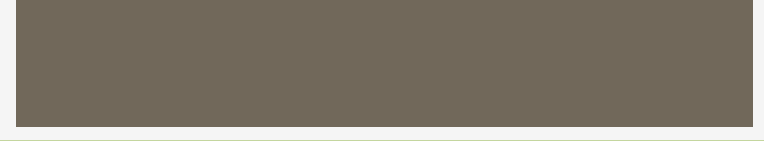


# Rekayasa Perangkat Lunak

## Rekayasa Kebutuhan



**Teknik Informatika**  
**UNIKOM**



# Rekayasa Kebutuhan

1. Kenapa butuh rekayasa kebutuhan?
2. Definisi kebutuhan dan rekayasa kebutuhan
3. Cara mendapatkan kebutuhan
4. Tipe-tipe kebutuhan
5. Kebutuhan fungsional dan non fungsional
6. Requirement Measures
7. Dokumen kebutuhan
8. Petunjuk penulisan kebutuhan
9. Pengguna dari dokumen kebutuhan

# Kenapa Butuh Rekayasa Kebutuhan?

Apa yang customer inginkan dari software ini?

"Mau dibawa kemana" software ini???



# Kenapa Butuh Rekayasa Kebutuhan?



What the heck is this? The site isn't anything like I thought it would be. You couldn't have taken a little more time and gotten it right? It's like you didn't even know what I wanted...

## If your customer isn't happy, you built the wrong software.

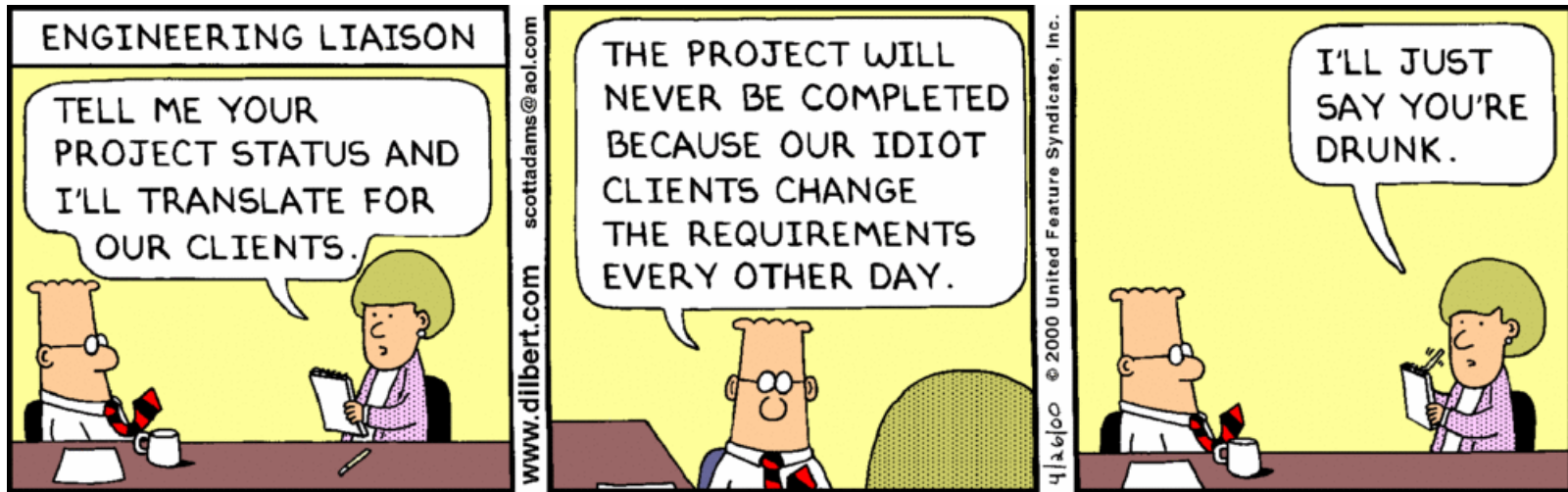
Big bang software usually means working a whole lot, but it also means not showing the customer much until your work is done. The risk with that approach is you **think** you're building what the customer wants with no real feedback until you **think** you're finished.

