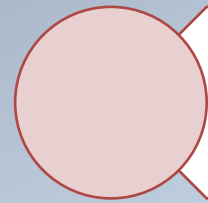




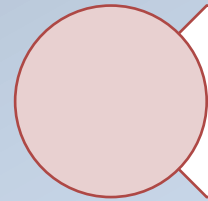
# Analisis dan Pemodelan Perangkat Lunak

Kuliah Terintegrasi Blok RPL  
Pertemuan 1

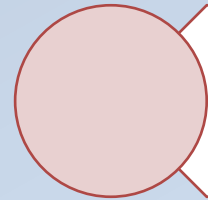
# Sub Topik



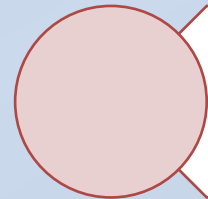
**1. *Software and Software Engineering***



**2. *Software Process***



**3. *User Requirement***



**4. *Use Case***

# 1. Software and Software Engineering



# Software (1)

- Merupakan program komputer dan dokumentasinya, seperti : *requirement*, *design model*, dan *user manual*.
- Produk *software* dikembangkan untuk *customer* tertentu atau untuk pangsa pasar tertentu.
- Dapat berupa :
  - Generik → Dikembangkan untuk dijual ke kelompok / pasar tertentu. Contoh : Ms. Office.
  - Custom → Dikembangkan untuk *customer* tunggal sesuai dengan spesifikasi yang diminta.

## Software (2)

- *Software* baru dapat dikembangkan dengan cara membuat **program yang baru, mengkonfigurasi *generic software*** atau **menggunakan kembali *software*** yang ada.

# Peran *Software*

- Banyak aktivitas manusia saat ini tergantung pada *software*.
- Terutama untuk membantu kegiatan operasional sehari-hari.
- Sehingga banyak perusahaan menempatkan pembelanjaan *software* sebagai hal yang penting.
- Hal ini mempengaruhi GNP (*Gross National Product*) suatu negara.

# Attributes of Good Software

1. Maintainability
2. Reliability
3. Security and safety
4. Efficiency
5. Acceptability



# Rekayasa Perangkat Lunak

- Merupakan suatu disiplin/ilmu rekayasa yang berhubungan dengan seluruh aspek produksi *software*.
- *Software engineer* harus mengadaptasi sebuah pendekatan yang sistematis dan terorganisasi pada pekerjaannya dan menggunakan *tool* dan teknik yang cukup berdasarkan :
  - Masalah yang diatasi
  - *Development constraints* (Batasan dalam pengembangan)
  - Sumber daya yang tersedia



# Pentingnya Rekayasa Perangkat Lunak

- Software harus **reliable**, **aman**, **usable**, dan **maintainable**.
- Secara eksplisit, terfokus pada menghasilkan *software* dengan atribut di atas.
- Penting bagi sistem yang digunakan oleh banyak orang dan bisnis tertentu, dan yang digunakan selama bertahun-tahun.

# Kegagalan Proyek Perangkat Lunak

- Peningkatan **kompleksitas** *software*.
  - Bermula dari sistem yang sederhana, kemudian ditambahkan banyak fitur lainnya sehingga **menjadi kompleks dan sulit dipelihara**.
- **Kegagalan** untuk **menggunakan metode** *software engineering*.
  - Banyak perusahaan **tidak menggunakan metode dan teknis software engineering** dalam aktivitas pengembangan *software* sehari-hari

# Biaya Rekayasa Perangkat Lunak

- Umumnya 60% merupakan biaya pengembangan dan 40% merupakan biaya pengujian.
- Untuk *custom software*, biaya perubahan / evolusi *software* melebihi biaya pengembangan *software*.

# Tantangan Utama *Rekayasa Perangkat Lunak*

- ***Heterogenity (keberagaman)***
  - Membangun teknik untuk membuat *software* dapat berjalan pada *platform* dan lingkungan eksekusi yang heterogen.
- ***Delivery (penyelesaian)***
  - Membangun teknik untuk mengarahkan pada *software delivery* yang lebih cepat.
- ***Trust (Kepercayaan)***
  - Membangun teknik yang mendemonstrasikan bahwa *software* dapat dipercaya oleh penggunanya.

