FAKULTAS TEKNIK DAN INFORMATIKA

UNIVERSITAS BINA SARANA INFORMATIKA

MEMPERSEMBAHKAN

UNIVERSITAS
BINA SARANA INFORMATIKA
FAKULTAS TEKNIK & INFORMATIKA

WEBINAR UBSI

NEXT

NOTABLE EXPERT TALK

"Unified Modelling Language Utilization for System Modeling"

Live Via ZOOM

Narasumber
Adi Supriyatna, M.Kom

Moderator
Rifki Permana, M.Kom

KAMIS
4 Juni 2020

Contact Person
Dewi Ayu Nur Wulandari (0856 9463 9750)

Supported by :

Media Partner :

www.bsi.ac.id @kuliahbsiaja @kuliahbsiaja Bina Sarana Informatika

UNIFIED MODELLING LANGUAGE UTILIZATION FOR SYSTEM MODELING

Adi Supriyatna, M.Kom

adi.asp@bsi.ac.id - http://adisupriyatna.6te.net

PROFIL ADI SUPRIYATNA

- SMA Yadika 4 Jatiwaringin (2003)
- A.Md, Manajemen Informatika AMIK BSI Jakarta (2006)
- S.Kom, Sistem Informasi STMIK Nusa Mandiri Jakarta (2008)
- M.Kom, Ilmu Komputer STMIK Nusa Mandiri Jakarta (2011)
- Dosen Bina Sarana Informatika (2007 - Sekarang)
- Dosen STMIK Nusa Mandiri (2012 - Sekarang)
- Ketua Program Studi Sistem Informasi Akuntansi (D3) FTI Universitas Bina Sarana Informatika (2019 - sekarang)

- Sertifikasi Pendidik Profesional (2013)
- Sertifikasi Kompetensi Programmer (2018)
- Asesor Kompetensi Programmer (2019)
- Sertifikasi Database Administrator (2020)
- Aktif Menulis Jurnal Nasional dan Internasional
- Bidang Ilmu : Web Programming, Data Warehouse, Data Base, Data Mining, etc.

WEBINAR OUTLINE

- APA ITU UML ?
- SEJARAH UML
- DIAGRAM UML
- PERANCANGAN UML
- PENUTUP

APA ITU UML ?

UML kependekan dari Unified Modelling Language

Ivar Jacobson, James Rumbaugh dan Grady Booch mengatakan *"UML can be used for modeling all processes in the development life cycle and across different implementation technologies"*

UML adalah bahasa standar untuk memvisualisasikan, menentukan, membangun, dan mendokumentasikan artefak dari sistem intensif perangkat lunak.

UML dapat digunakan sebagai alat komunikasi dalam pengembangan sistem dalam sebuah tim maupun dengan stakeholder.