And, no matter how great YOU think your software is, it's the customer you have to make happy. So if the customer doesn't like what you've built, don't waste time trying to tell them they're wrong. Just get ready to do some rework.

But how do you figure out what the customer really wants? It's not always easy...

The emphasis here is that you think you're finished... but you may not be.

# Kenapa Butuh Rekayasa Kebutuhan?



# Definisi Rekayasa Kebutuhan

Proses pembentukan layanan-layanan yang customer butuhkan dari sebuah sistem dan batasan-batasannya dimana ia beroperasi dan dikembangkan.

# Definisi Kebutuhan

1. Deskripsi dari layanan sistem dan batasan-batasannya yang dihasilkan selama proses rekayasa kebutuhan.
2. Kebutuhan berkisar antara pernyataan abstrak tingkat tinggi dari layanan atau dari batasan sistem hingga spesifikasi fungsional matematis.



# Cara Mendapatkan Kebutuhan

1. Interviews
2. Questionnaires
3. Observation
4. Searching





# Cara Mendapatkan Kebutuhan





**Watch it!**

**Avoid office politics.**

*Nothing will stifle creative bluesky thinking like a boss that won't let people speak up. Try as much as possible to leave job descriptions and other baggage at the door when blueskying requirements. Everyone should get an equal say to ensure you get the most out of each brainstorming session.*

# Tipe-Tipe Kebutuhan

## 1. **User requirements**

Statements in natural language plus diagrams of the services the system provides and its operational constraints. Written for customers

## 2. **System requirements**

A structured document setting out detailed descriptions of the system services. Written as a contract between client and contractor

## 3. **Software specification**

A detailed software description which can serve as a basis for a design or implementation. Written for developers

# Tipe-Tipe Kebutuhan

## User Requirement Definition/Requirement Definition

1. The software must provide a means of representing and accessing external files created by other tools.

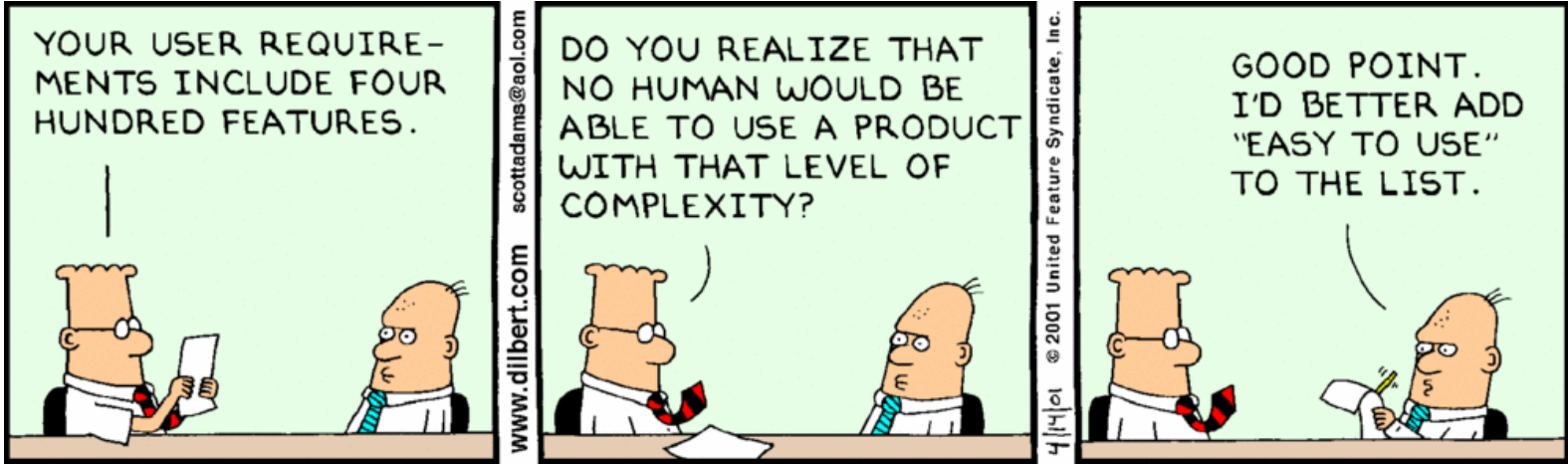
## System Requirement Specification/Requirement Specification

- 1.1 The user should be provided with facilities to define the type of external files.
- 1.2 Each external file type may have an associated tool which may be applied to the file.
- 1.3 Each external file type may be represented as a specific icon on the user's display.
- 1.4 Facilities should be provided for the icon representing an external file type to be defined by the user.
- 1.5 When a user selects an icon representing an external file, the effect of that selection is to apply the tool associated with the type of the external file to the file represented by the selected icon.