## 2. Software Process



# Aktifitas Proses Perangkat Lunak

- Software specification, di mana *customer* dan *engineer* mendefinisikan *software* yang akan dihasilkan dan batasannya.
- Software development, di mana *software* dirancang dan diprogramkan.
- Software validation, di mana *software* dicek untuk menjamin kesesuaian dengan yang dibutuhkan oleh *user*.
- Software evolution, di mana *software* dimodifikasi untuk mencerminkan perubahan kebutuhan *customer* dan pasar.

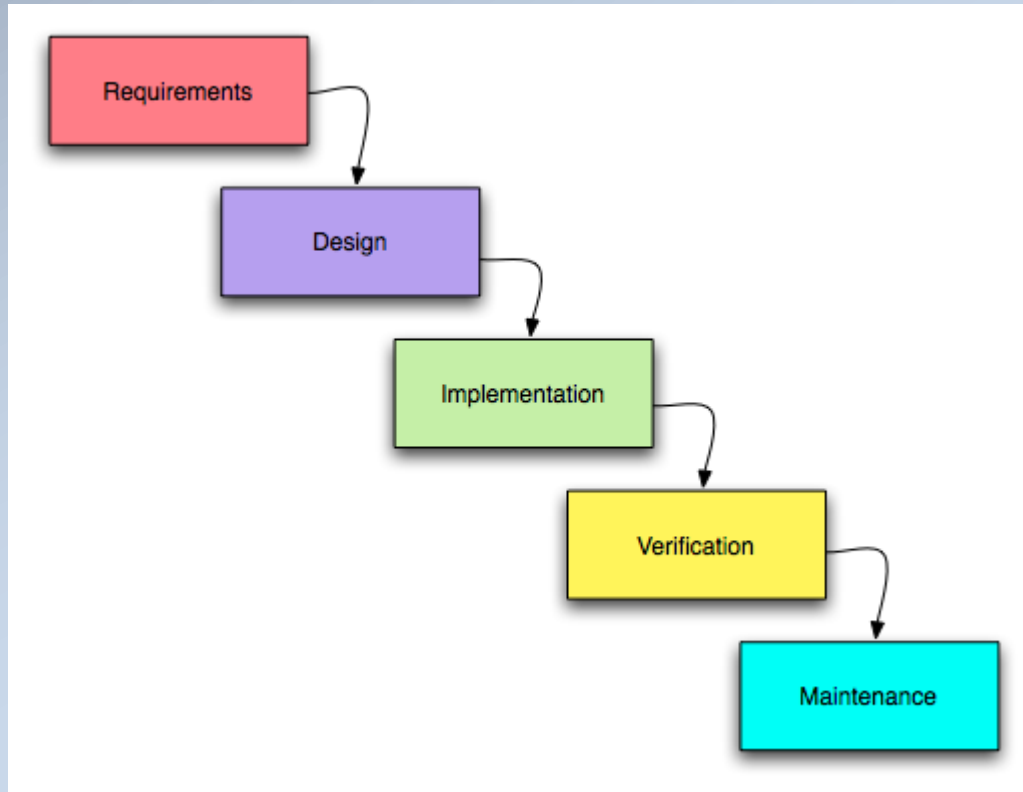
# Model Proses Perangkat Lunak

- Merupakan sebuah representasi sederhana dari sebuah *software process*, yang dipresentasikan dari sebuah perspektif tertentu.
- Contoh *process perspective* :
  - *Workflow perspective* – urutan aktivitas.
  - *Data flow perspective* – alur informasi.
  - *Role/action perspective* – siapa mengerjakan apa

