

## Pembahasan Responsi Analisis Perancangan Sistem dan Informasi (APSI) 2020

### Pilihan Ganda

#### 1. B. Component, Boundary, Environment, Input, Process, Output

##### Pembahasan :

Suatu sistem memiliki beberapa karakteristik, yaitu:

- Komponen/ Elemen (*Component*)  
Suatu sistem terdiri dari komponen yang saling berinteraksi, artinya saling bekerjasama membentuk satu kesatuan. Kumpulan dari beberapa komponen biasa dikenal dengan sub sistem.
- Batas sistem (*Boundary*)  
Batas sistem merupakan daerah yang membatasi antara sistem yang satu dengan sistem lainnya atau dengan lingkungan luarnya. Batasan sistem ini merupakan ruang lingkup atau *scope* dari sistem/ subsistem itu sendiri.
- Lingkungan luar sistem (*Environment*)  
Segala sesuatu diluar dari batas sistem yang memengaruhi operasi dari suatu sistem disebut lingkungan luar sistem (*environment*). Lingkungan luar sistem bersifat menguntungkan atau merugikan, jika menguntungkan harus dipelihara dan dijaga agar pengaruhnya tidak hilang, namun apabila merugikan maka harus dikendalikan atau dimusnahkan agar tidak mengganggu operasi dari sistem.
- Masukan (*Input*)  
Masukan dalam sistem dapat berupa:
  - a. *Maintenance input* : suatu inputan yang digunakan agar sistem dapat beroperasi (contoh: program untuk mengoperasikan komputer)
  - b. *Signal input* : energi yang diproses untuk diperolehnya suatu keluaran (contoh: data)
- Pengolahan (*Process*)  
Suatu sistem mempunyai bagian pengolah yang akan mengubah input menjadi output (contoh: CPU pada komputer, bagian produksi yang mengubah bahan baku menjadi barang jadi)

- Keluaran (*Output*)

*Output* merupakan hasil dari energi yang telah diolah (contoh : informasi yang dikeluarkan oleh komputer)

## 2. C. Complex organizational procedure

### Pembahasan :

Berikut merupakan analisis dan desain sistem informasi, yaitu:

- Proses organisasi yang kompleks (*Complex organizational process*)
- Digunakan untuk mengembangkan dan memelihara sistem informasi berbasis komputer (*Use to develop and maintain computer-based information systems*)
- Digunakan oleh tim profesional bisnis dan sistem (*Used by a team of business and systems professionals*).

## 3. D. Strategy



## 4. A. Organizational role most responsible for analysis and design of information systems

## 5. B. Standard process followed in an organization to conduct all the steps necessary to analyze, design, implement, and maintain information systems?

## 6. B. Information System

## 7. C. Planning – Analysis – Design – Development – Testing – Implementation – Operation & Maintenance

### **Pembahasan:**

#### **1. Perencanaan**

Membuat semua rencana yang berkaitan dengan proyek sistem informasi. Seperti layaknya kita ingin membangun rumah maka kita akan melakukan perencanaan bagaimana pondasinya, bagaimana struktur bangunannya, material nya apa, warnanya, dan budget yang harus kita keluarkan.

#### **2. Analisa**

Menganalisa workflow sistem informasi yang sedang berjalan dan mengidentifikasi apakah workflow telah efisien dan sesuai dengan standar. Selama tahap analisis, sistem analis terus bekerjasama dengan manajer, dan komite pengarah SIM terlibat dalam titik-titik yang penting mencakup kegiatan sebagai berikut:

- Menetapkan rencana penelitian system
- Mengorganisasikan tim proyek
- Mendefinisikan kebutuhan informasi
- Mendefinisikan kriteria kinerja system
- Menyiapkan usulan rancangan sistem
- Menyetujui atau menolak rancangan proyek pengembangan sistem.

#### **3. Desain**

Langkah yang sangat penting karena langkah ini menentukan fondasi sistem informasi. kesalahan dalam mendesain akan menimbulkan hambatan bahkan kegagalan proyek yang akan kita buat.

Langkah-langkah tahap rancangan sistem mencakup:

- Menyiapkan detail rancangan system
- Mengidentifikasi berbagai alternatif konfigurasi/rancang banun system
- Mengevaluasi berbagai alternatif konfigurasi system
- Memilih konfigurasi terbaik
- Menyiapkan usulan penerapan/aplikasi

- Menyetujui atau menolak aplikasi system

#### **4. Pengembangan**

Pekerjaan yang dilakukan di tahap pengembangan ini adalah pemrograman.

Seorang yang mengembangkan ini adalah seorang programmer yang membuat dengan Bahasa pemrograman tertentu

#### **5. Testing**

Proses yang dibuat sedemikian rupa dalam mengidentifikasi ketidaksesuaian dari hasil sebuah sistem informasi dengan hasil yang kita harapkan.

#### **6. Implementasi**

Proses untuk menerapkan suatu sistem informasi yang telah kita buat agar pengguna dapat menggunakannya menggantikan sistem informasi yang lama.

Proses Implementasi :

- Memberitahu user
- Melatih user
- Memasang system
- Entri / Konversi suatu data
- Siapkan user ID

#### **7. Pengoperasian dan Pemeliharaan**

Langkah terakhir adalah pengoperasian dan pemeliharaan. Selama sistem informasi telah beroperasi, terdapat beberapa pekerjaan rutin yang harus dilakukan terhadap sistem informasi yaitu:

- *System Maintenance*

System Maintenance adalah pemeliharaan sistem informasi, baik dari segi hardware maupun software.

- *Backup & Recovery*

Sistem informasi yang baik harus mempunyai perencanaan backup dan recovery.

- *Data Archive*

Data Archive adalah proses mengekstraksi data dari database dan menyimpannya di media lain seperti tape dan CD yang disebut data off-line . dan menghapusnya dari hard disk.

8. **C. SDLC**
9. **D. Each phase is organized into a number of separate iterations**
10. **B. Physical design**
11. **D. Development Tools**
12. **E. Management dan kontrol**
13. **B. Design System**
14. **C. Merancang *Technical Architecture***
15. **B. 3-1-2-5-7-4-6**
16. **B. Tidak memiliki persyaratan**

#### **Pembahasan (8 – 16)**

##### **Pengertian**

SDLC (system Development Life Cycle) adalah tahapan pekerjaan atau proses yang digunakan oleh analis sistem untuk mengembangkan dan membangun sistem informasi mulai dari penentuan kebutuhan, perancangan, validasi, pelatihan dan penyerahan pada konsumen.

**Berikut ini adalah Fase-fase Sistem Development Life Cycle (SDLC) meliputi:**

##### ***A. Perencanaan Sistem (Systems Planning)***

Lebih menekankan pada aspek studi kelayakan pengembangan sistem (feasibility study).

Aktivitas-aktivitas yang ada meliputi:

- Pembentukan dan konsolidasi tim pengembang.
- Mendefinisikan tujuan dan ruang lingkup pengembangan.
- Mengidentifikasi apakah masalah-masalah yang ada bisa diselesaikan melalui pengembangan sistem.
- Menentukan dan evaluasi strategi yang akan digunakan dalam pengembangan sistem.
- Penentuan prioritas teknologi dan pemilihan aplikasi.

##### ***B. Analisis Sistem (Systems Analysis)***

Analisa sistem adalah tahap di mana dilakukan beberapa aktivitas berikut:

- Melakukan studi literatur untuk menemukan suatu kasus yang bisa ditangani oleh sistem.
- Brainstorming dalam tim pengembang mengenai kasus mana yang paling tepat

dimodelkan dengan sistem.

- Mengklasifikasikan masalah, peluang, dan solusi yang mungkin diterapkan untuk kasus tersebut.
- Analisa kebutuhan pada sistem dan membuat batasan sistem.
- Mendefinisikan kebutuhan sistem

### ***C. Perancangan Sistem (Systems Design)***

Pada tahap ini, features dan operasi-operasi pada sistem dideskripsikan secara detail.