# User Requirement

1. Menjelaskan **kebutuhan fungsional** dan **non-fungsional** dalam berbagai cara sehingga mereka dimengerti oleh pengguna sistem yang tidak mempunyai pengetahuan teknik yang detail.
2. Kebutuhan user didefinisikan dalam bahasa natural, tabel-tabel, dan diagram-diagram yang dimengerti oleh pengguna.

# User Requirement



www.dilbert.com scottadams@aol.com

4/14/01 © 2001 United Feature Syndicate, Inc.

# System Requirement

1. Spesifikasi yang lebih detail dari fungsi-fungsi sistem, layanan-layanan, dan batasan-batasan jika dibandingkan dengan user requirements.
2. Dibuat sebagai dasar untuk mendesain sistem.
3. Didefinisikan dan diilustrasikan menggunakan model sistem.



# Kebutuhan Fungsional

1. Pernyataan dari layanan sistem yang **harus disediakan**, bagaimana sistem harus bereaksi terhadap input tertentu dan bagaimana sistem harus berperilaku dalam situasi tertentu.
2. Menjelaskan fungsionalitas dari sistem.
3. Menjelaskan layanan dari sistem secara detail

# Contoh Kebutuhan Fungsional

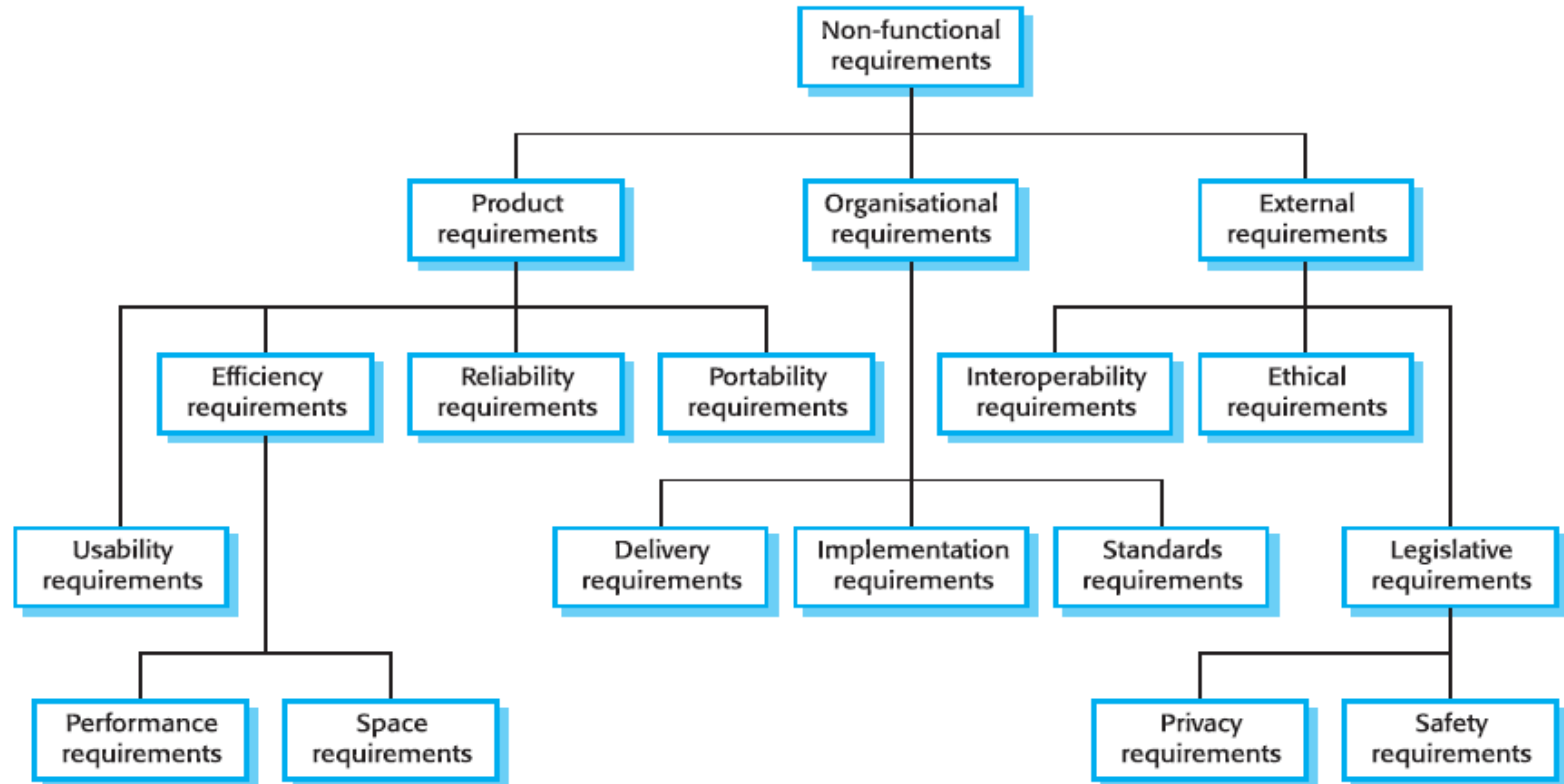
1. The user shall be able to search either all of the initial set of databases or select a subset from it.
2. The system shall provide appropriate viewers for the user to read documents in the document store.
3. Every order shall be allocated a unique identifier (ORDER\_ID) which the user shall be able to copy to the account's permanent storage area