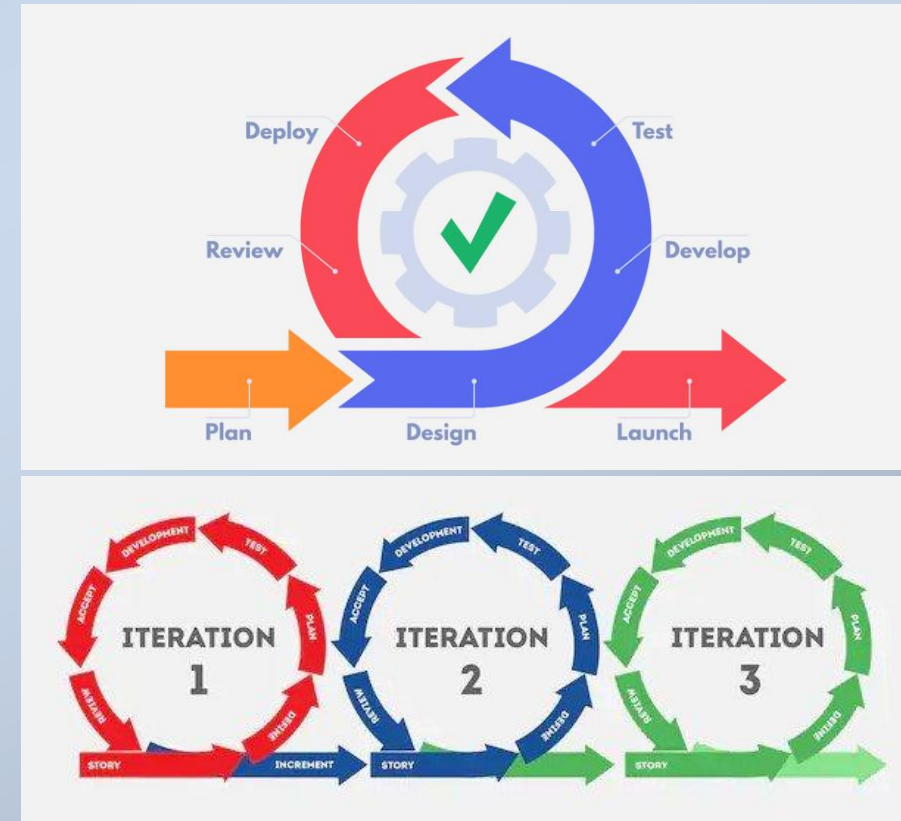


# Generic Process Model

- Water Fall

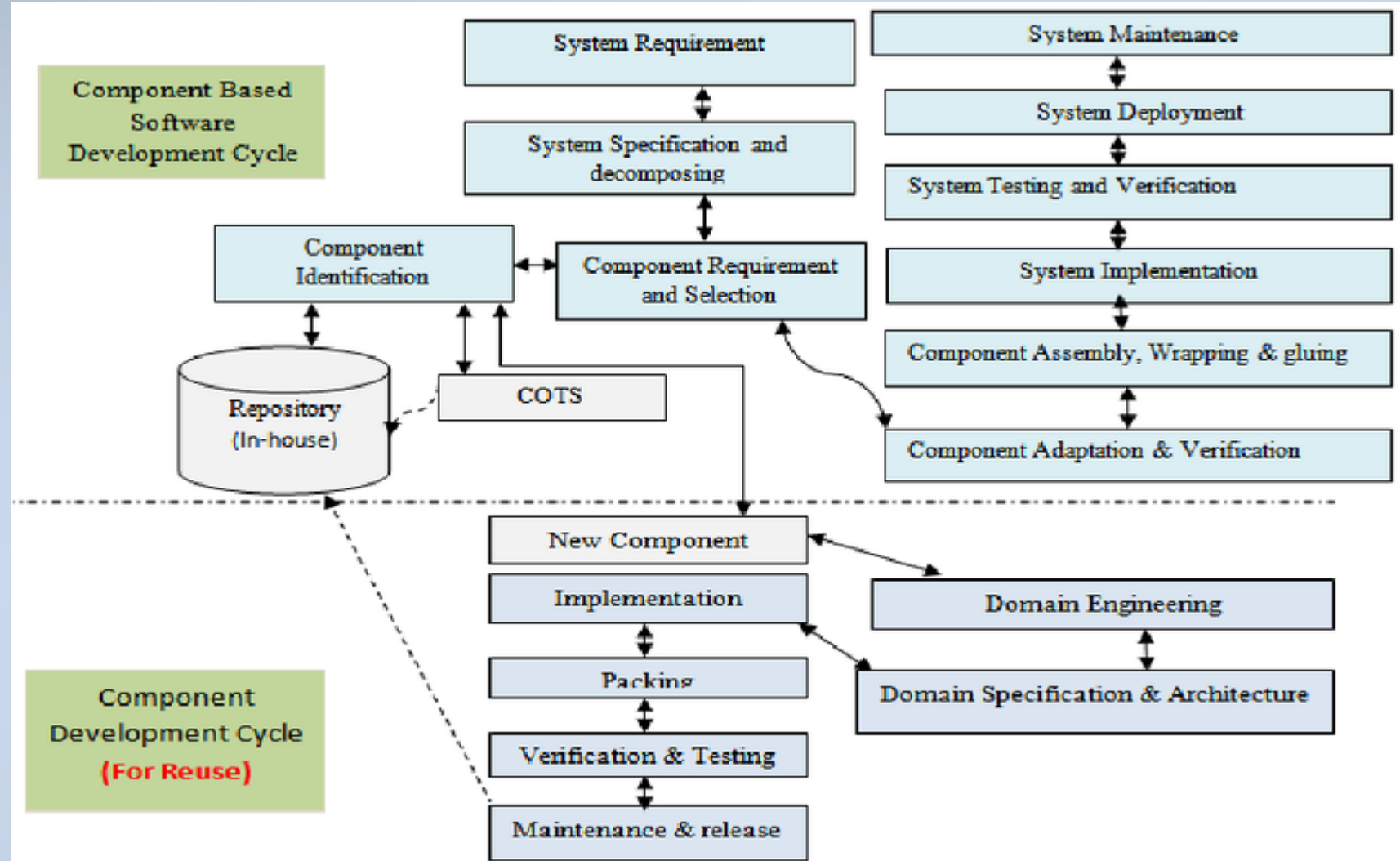


- Iterative development



# Generic Process Model

- Component-based software engineering



### 3. User Requirement



# User Requirement

- Proses membangun layanan yang diinginkan *customer* dari sebuah sistem dan batasan yang dapat dioperasikan dan dikembangkan.
- Kebutuhan perangkat lunak merupakan deskripsi dari layanan dan batasan sistem yang terbentuk selama proses rekayasa kebutuhan perangkat lunak.
- Metode yang dilaksanakan secara umum adalah kuesioner dan wawancara



# Kebutuhan Fungsional (1)

- Mendeskripsikan fungsionalitas / kegunaan dari suatu sistem.
- Tergantung dari jenis software, user, dan jenis sistem di mana software tersebut digunakan.
- Functional user requirements dapat berupa high-level statement dari apa yang perlu sistem lakukan.
- Functional system requirements harus mendeskripsikan layanan sistem secara detail.

# Kebutuhan Fungsional (2)

Harus mencantumkan :

- Deskripsi fungsional atau entitas yang terlibat.
- Deskripsi input dan asalnya.
- Deskripsi output dan akan digunakan untuk apa.
- Informasi yang dibutuhkan untuk komputasi atau entitas lainnya pada sistem.
- Deskripsi *action* yang dijalankan.
- Terdapat pre-condition yang memicu (bernilai TRUE) *action* dijalankan, dan post-condition yang akan memicu proses / *action* lain dijalankan.
- Deskripsi untuk efek samping dari operasi.



# Contoh Dokumen

	<i>Name</i>	<i>Role</i>	<i>Interest</i>
1	Super Users	Authorized PMP and Appriss staff responsible for administering, supporting, and maintaining the PMP software.	Uses the Web Portal to configure and troubleshoot PMP software behavior.
2	PMP Administrators	PMP staff responsible for reporting on and administering the data contained in the PMP software.	Uses the Web Portal to perform day-to-day reporting and administrative work necessary in order to facilitate the running of a state's PMP. This includes full access to the data and functions the Requestors and Data Submitters use.
3	Requestors	Authorized staff responsible for requesting data from the PMP software.	Uses the Web Portal to request and view reports on data that has been processed by the PMP software.
4	Data Submitters	Dispenser staff or their representatives responsible for submitting prescription data to the PMP.	Uses the Web Portal to submit data, correct errors, and track the overall status of the pharmacy's PMP software reporting.
5	System Users	Specialized accounts used only by computer systems to communicate with the PMP software.	None. There are no human users for these accounts.