Aktivitas-aktivitas yang dilakukan adalah:

- Menganalisa interaksi obyek dan fungsi pada sistem.
- Menganalisa data dan membuat skema database.
- Merancang user interface.
- Merancang atau mendesain Technical Architecture.
- Mendesain model.

### ***D. Implementasi Sistem (Systems Implementation)***

Tahap berikutnya adalah implementasi yaitu mengimplementasikan rancangan dari tahap-tahap sebelumnya dan melakukan uji coba.

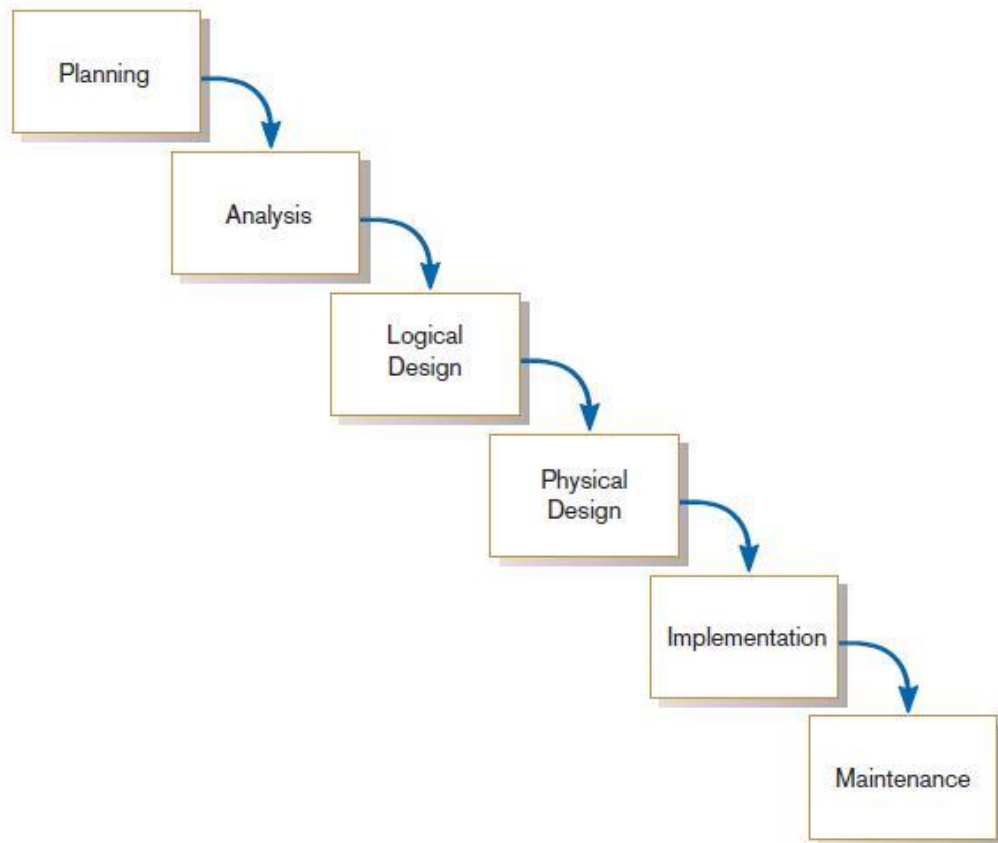
Dalam implementasi, dilakukan aktivitas-aktivitas sebagai berikut:

- Pembuatan database sesuai skema rancangan.
- Pembuatan aplikasi berdasarkan desain sistem.
- Pengujian dan perbaikan aplikasi (debugging).

### ***E. Pemeliharaan Sistem (Systems Maintenance)***

Dilakukan oleh admin yang ditunjuk untuk menjaga sistem tetap mampu beroperasi secara benar melalui kemampuan sistem dalam mengadaptasikan diri sesuai dengan kebutuhan.

### Traditional WATERFALL SDLC



### Permasalahan Metode Waterfall

1. Feedback diabaikan, milestones mengunci spesifikasi desain meskipun kondisi berubah. (jika ada perubahan, harus diulang dari tahap awal lagi)
2. Keterlibatan pengguna terbatas (hanya dalam tahap persyaratan)
3. Terlalu banyak fokus pada deadline milestone pada fase SDLC sehingga merugikan praktik pengembangan yang baik

### Metode Agile

Merupakan metode yang lebih modern dari pada Waterfall. Metode ini termotivasi dari pengakuan dimana pengembangan perangkat lunak itu seperti cairan, tidak dapat diprediksi, dan dinamis. Tidak seperti waterfall yang terkesan kaku dan sulit untuk menerima perubahan.

### Three key principles

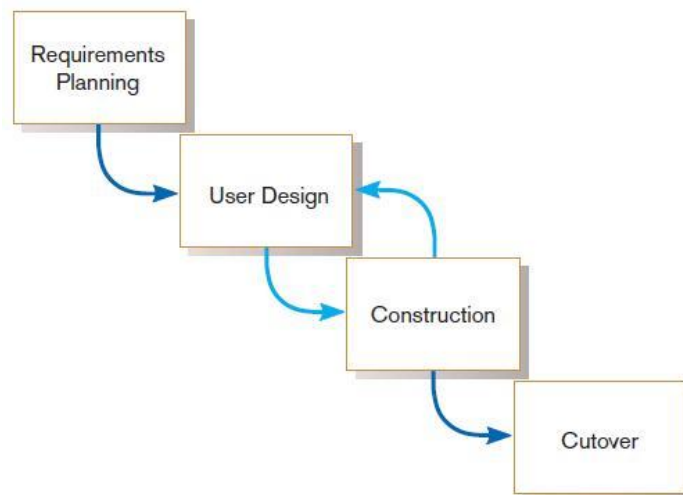
- Adaptif dari pada prediktif

- Menekankan pada *people* daripada *roles*
- Self-adaptive processes

### When to use Agile?

- If your project involves:
  - ☐ Unpredictable or dynamic requirements
  - ☐ Responsible and motivated developers
  - ☐ Customers who understand the process and will get involved

### Metode Rapid Application Development (RAD)



- Decreases design and implementation time
- Involves: extensive user involvement, prototyping, integrated CASE tools, code generators
- More focus on user interface and system function, less on detailed business analysis and system performance



### Metode eXtreme Programming

- Short, incremental development cycles
- Automated tests
- Two-person programming teams
- Coding, testing, listening, designing (coding & testing dilaksanakan bersamaan)
- **Advantages:**
  - Communication between developers
  - High level of productivity
  - High-quality code

17. **D. a dan b Benar**
18. **A. Identifying potential development projects**
19. **A. Corporate Strategic Planning**
20. **B. Information Systems Planning**
21. **E. Short Term Plan**
22. **E. Establishing Information System**
23. **E. Benar semua**
24. **E. Benar semua**
25. **B. Environmental**
26. **C. Selection Group**
27. **C. Anggap segala sesuatu mungkin terjadi**
28. **B. Existing written documents**
29. **E. Case Tools**
30. **A. Useful document: Written work procedure**
31. **A. Joint Application Design (JAD)**

### Pembahasan (17-31)

### PLANNING

Three main steps Identifying and Selecting Systems Development Projects

1. Identifying potential development projects
2. Classifying and ranking IS development projects
3. Selecting IS development projects

Bagian dari identifying potential development projects :

1. Identifikasi dari kelompok pemangku kepentingan
2. Sumber top-down adalah proyek yang diidentifikasi oleh manajemen puncak atau oleh komite pengarah yang beragam
3. Sumber bottom-up adalah inisiatif proyek yang berasal dari manajer, unit bisnis, atau kelompok pengembangan
4. Variasi proses di seluruh organisasi.