# Kebutuhan Non-Fungsional

Batasan-batasan dari layanan-layanan dan fungsi-fungsi dari sebuah sistem, seperti:

1. Batasan waktu
2. Batasan dari pengembangan proses
3. Batasan pengguna.

# Kebutuhan Non-Fungsional



# Contoh Kebutuhan Non-Fungsional

## 1. Product requirement

The user interface for LIBSYS shall be implemented as simple HTML without frames or Java applets.

## 2. Organisational requirement

The system development process and deliverable documents shall conform to the process and deliverables defined in XYZCo-SP-STAN-95.

## 3. External requirement

The system shall not disclose any personal information about customers apart from their name and reference number to the operators of the system.

# Requirement Measures

| <b>Property</b> | <b>Measure</b>   |
|-----------------|--|
| Speed           | Processed transactions/second<br>User/Event response time<br>Screen refresh time                                   |
| Size            | K Bytes<br>Number of RAM chips   |
| Ease of use     | Training time<br>Number of help frames   |
| Reliability     | Mean time to failure<br>Probability of unavailability<br>Rate of failure occurrence<br>Availability                |
| Robustness      | Time to restart after failure<br>Percentage of events causing failure<br>Probability of data corruption on failure |
| Portability     | Percentage of target dependent statements<br>Number of target systems  |

# Dokumen Kebutuhan

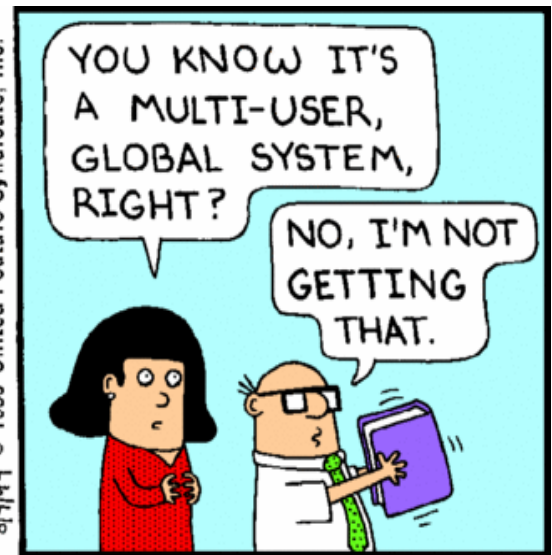
1. Dokumen kebutuhan adalah pernyataan yang resmi dari apa yang dibutuhkan oleh developer sistem.
2. Menggabungkan antara definisi dan spesifikasi kebutuhan.
3. Dokumen kebutuhan bukan dokumen desain.  
Dokumen kebutuhan lebih berisi apa yang sebuah sistem bisa lakukan dibandingkan bagaimana cara sistem melakukannya.