SEJARAH UML

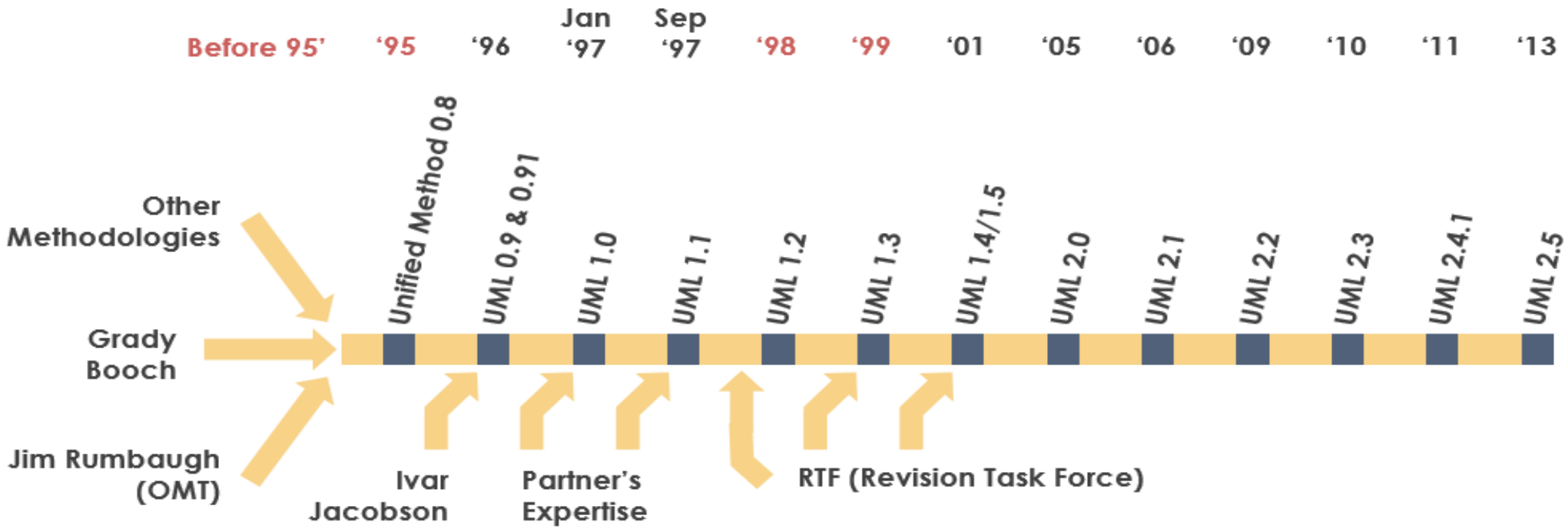
GRADY BOOCH

JAMES RUMBAUGH

IVAR JACOBSON

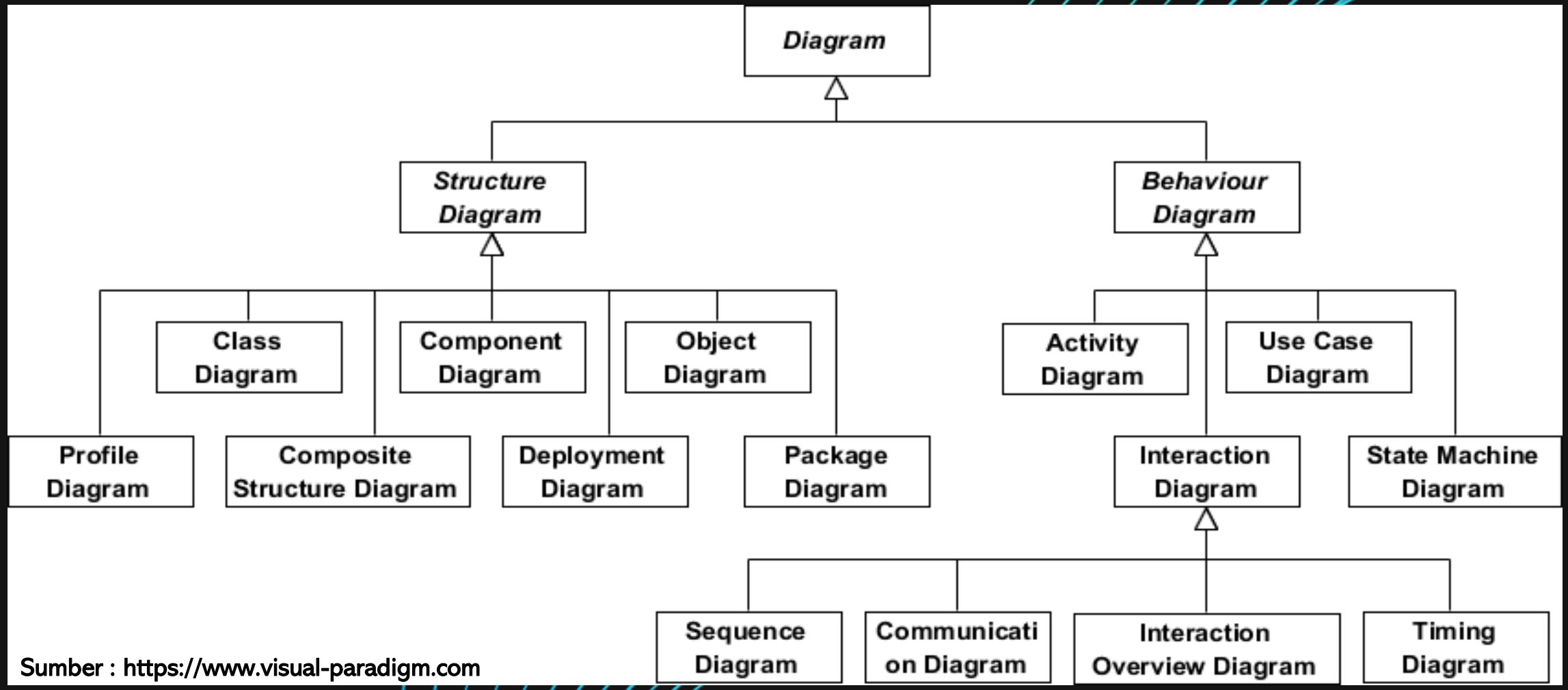
"THE THREE AMIGOS"

SEJARAH UML (LANJ.)