Corporate and Information Systems Planning Strategi perusahaan melibatkan:

1. Pernyataan misi : pernyataan yang menjelaskan bisnis apa yang ada di perusahaan
2. Pernyataan obyektif : serangkaian pernyataan yang mengungkapkan tujuan kualitatif dan kuantitatif organisasi untuk mencapai posisi masa depan yang diinginkan
3. Deskripsi strategi kompetitif : metode yang digunakan organisasi untuk mencapai misi dan tujuannya, dengan beberapa tipe seperti :
  - a. Low-cost producer
  - b. Product differentiation
  - c. Product focus or niche

### Information Systems Planning (ISP)

Sarana yang teratur untuk menilai kebutuhan informasi organisasi dan mendefinisikan sistem, database, dan teknologi yang paling sesuai dengan kebutuhan tersebut. ISP harus dilakukan sesuai dengan misi, tujuan, dan strategi kompetitif organisasi.

1. *Top-down planning* : upaya untuk memperoleh pemahaman luas tentang kebutuhan sistem informasi seluruh organisasi dan penawaran:

- a. Perspektif yang lebih luas.

- b. Peningkatan integrasi.
- c. Dukungan manajemen yang ditingkatkan.
- d. Pemahaman yang lebih baik.

2. *Bottom-up planning* : perencanaan mengidentifikasi proyek pengembangan IS berdasarkan penyelesaian masalah bisnis operasional tertentu atau memanfaatkan peluang spesifik

3. *Functional Decomposition* : memecah informasi abstrak tingkat tinggi ke dalam unit yang lebih kecil untuk perencanaan yang lebih rinci. Contoh: Kegiatan paralel perencanaan strategis perusahaan dan perencanaan sistem informasi

Komponen Information System Plan:

1. Organizational Mission, Objectives, and Strategy : Deskripsi singkat mengenai misi, tujuan, dan strategi organisasi
2. Information Inventory : Ringkasan proses, fungsi, entitas data, dan kebutuhan informasi perusahaan
3. Mission and Objectives of IS : Peran utama IS akan berperan dalam organisasi untuk mengubah perusahaan dari kondisi saat ini menjadi masa depan
4. Constraints on IS Development : Keterbatasan yang dipaksakan oleh teknologi dan tingkat sumber daya keuangan, teknis, dan personel saat ini
5. Systems Needs and IS Strategy : Ringkas semua kebutuhan sistem informasi di perusahaan dan tentukan strategi jangka panjang (2-5 tahun) untuk mengisi kebutuhan
6. Short Term Plan : Inventarisasi terperinci untuk proyek dan sistem saat ini dan rencana terperinci untuk tahun berjalan
7. Conclusions : Peristiwa tidak diketahui tetapi kemungkinan yang dapat mempengaruhi rencana, saat ini diketahui elemen perubahan bisnis dan dampaknya terhadap rencana

#### *The Process of Initiating and Planning IS Development Projects*

Kegiatan utama perencanaan proyek adalah proses mendefinisikan kegiatan yang jelas dan terpisah dan pekerjaan yang diperlukan untuk menyelesaikan setiap kegiatan dalam

satu proyek.

- a. Establishing the Project Initiation Team
- b. Establishing a Relationship with the Customer
- c. Establishing the Project Initiation Plan
- d. Establishing Management Procedures
- e. Establishing the Project Management Environment and Project Workbook
- f. Developing the Project Charter

#### Business Case

- a. Pembenaran untuk sistem informasi
- b. Disajikan dalam bentuk manfaat dan biaya ekonomi yang nyata dan tidak berwujud
- c. Kelayakan teknis dan organisasi dari sistem yang diusulkan

#### Baseline Project Plan (BPP)

- a. Hasil utama dan hasil dari fase PIP
- b. Berisi estimasi terbaik dari ruang lingkup proyek, manfaat, biaya, risiko, dan persyaratan sumber daya

#### Project Scope Statement (PSS)

- a. Dokumen yang disiapkan untuk pelanggan
- b. Menjelaskan apa yang akan disampaikan proyek
- c. Garis besar pada semua pekerjaan yang diperlukan untuk menyelesaikan proyek

#### Assessing Project Feasibility

1. Economic : proses mengidentifikasi manfaat finansial dan biaya yang terkait dengan proyek pengembangan
2. Technical : proses menilai kemampuan organisasi pembangunan untuk membangun sistem yang diusulkan
3. Operational : Apakah sistem yang diusulkan memecahkan masalah atau memanfaatkan peluang
4. Scheduling : Dapatkah kerangka waktu proyek dan tanggal penyelesaian memenuhi

tenggat waktu organisasi

5. Legal and contractual : Dapatkah kerangka waktu proyek dan tanggal penyelesaian memenuhi tenggat waktu organisasi?

6. Political : How do key stakeholders view the proposed system?

### Project Risk Factors

1. Ukuran proyek :Ukuran tim, departemen organisasi, durasi proyek, upaya pemrograman
2. Struktur proyek : Sistem baru vs. direnovasi, menghasilkan perubahan organisasi, komitmen manajemen, persepsi pengguna
3. Development Group : Keakraban dengan platform, perangkat lunak, metode pengembangan, area aplikasi, pengembangan sistem serupa
4. User Group : Keakraban dengan proses pengembangan IS, area aplikasi, penggunaan sistem serupa

## **REQUIREMENT**

### Good Systems Analyst Characteristics

- a. Impertinence : mempertanyakan segalanya
- b. Impartiality : pertimbangkan semua masalah untuk menemukan solusi organisasi terbaik
- c. Relax constraints : anggap segala sesuatu mungkin terjadi
- d. Attention to details : setiap fakta harus sesuai
- e. Reframing : tantang diri Anda dengan cara baru

### Deliverables for Requirements Determination:

- a. From interviews and observations : transkrip wawancara, catatan observasi, notulen rapat
- b. From existing written documents: pernyataan misi dan strategi, formulir bisnis, manual prosedur, deskripsi pekerjaan, manual pelatihan, dokumentasi sistem, diagram alur
- c. From computerized sources : Hasil sesi Desain Aplikasi Bersama, repositori KASUS,

laporan dari sistem yang ada, tampilan dan laporan dari prototipe sistem

*Traditional Methods for Determining Requirements :*

1. Interviewing individuals
2. Interviewing groups
3. Observing workers
4. Studying business documents

*Choosing Interview Questions*

1. Pertanyaan terbuka: pertanyaan yang tidak memiliki jawaban yang ditentukan sebelumnya
2. Pertanyaan tertutup: pertanyaan yang meminta mereka yang menjawab untuk memilih dari antara serangkaian tanggapan tertentu

*Analyzing Procedures and Other Documents*

1. Document Analysis

Tinjau dokumen bisnis yang ada Dapat memberikan pandangan historis dan "formal" tentang persyaratan sistem

2. Useful document: Written work procedure : Menjelaskan bagaimana pekerjaan atau tugas tertentu dilakukan, termasuk data dan informasi yang digunakan dan dibuat dalam proses
3. Formal Systems: cara resmi suatu sistem bekerja seperti yang dijelaskan dalam dokumentasi organisasi (yaitu prosedur kerja)
4. Informal Systems: cara sistem sebenarnya bekerja (mis. Wawancara, pengamatan)
5. Useful document: Business form

Digunakan untuk semua jenis fungsi bisnis, Menunjukkan secara eksplisit aliran data apa yang masuk dan keluar dari suatu sistem dan data yang diperlukan agar sistem berfungsi dan Memberikan informasi penting tentang sifat organisasi

*Contemporary Methods for Determining System Requirements*

**Joint Application Design (JAD)**

- Brings together key users, managers, and systems analysts
- Purpose: collect system requirements simultaneously from key people
- Conducted off-site

