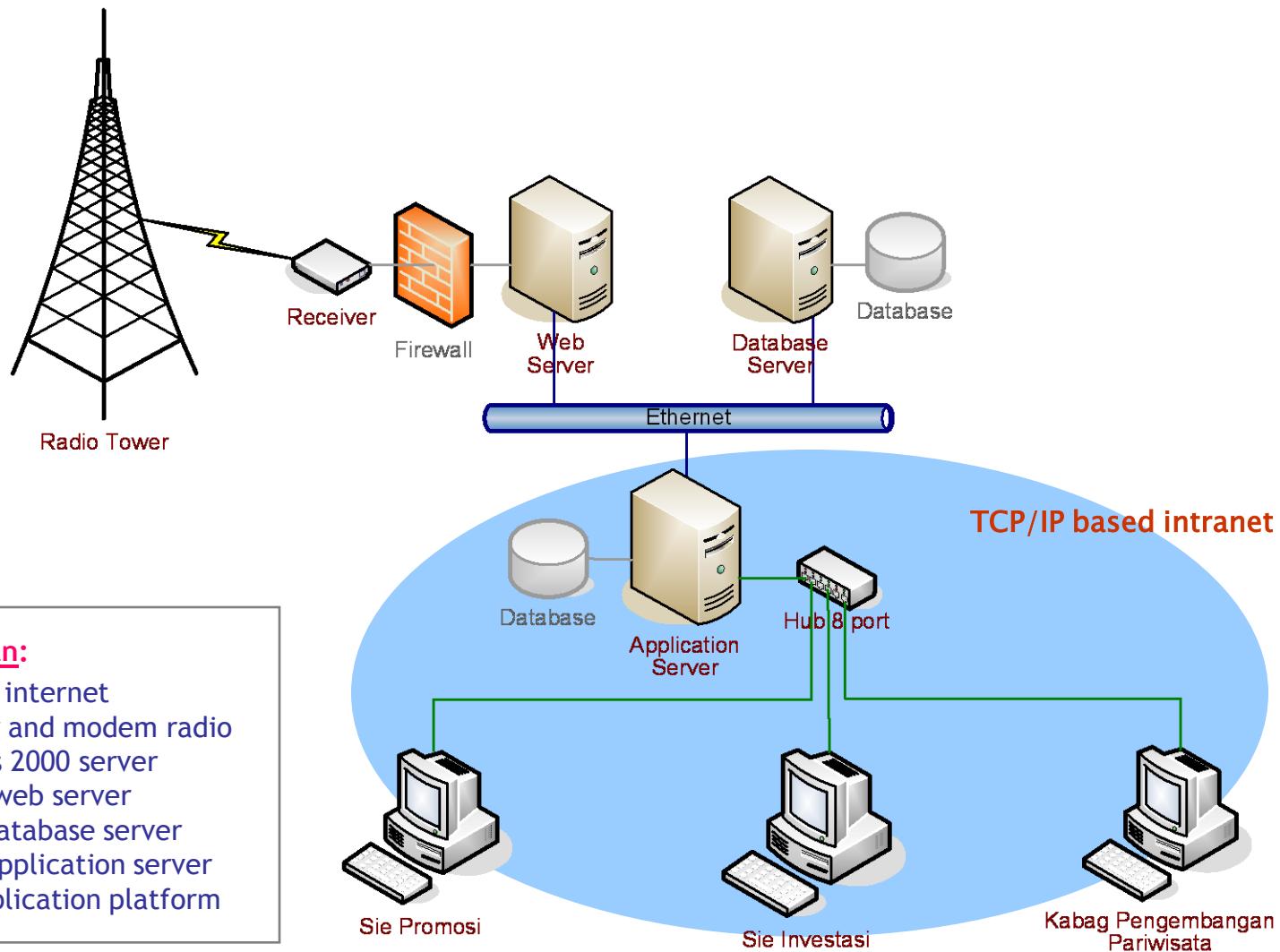
PEMODELAN PERSOALAN – BUSINESS ACTIVITY DIAGRAM



PENENTUAN SOLUSI (USER VIEW)



PENENTUAN SOLUSI (PLATFORM)



Contoh: Kebutuhan Perangkat Lunak Aplikasi

No.	Nama Aplikasi	Peruntukan	Manfaat / Kegunaan
1	Promosi Online	Wisatawan	Melihat informasi objek wisata dan fasilitasnya
2	Penawaran Online	Calon Investor	<ul style="list-style-type: none">• Melihat proposal penawaran pengelolaan potensi wisata• Mendaftar/mengajukan penawaran
3	Content Management Promosi	Sie Promosi	<ul style="list-style-type: none">• Mengelola data objek wisata dan fasilitasnya• Mengelola data jadwal pertunjukan• Retrieval informasi objek wisata
4	Content Management Investasi	Sie Investasi	<ul style="list-style-type: none">• Mengelola data potensi wisata• Mengelola data regulasi• Mencatat penawaran dari investor
5	Report Management	Kabag Pengembangan Pariwisata	<ul style="list-style-type: none">• Menyajikan laporan mengenai:<ul style="list-style-type: none">- Promosi objek wisata- Penawaran pengelolaan potensi wisata

UNTUK DIINGAT LAGI

- ▶ Perangkat lunak digunakan oleh pemakai di **lingkungan operasional** dengan **teknologi** tertentu sebagai **bagian dari sistem** yang lebih besar.
- ▶ Oleh karena itu, hal kedua yang harus dilakukan saat membuat perangkat lunak adalah **memahami lingkungan operasional, teknologi** dan **sistem** yang menjadi tempat dimana perangkat lunak digunakan.

BAGAIMANA UNTUK MASALAH INI?

Masalah	Sistem	Kebutuhan PL
<ul style="list-style-type: none">• Pengaturan distribusi gas dari station pengisian gas ke rumah-rumah konsumen.• Proses monitoring untuk mengantisipasi kemungkinan terjadinya kebocoran.	<ul style="list-style-type: none">• Elemen sistem:<ul style="list-style-type: none">▪ <i>sensor</i>▪ <i>embedded software</i>▪ perangkat komunikasi▪ operation center• Arsitektur logis sistem:<ul style="list-style-type: none">▪ susunan elemen sistem▪ keterkaitan antar elemen▪ mekanisme kerja	<p>Kebutuhan:</p> <ul style="list-style-type: none">• fungsional• non-fungsional <p><i>embedded software</i></p>