Sumber : https://www.visual-paradigm.com

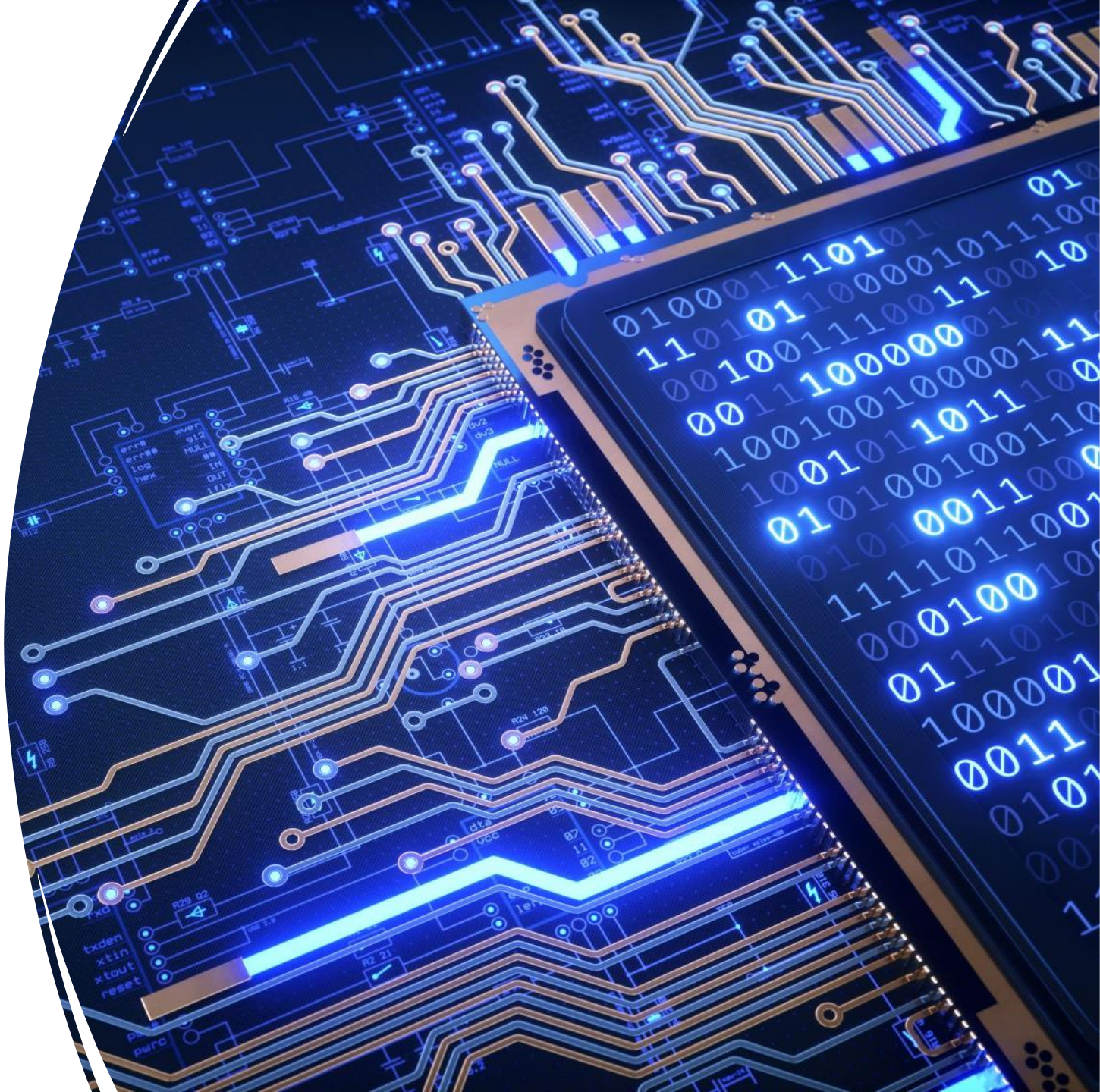
DIAGRAM UML 2.0



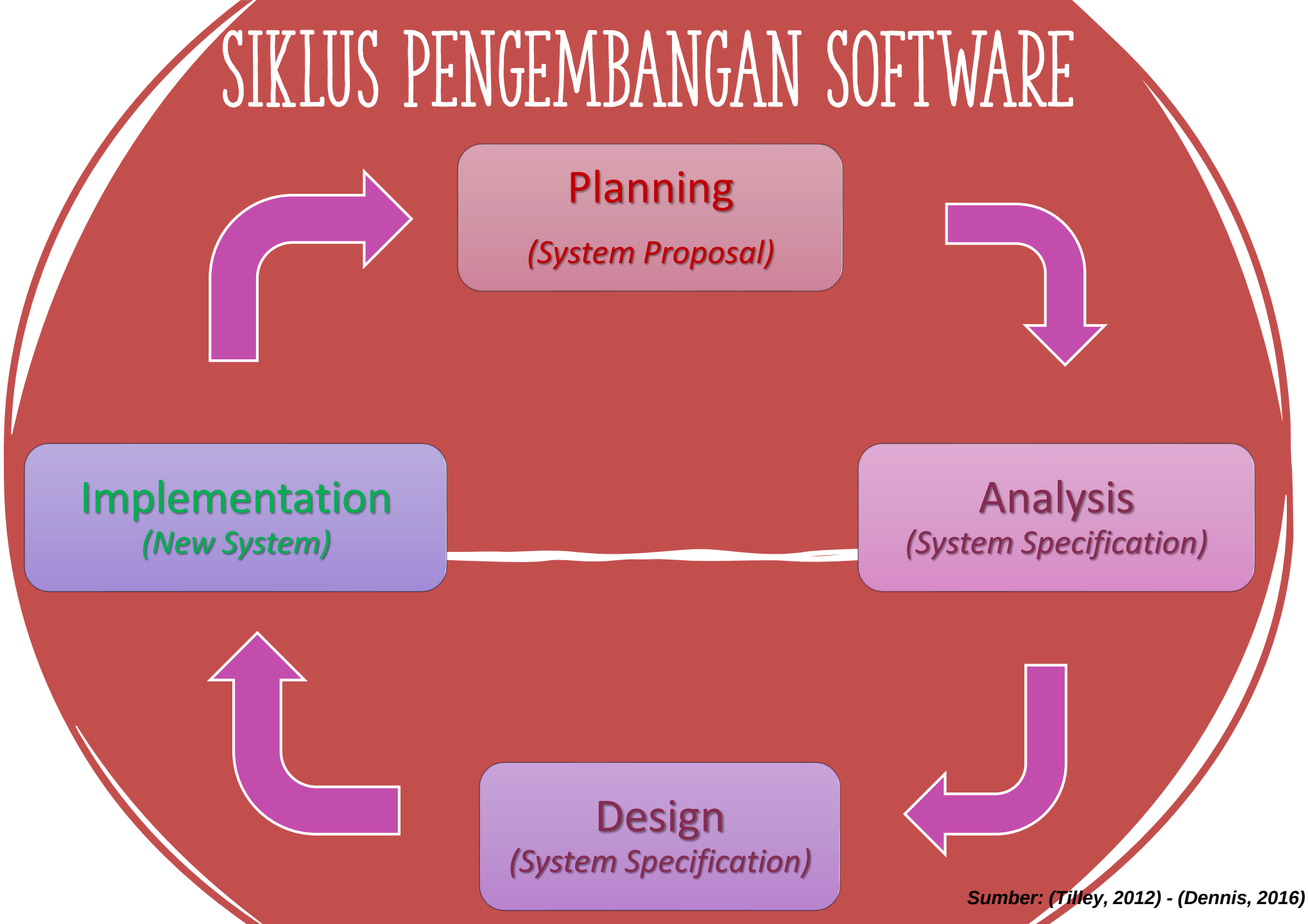
Sumber : https://www.visual-paradigm.com

PERTANYAN YG MUNCUL

- Apakah semua diagram UML harus digunakan pada setiap phase pengembangan perangkat lunak?
- Diagram apa yang digunakan untuk memulai pemodelan sistem?



SIKLUS PENGEMBANGAN SOFTWARE

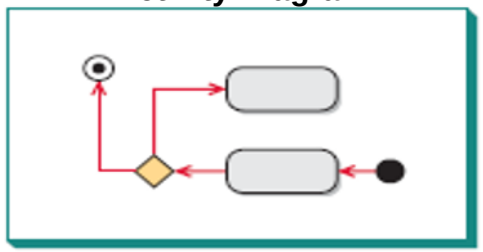


Sumber: (Tilley, 2012) - (Dennis, 2016) - (Valacich, 2017)

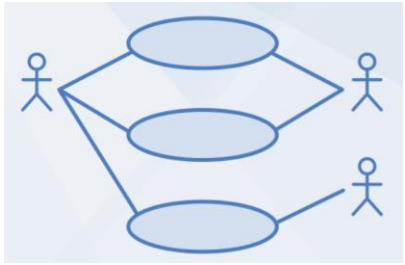
HUBUNGAN UML DGN PHASE PENGEMBANGAN SOFTWARE

Diagram Name	Used to	Primary Phase
Structure Diagrams		
Class	Illustrate the relationships between classes modeled in the system.	Analysis, Design
Object	Illustrate the relationships between objects modeled in the system. Used when actual instances of the classes will better communicate the model.	Analysis, Design
Package	Group other UML elements together to form higher level constructs.	Analysis, Design, Implementation
Deployment	Show the physical architecture of the system. Can also be used to show software components being deployed onto the physical architecture.	Physical Design, Implementation
Component	illustrate the physical relationships among the software components.	Physical Design, Implementation
Composite Structure	Illustrate the internal structure of a class, i.e., the relationships among the parts of a class.	Analysis, Design
Behavioral Diagrams		
Activity	Illustrate business workflows independent of classes, the flow of activities in a use case, or detailed design of a method.	Analysis, Design
Sequence	Model the behavior of objects within a use case. Focuses on the time-based ordering of an activity.	Analysis, Design
Communication	Model the behavior of objects within a use case. Focuses on the communication among a set of collaborating objects of an activity.	Analysis, Design
Interaction Overview	Illustrate an overview of the flow of control of a process.	Analysis, Design
Timing	Illustrate the interaction that takes place among a set of objects and the state changes in which they go through along a time axis.	Analysis, Design
Behavioral State Machine	Examine the behavior of one class.	Analysis, Design
Protocol State Machine	Illustrates the dependencies among the different interfaces of a class.	Analysis, Design
Use-Case	Capture business requirements for the system and to illustrate the interaction between the system and its environment.	Analysis