**Group Support Systems**

- Facilitate sharing of ideas and voicing of opinions about system requirements

**CASE tools**


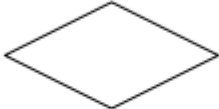



- Used to analyze existing systems
- Help discover requirements to meet changing business conditions

**System prototypes**

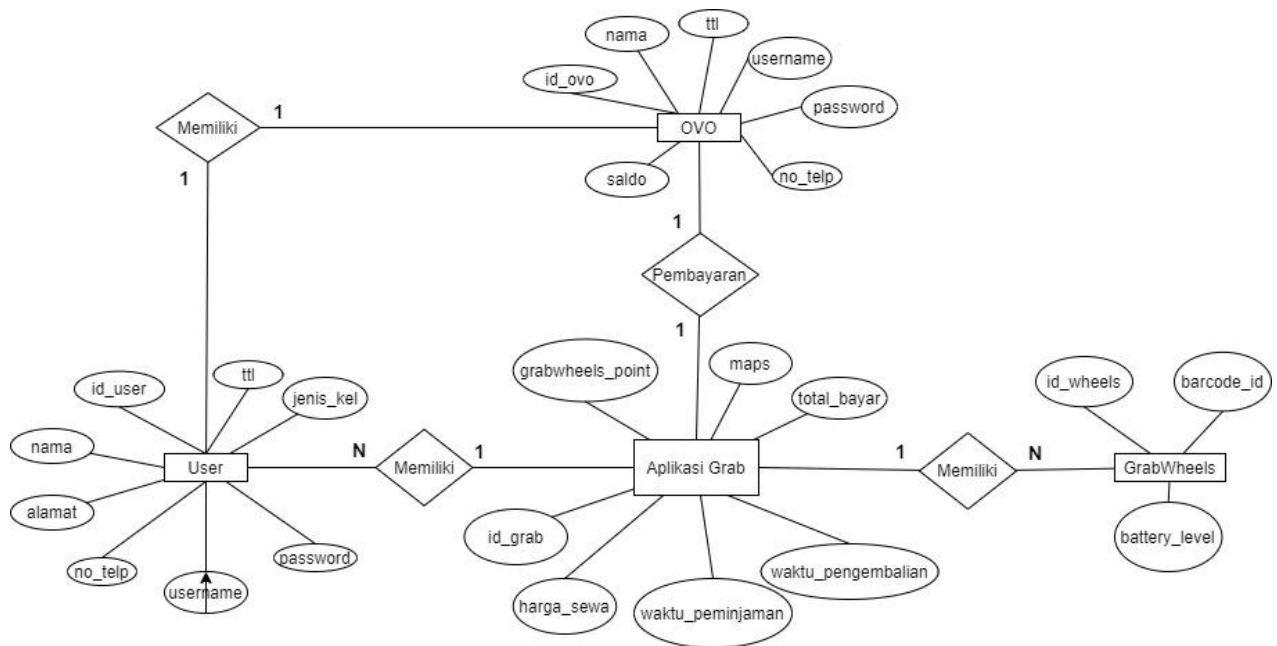
- Iterative development process
- Rudimentary working version of system is built
- Refine understanding of system requirements in concrete terms

**Essay!**

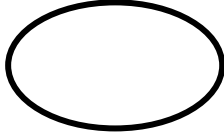


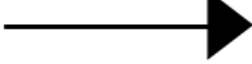
**ERD**

No	Simbol	Nama	Keterangan
1		Entitas	Jenis entitas dapat berupa suatu elemen lingkungan, sumber daya atau transaksi yang <i>field-fieldnya</i> dipergunakan dalam aplikasi program
2		Hubungan atau Relasi	Menunjukkan nama relasi antar satu entitas dengan entitas lainnya
3		Atribut	Atribut adalah karakteristik dari sebuah entitas
4		Garis Relasi	Menunjukkan hubungan (keterkaitan) antar entitas
5		Entitas Lemah	Entitas yang kemunculannya tergantung dari entitas lain yang lebih kuat

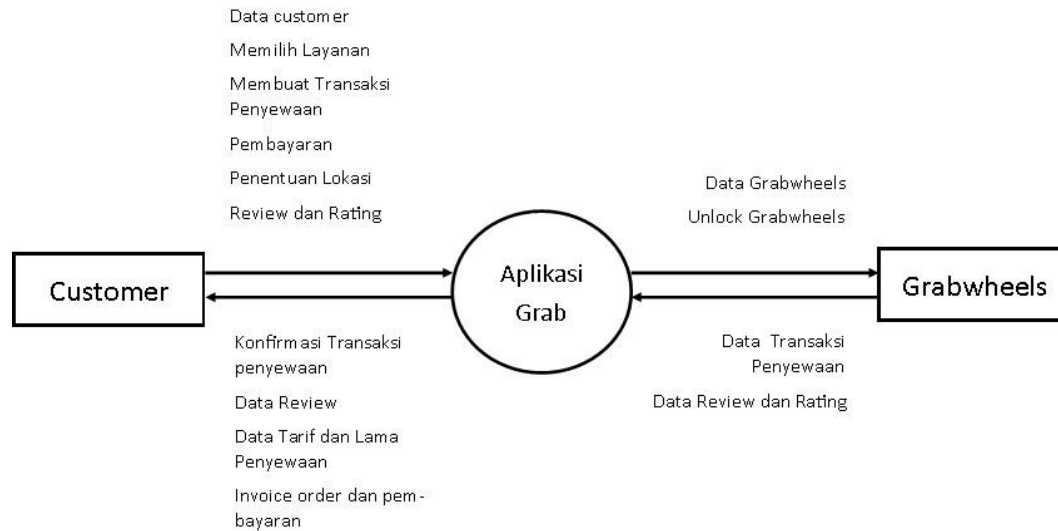




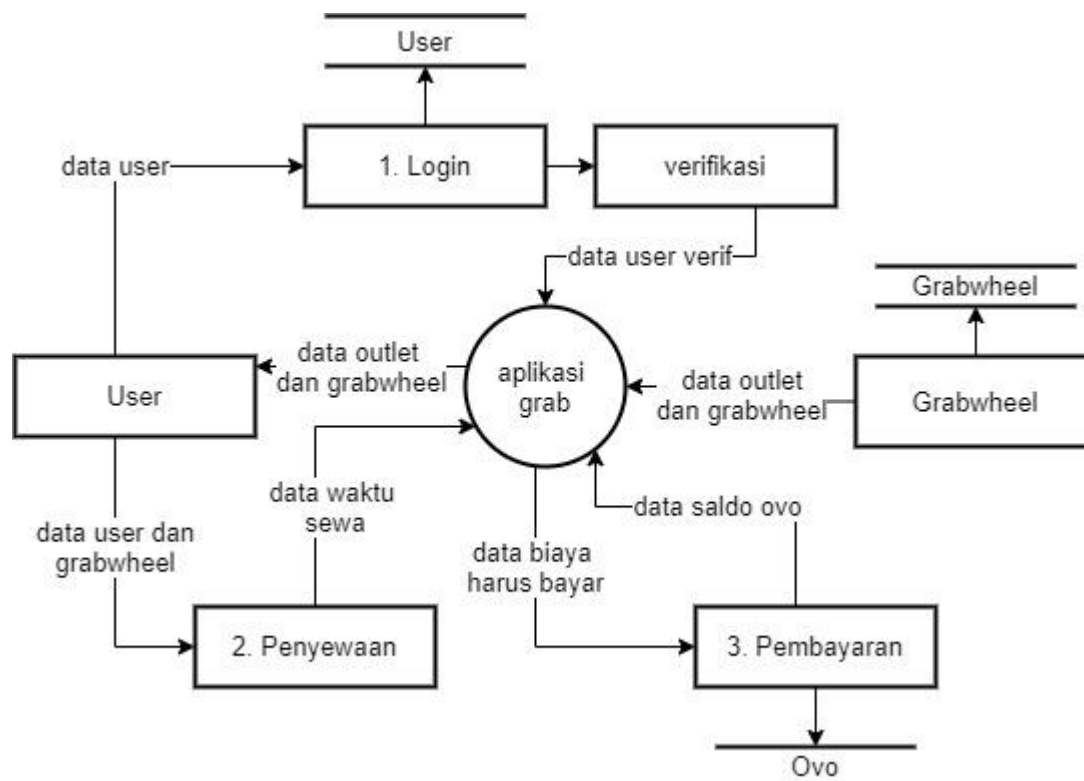
## DFD

Simbol	Nama	Keterangan
	<b>Process</b>	work or actions performed on data (inside the system)
	<b>Data Store</b>	data at rest (inside the system)
	<b>Source/Sink</b>	external entity that is the origin or destination of data (outside the system)
	<b>Data Flow</b>	arrows depicting movement of data