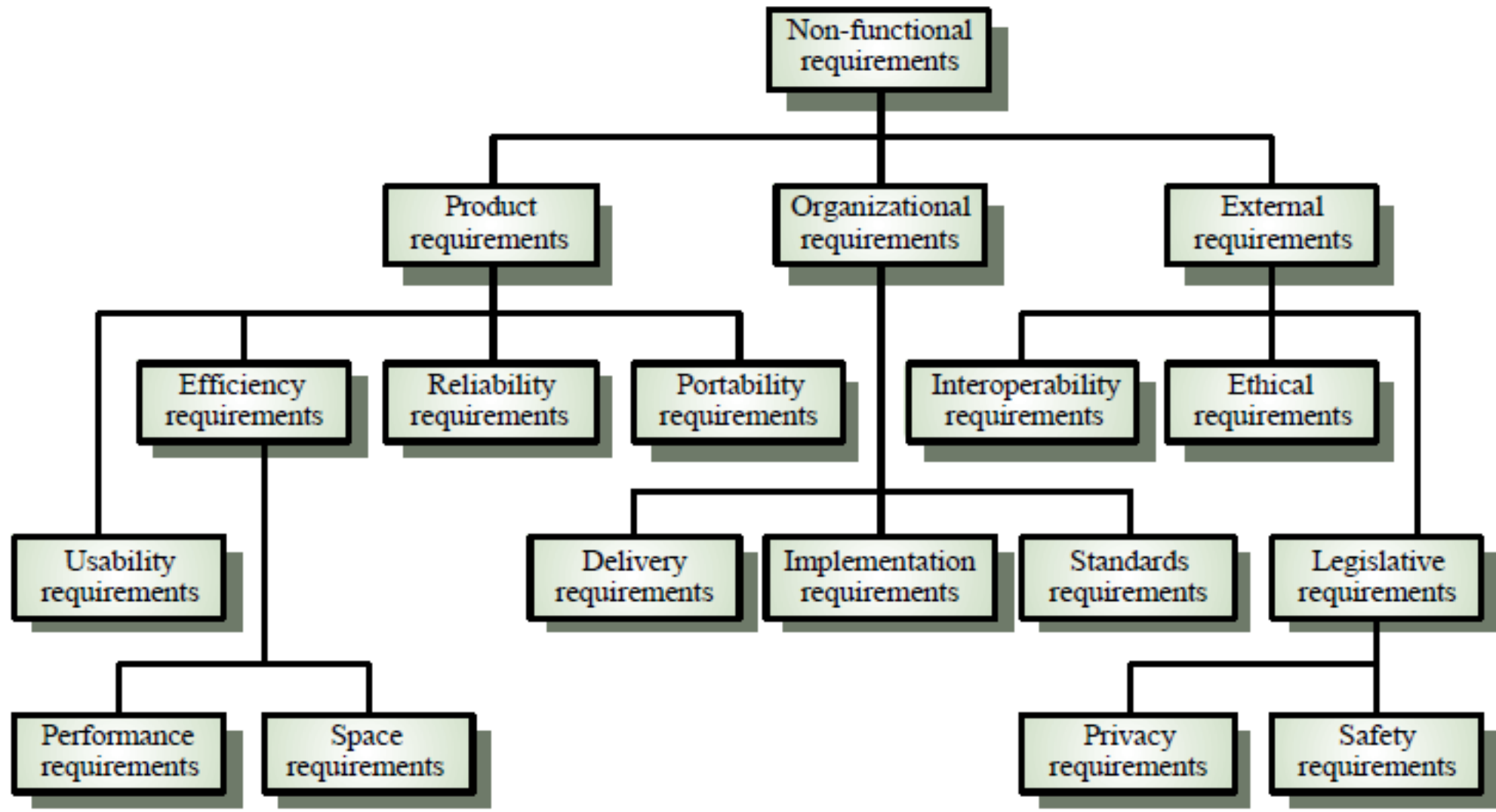
## PMP NextGen

Functional Requirements

[Contoh dokumen  
requirement  
Stanford University](#)

# Kebutuhan Non Fungsional

- Mendeskripsikan keterangan dan batasan sistem yaitu reliability, response time, storage requirements. Batasan berupa I/O device capability, representasi sistem, dll.
- Kebutuhan proses dapat diterjemahkan melalui sistem CASE tertentu, bahasa pemrograman, atau metode pengembangan tertentu.
- Dapat lebih penting daripada functional requirements.