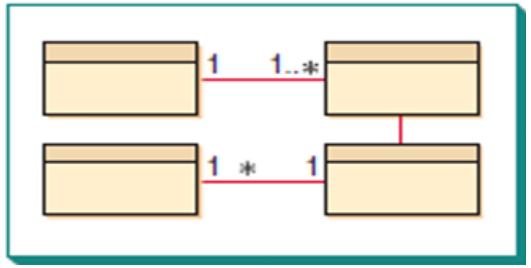
Activity Diagram

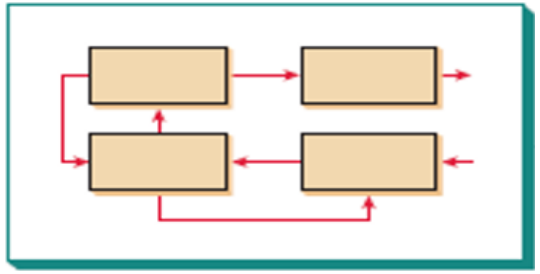


Use Case Diagram

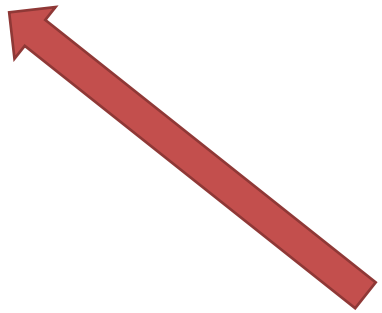


Class Diagram



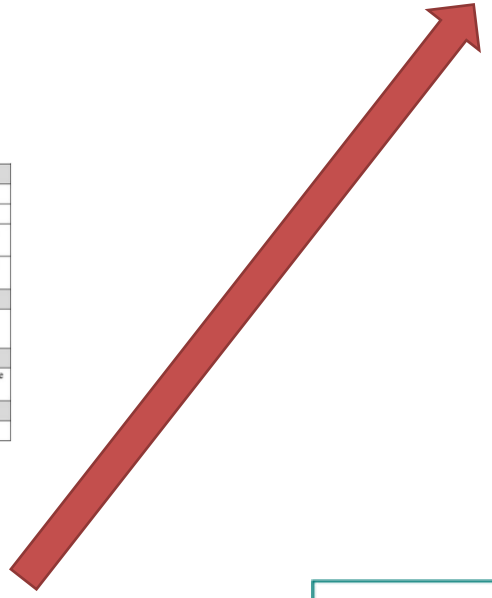


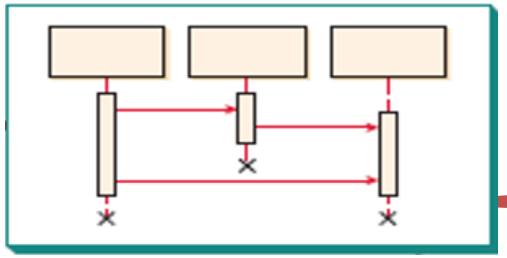
Statechart Diagram



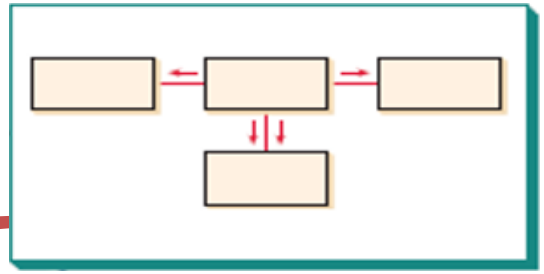
Scenario Use Case

Overview	
Title	[Title of the basic flow use case]
Description	[Short description of the basic flow]
Actors and Interfaces	[Identifies the Actors and Interfaces to components and services that participate in the use case]
Initial Status and Preconditions	[A pre-condition (of a use case) is the state of the system that must be present prior to a use case being performed]
Basic Flow	
STEP 1: …	
STEP 2: …	
Post Condition	
[A post-condition (of a use case) is a list of possible states the system can be in immediately after a use case has finished]	
Alternative Flow(s)	
[Alternative flows are described here if needed]	