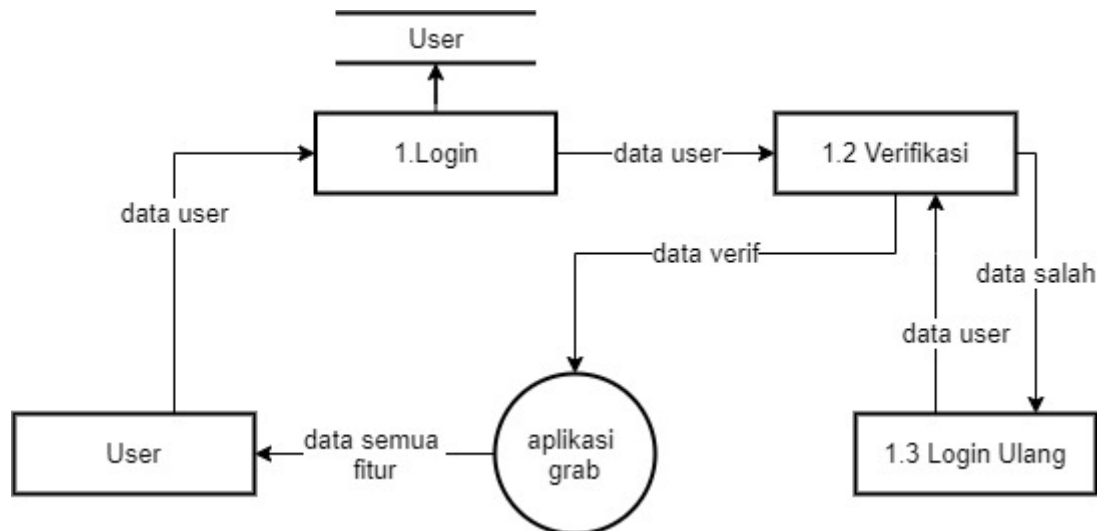
### Data Flow Diagram Level 0



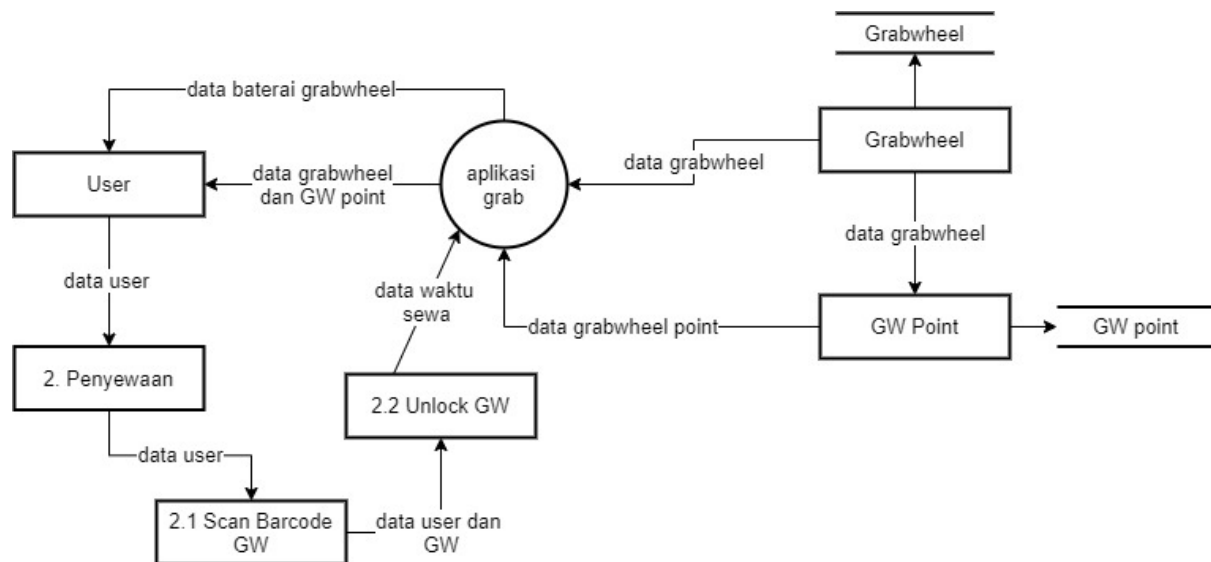
### Data Flow Diagram Level 1



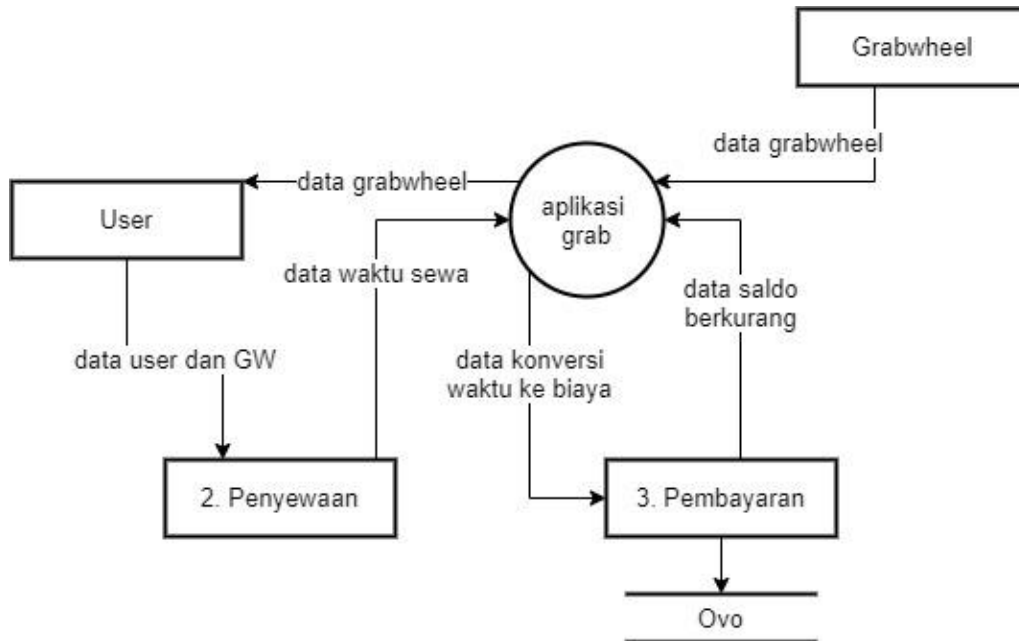
### Data Flow Diagram Level 2 Login



### Data Flow Diagram Level 2 Penyewaan



### Data Flow Diagram Level 2 Pembayaran



### Use Case Diagram


Rosa dan Shalahuddin (2013), menjelaskan tentang use case diagram sebagai berikut : Use case merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem .