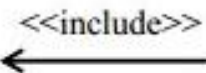
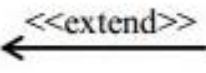
## 4. Use Case



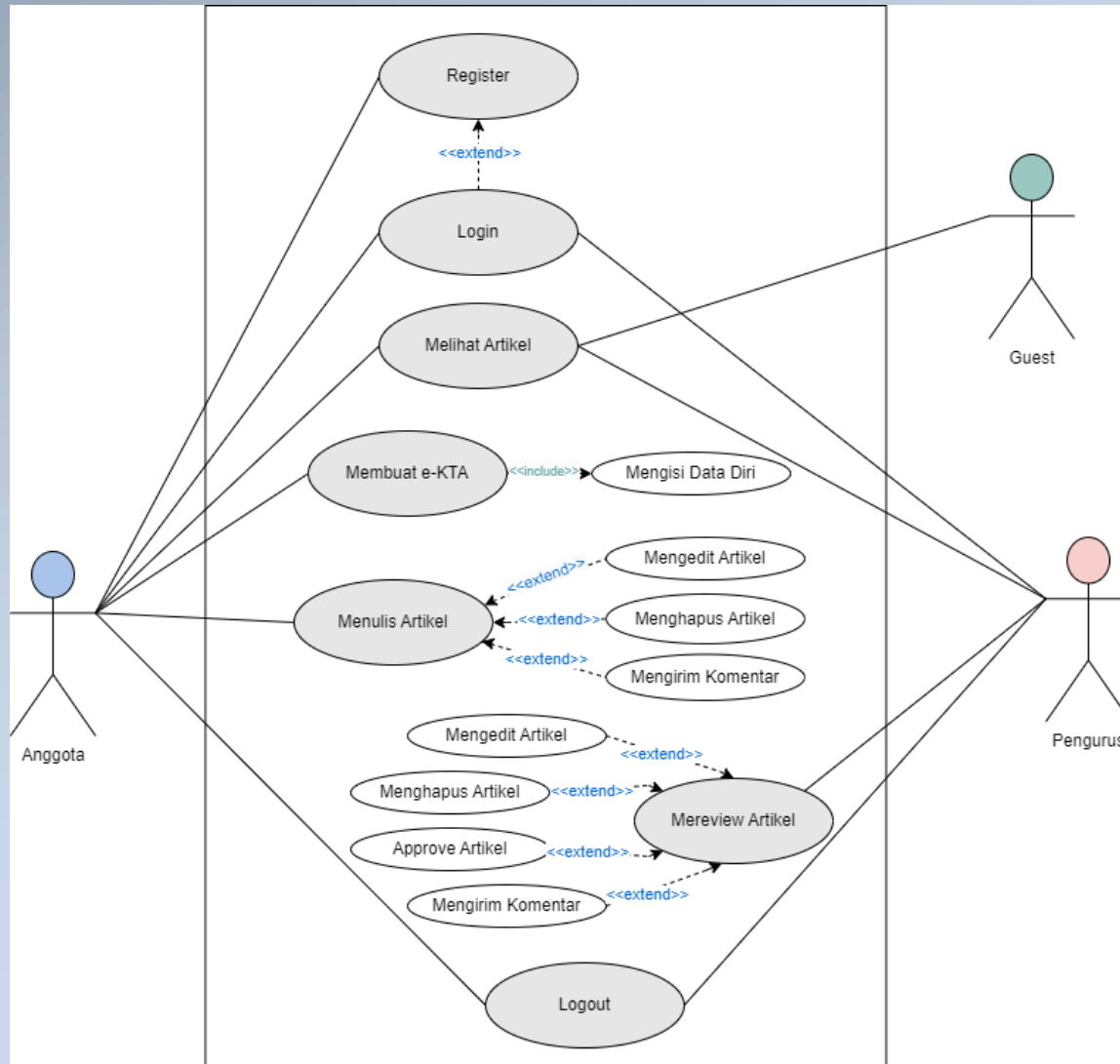
# Use Case

- Definisi : Use Case menjelaskan interaksi dari *user*(pengguna) dengan produk/sistem

# Simbol Use Case

Simbol	Keterangan
	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i>
	<i>Use case</i> : Abstraksi dan interaksi antara sistem dan aktor
	<i>Association</i> : Abstraksi dari penghubung antara aktor dengan use case
	<i>Generalisasi</i> : Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan use case
	Menunjukkan bahwa suatu use case seluruhnya merupakan fungsionalitas dari use case lainnya
	Menunjukkan bahwa suatu use case merupakan tambahan fungsional dari use case lainnya jika suatu kondisi terpenuhi

# Integrasi Penelitian



Judul Penelitian :

**PENGEMBANGAN APLIKASI  
GROUPWARE UNTUK  
MENDUKUNG OPERASIONAL  
KOMUNITAS VIRTUAL GEOMUDA  
INDONESIA**

Author :

Aisyah Azahra Al Amanah  
Ahmad Zuhdi  
Ratna Shofiati