Squence Diagram

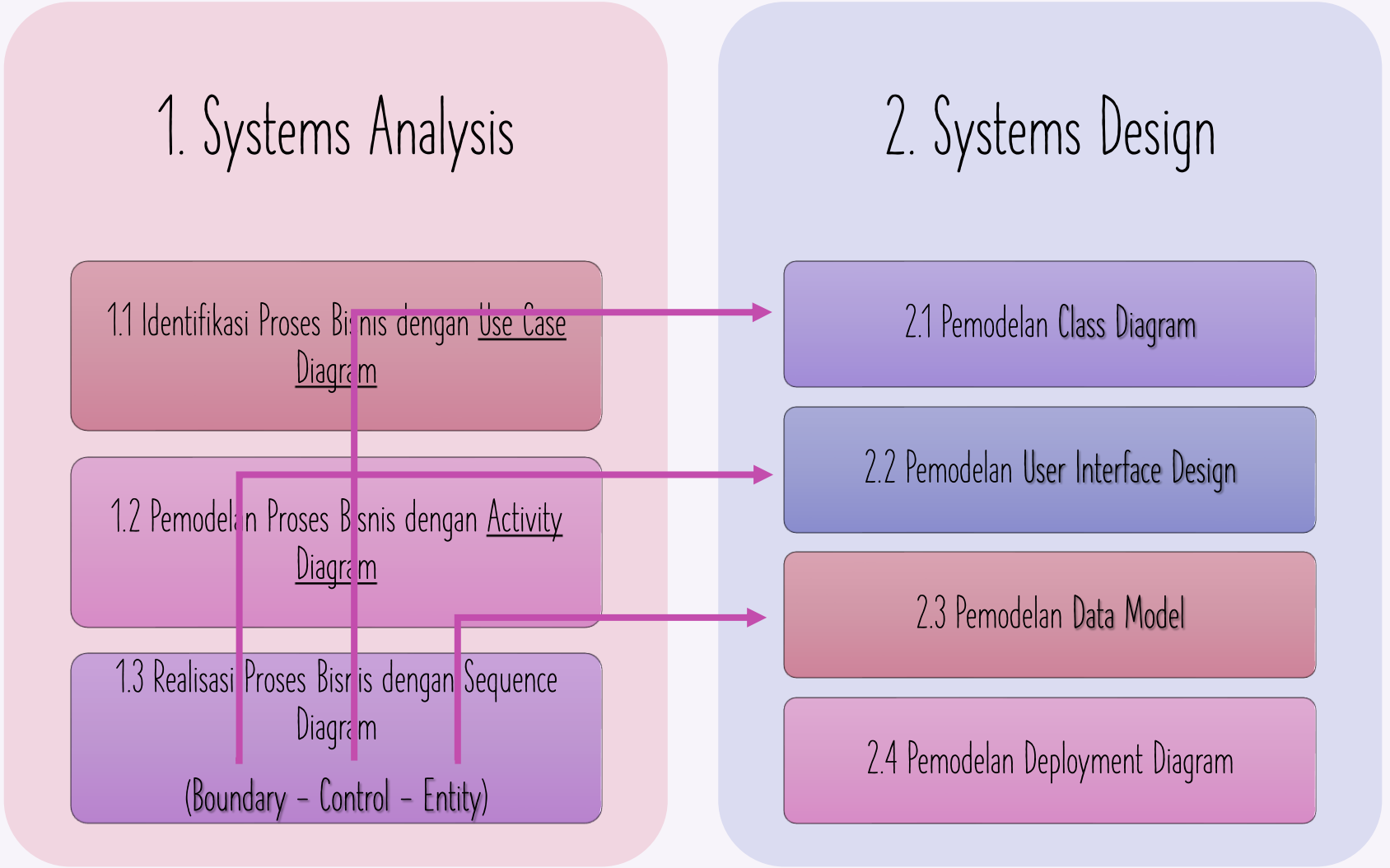


Communication Diagram

MEMULAI PEMODELAN SISTEM

Sumber : Kendall and Kendall, 2011

ANALISIS DAN DESAIN PERANGKAT LUNAK BERBASIS UML



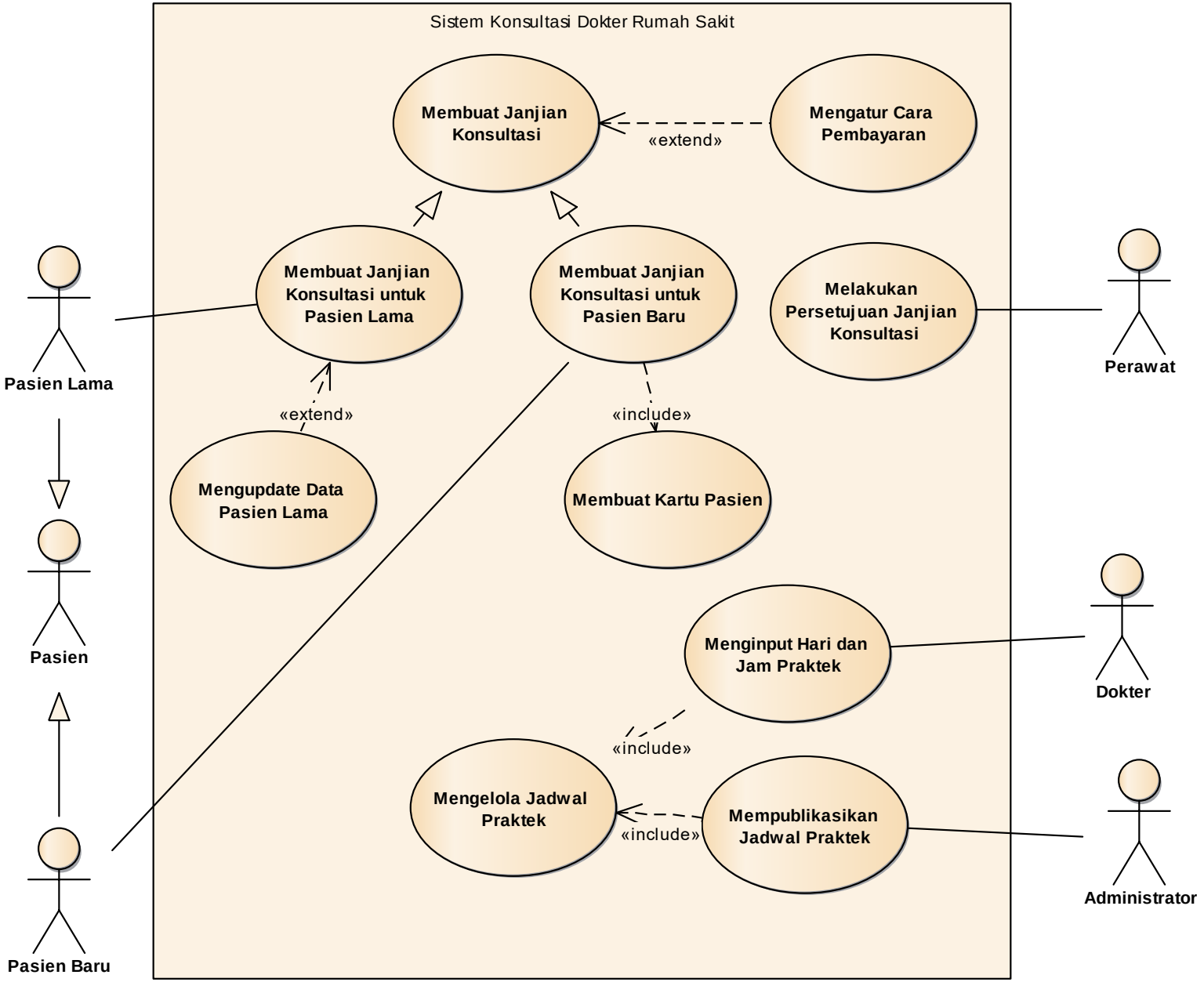
Sumber : Romi Satrio Wahono, 2009