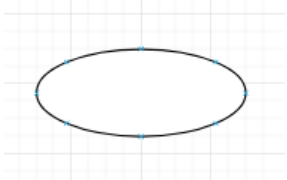



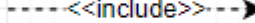

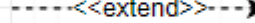

Secara umum komponen pembentuk diagram use case adalah:

1. Aktor (actor), menggambarkan pihak-pihak yang berperan dalam sistem.
2. Use case, aktivitas / sarana yang disiapkan oleh bisnis / sistem.
3. Hubungan (link), aktor mana saja yang terlibat dalam use case ini.

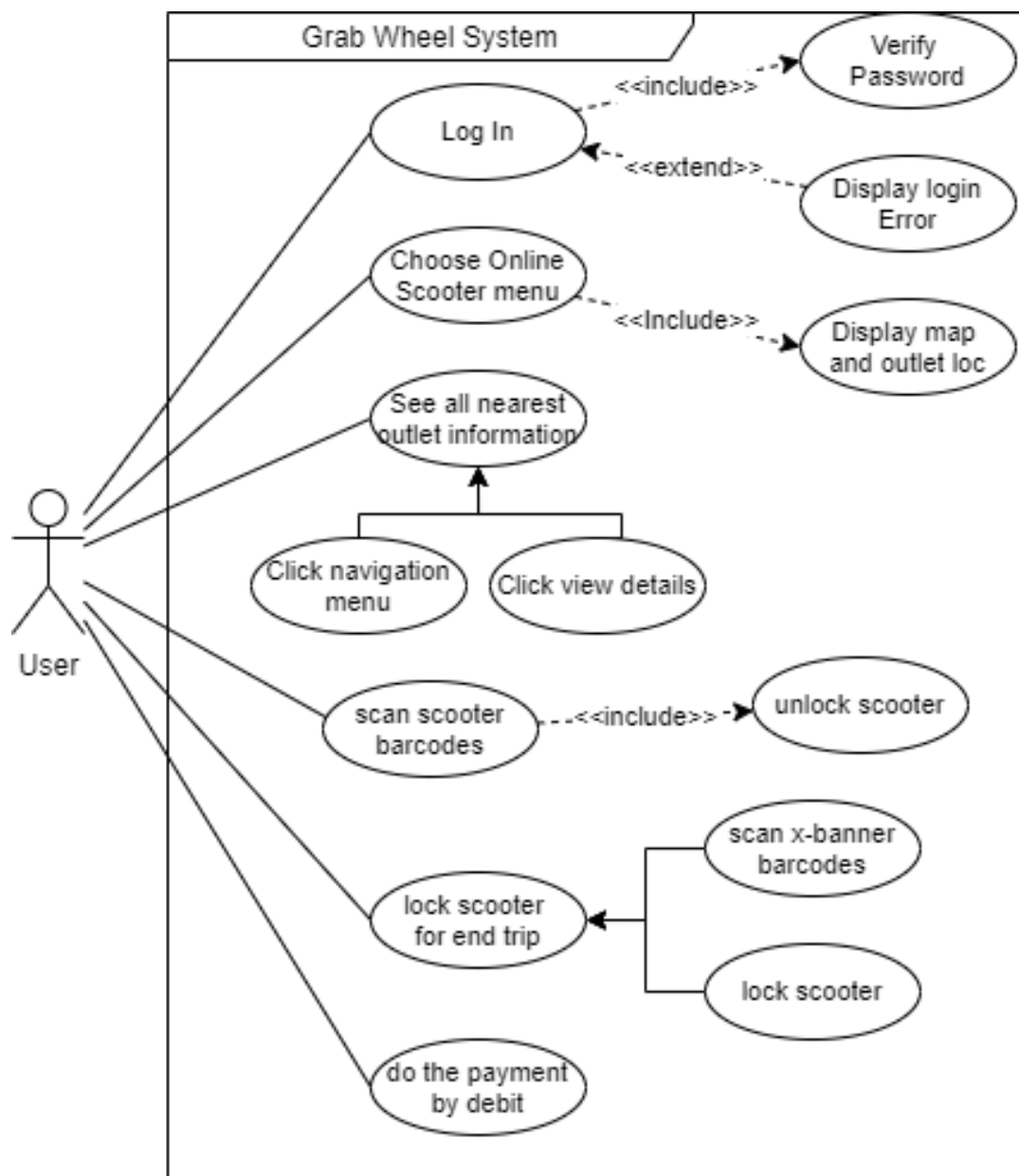
Boundary, ruang lingkup sistem bekerja secara sistematis.

Adapun simbol-simbol yang digunakan dalam use case adalah sebagai berikut:








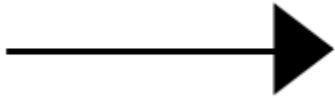
Simbol	Nama	Keterangan
	<b>Actor</b>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri. (user, admin, operator, customer, instansi, organisasi, dst.) diluar

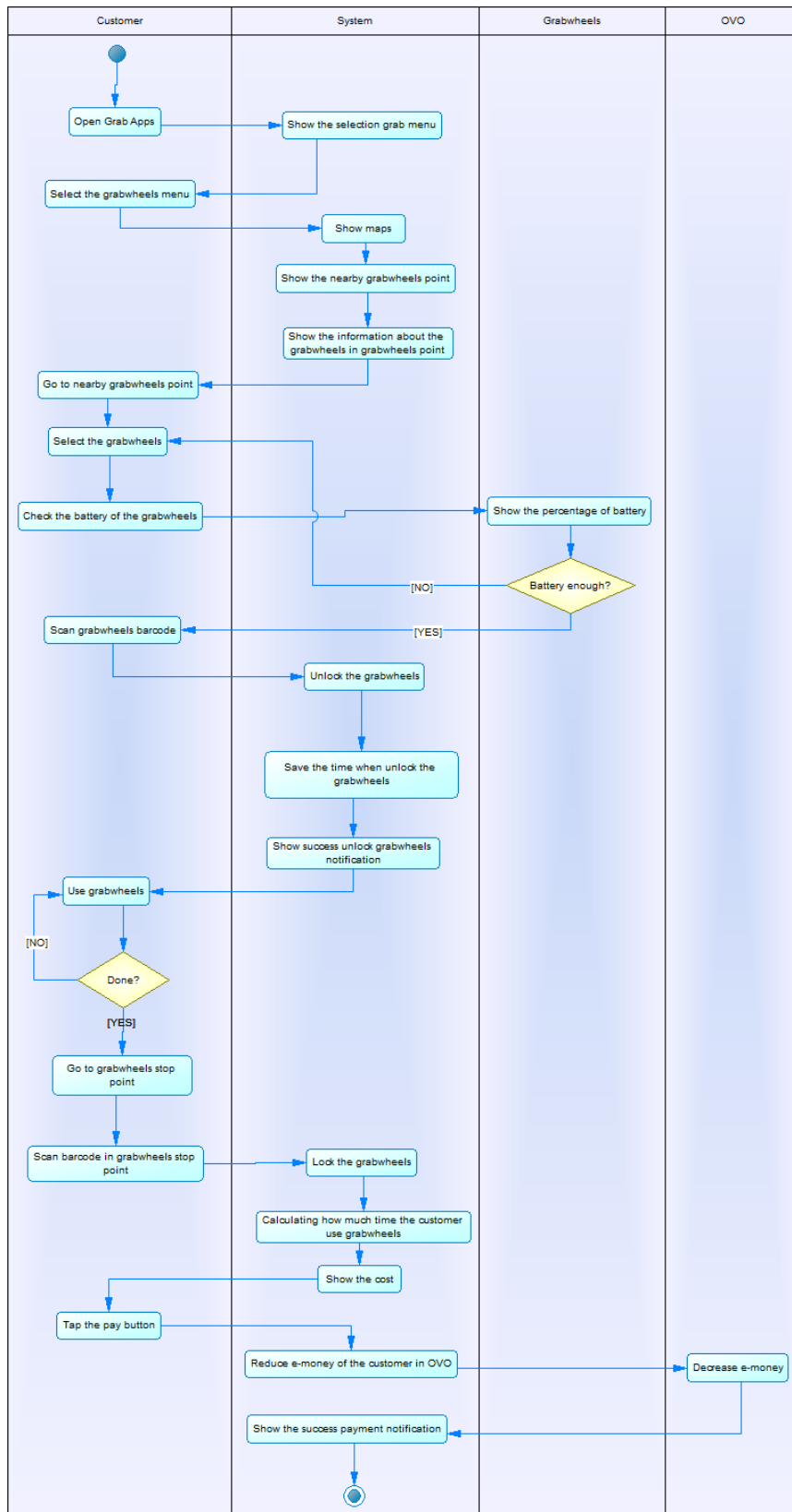
		system boundaries
	<b>Use Case</b>	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau actor.  (kata kerja + kata benda) (pilih menu <u>stirling</u> ), (scan <u>barcodes</u> )
	<b>Association</b>	Komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor.
 	<b>Generalization</b>  Mengarah ke parent	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah use case dimana fungsi usecase parent adalah fungsi yang lebih umum dari lainnya.
 	<b>Include</b>  Mengarah ke child	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case parent untuk menjalankan fungsinya atau sebagai syarat dijalankan use case ini.
 	<b>Extend</b>  Mengarah ke parent	Relasi use case tambahan ke sebuah use case yang parent sebenarnya masih dapat berdiri sendiri walau tanpa use case tambahan.