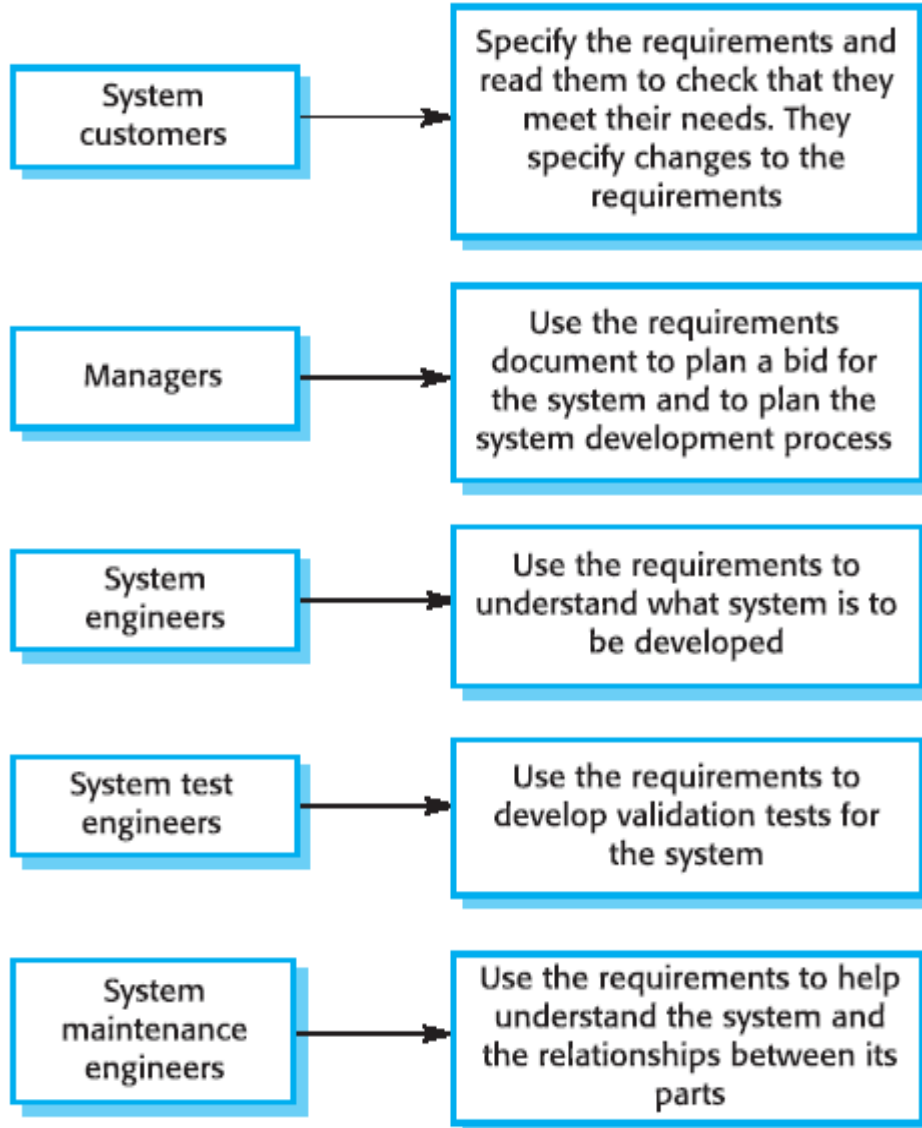


# Petunjuk Penulisan Kebutuhan

1. Menggunakan format standar untuk semua kebutuhan.
2. Menggunakan bahasa yang konsisten.
3. Menandai bagian-bagian penting dari seluruh kebutuhan.
4. Jangan menggunakan bahasa jargon.
5. Complete but not Complicated

# Dokumen Kebutuhan





# Pengguna Dokumen Kebutuhan

**SELESAI...**