1. USE CASE DIAGRAM

Rangkuman dari sistem yang akan dibangun yang Berisi apa yang dapat dilakukan user terhadap sistem Bukan apa yang dilakukan oleh sistem.

Simbol Use Case

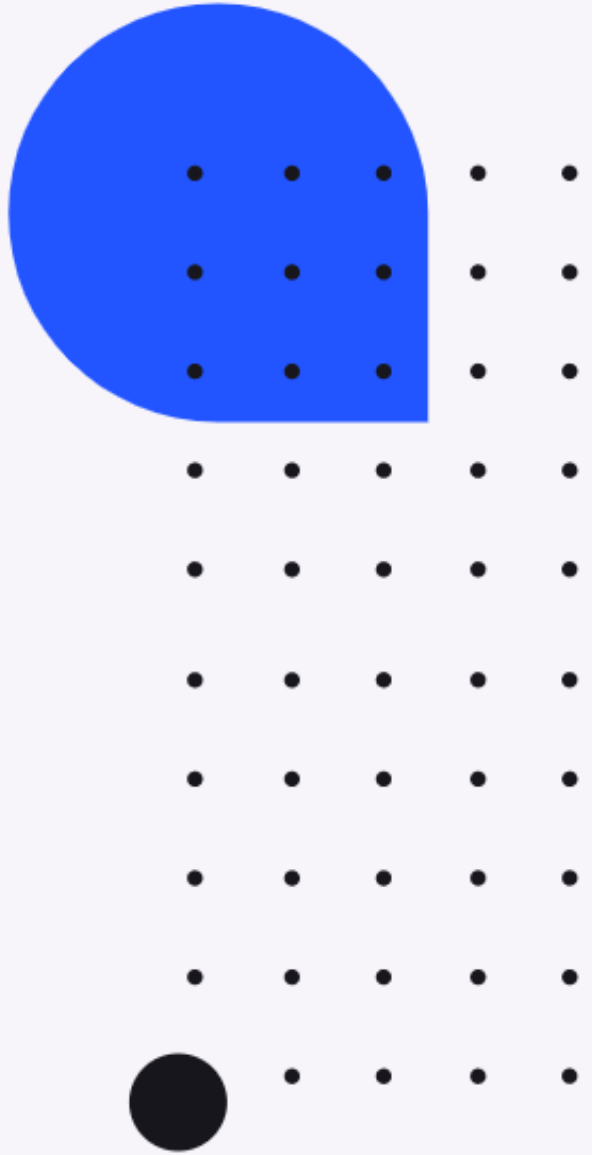
Actor | Use Case | Association | Include | Extend | Generalization