<div style="border: 1px solid black; width: 100px; height: 100px; margin: 10px auto; text-align: center; line-height: 100px;">system blabla</div>	<b>System Boundary</b>	Merupakan ruang lingkup suatu sistem untuk bekerja sesuai dengan tujuan sistem dibuat.
---	------------------------	--



### Activity Diagram

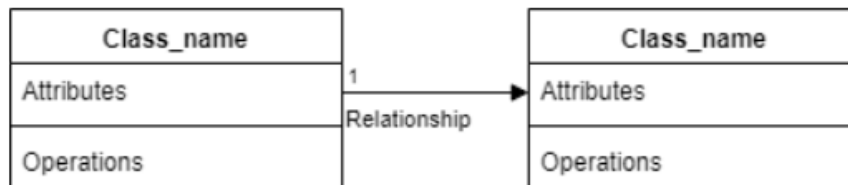
Simbol	Nama	Keterangan
	<b>Initial State</b>	Present the start of a system's workflow in the activity diagram.
	<b>Final State</b>	Present that the flow of a system has ended in the activity diagram.
	<b>Activity</b>	Describe a business process or business activity
	<b>Fork</b>	Used to indicate activities carried out in parallel or split behavior into a set of parallel or concurrent flows of activities (or actions)
	<b>Join</b>	Combining two parallel activities into one or Bring back together a set of parallel or concurrent flows of activities (or actions)
	<b>Decision</b>	Describe choices for decision making, true or false or represent a test condition to ensure that the control flow or object flow only goes down one path
	<b>Swimlane</b>	A way to group activities performed by the same actor on an activity diagram or to group activities in a single thread or the division of activity diagrams to show who does what
	<b>Control Flow / Flow</b>	Shows the sequence of execution










## Class Diagram

### a. Component



Nama	Keterangan
<b>Class name</b>	to differentiate between one class and another class.
<b>Attributes</b>	are the variables that belong to a class.
<b>Operations</b>	operations or activities that a class can perform.

### b. Relationship

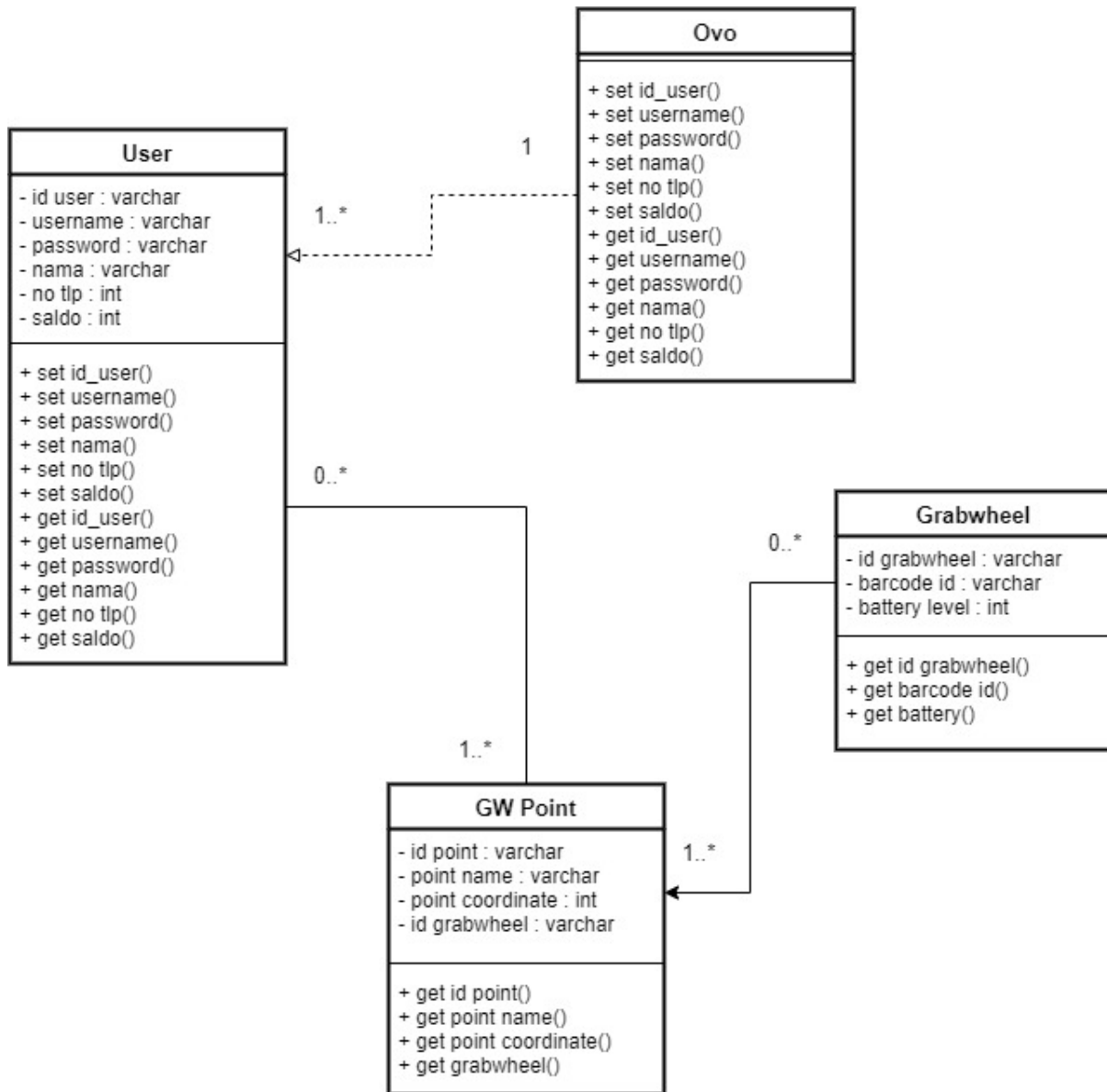
<p style="text-align: center;">Association</p> 	Relations between classes with general meaning, associations are usually accompanied by multiplicity
<p style="text-align: center;">Directed Association</p> 	Relationships between classes with the meaning of one class used by another class, associations are usually accompanied by multiplicity.
<p style="text-align: center;">Generalization</p> 	Relationships between classes with the meaning of generalization-specialization (special general).
<p style="text-align: center;">Dependency</p> 	Relationships between classes with the meaning of interdependence between classes.
<p style="text-align: center;">Aggregation</p> 	Relationships between classes with the meaning of all-part (whole-part).

**c. Visibility**

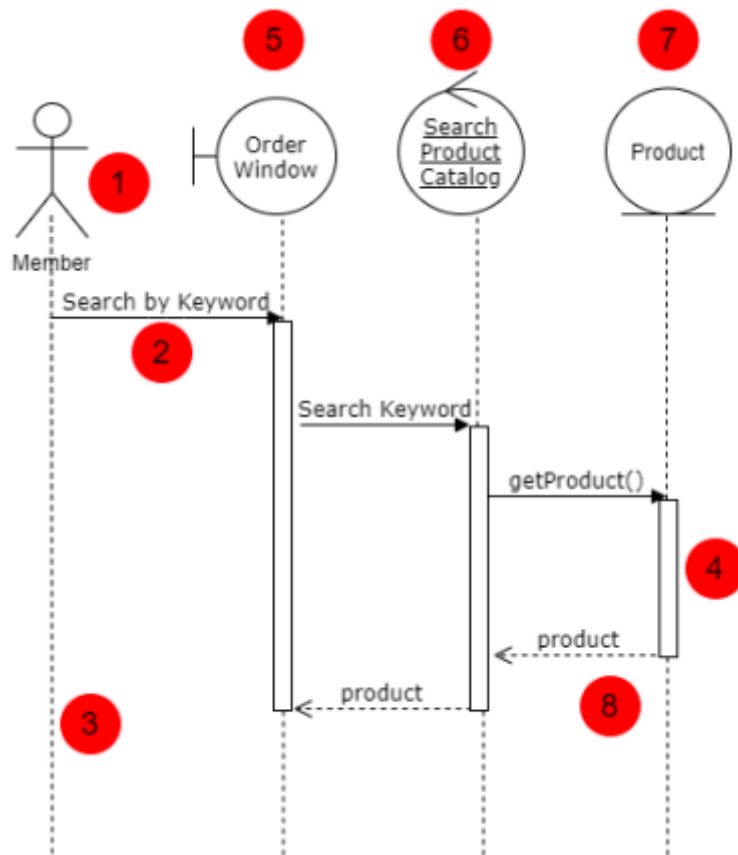
Symbol	Meaning
“+”	Public (can be called from all objects)
“-“	Private (can only be called from inside the class itself)
“#”	Protected (can only be called by corresponding class and its derived class)
“ “	Default/ without symbol

**d. Kardinalitas**

Nilai Kardinalitas	Arti	Contoh	
 0..1	Nol atau satu	karyawan	0..1 istri
1	Hanya satu	negara	1 presiden
0..*	Nol atau lebih	karyawan	0..* anak
1..*	Satu atau lebih	bos	1..* bawahan
n	Hanya n (dengan n > 1)	karyawan	n cek up
0..n	Nol sampai n (dengan n > 1)	karyawan	0..n sim
1..n	Satu sampai n (dengan n > 1)	kereta api	1..n gerbong

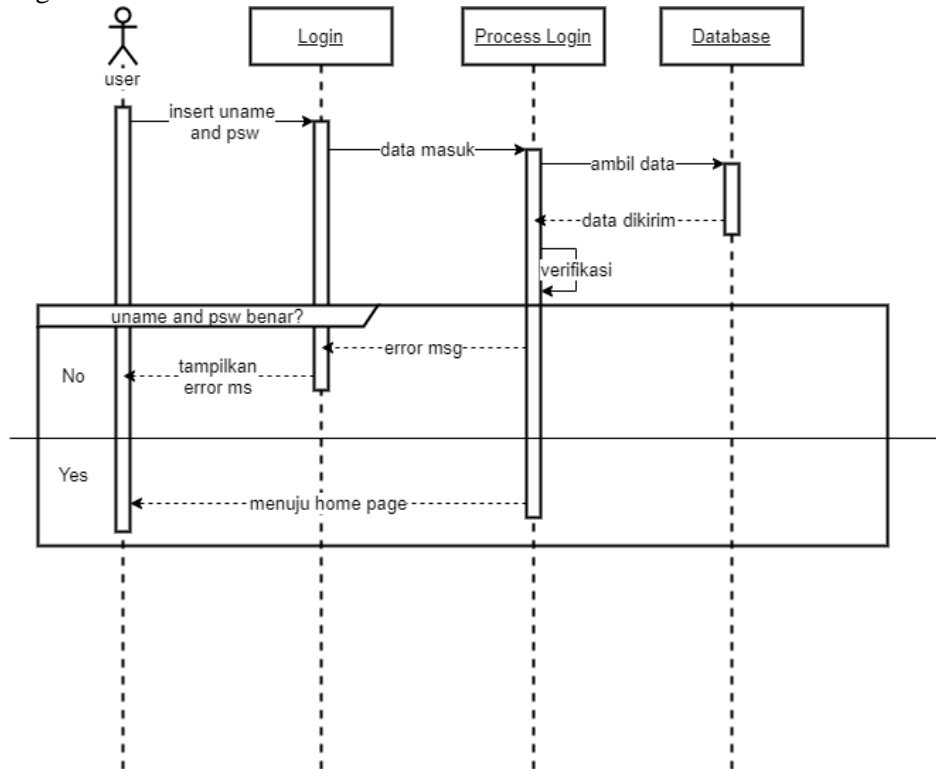


## Sequence Diagram

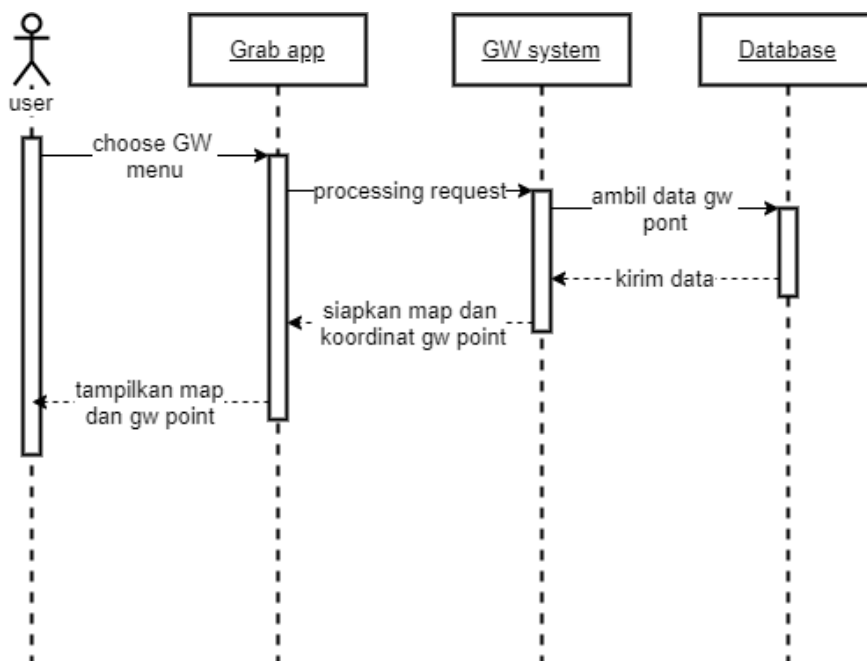


No.	Nama	Keterangan
1	Actor	User who started the event / event flow.
2	Message	Described as the arrow line from one object to another object.
3	Lifeline	A dashed vertical line extending down from the actor or other system symbol, which shows the object's life during interaction.
4	Activation bars	Shows how long a message is processed.
5	Boundary Class	Describes a depiction of the form.
6	Control Class	Describes a depiction of the form.
7	Entity Class	Describe the relationship of activities to be carried out.
8	Return Message	Shows the return of a behavior method.

1. Login



2. Pilih menu grabwheel dan ke tenant



### 3. Menyewa Grabwheel di tenant