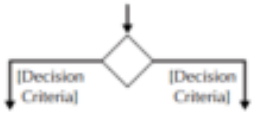
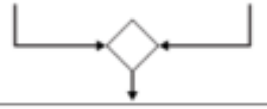
CONTOH USE CASE DIAGRAM

2. ACTIVITY DIAGRAM

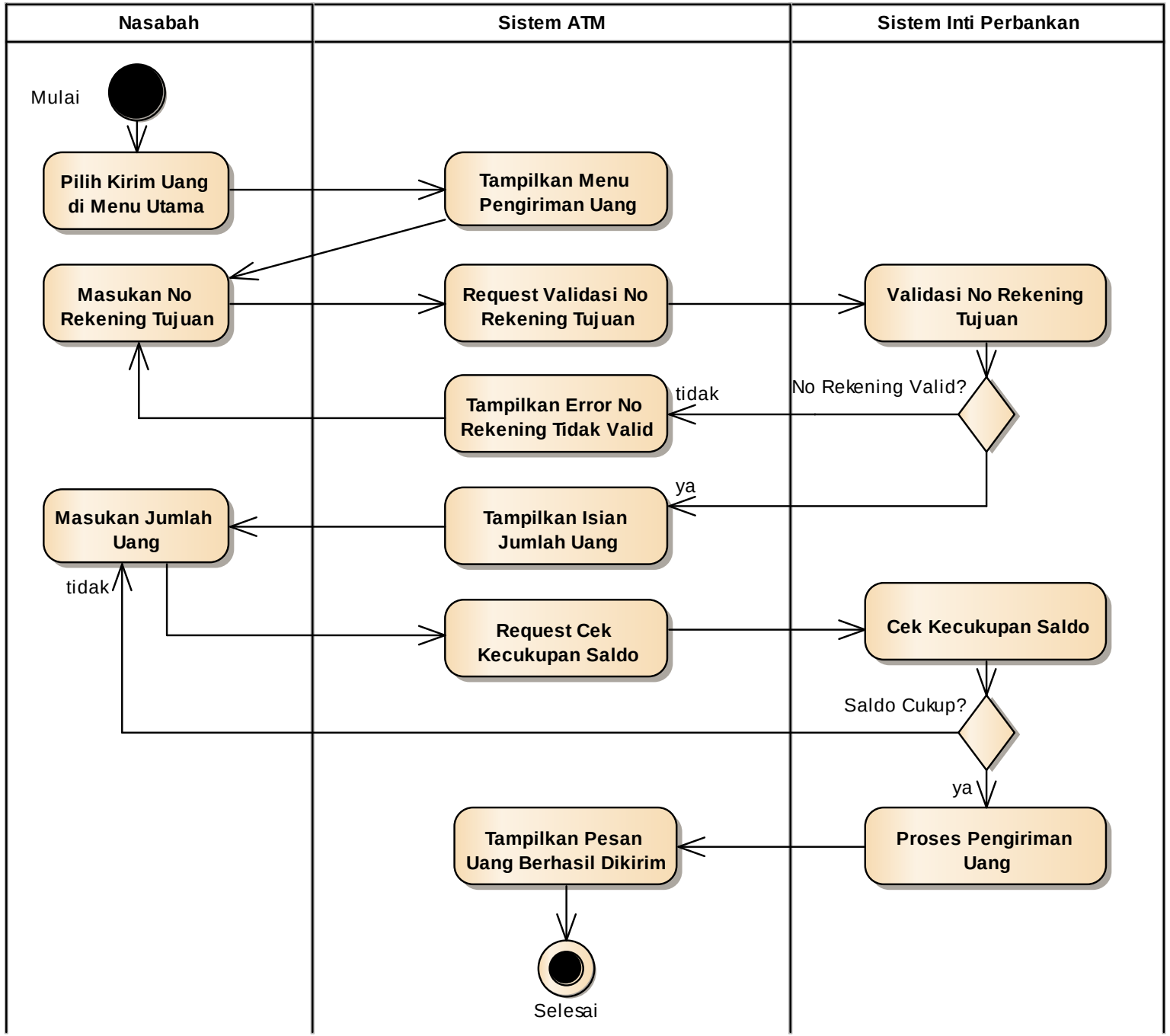
- Merupakan bentuk visualisasi dari alur kerja yang berisi aktivitas dan tindakan, dapat juga berisi pilihan, perulangan dan concurrency.
- Dibuat untuk menjelaskan aktivitas sistem maupun alur aktivitas atau proses bisnis dalam organisasi.



SIMBOL ACTIVITY DIAGRAM

An action: ■ Is a simple, nondecomposable piece of behavior. ■ Is labeled by its name.	
An activity: ■ Is used to represent a set of actions. ■ Is labeled by its name.	
An object node: ■ Is used to represent an object that is connected to a set of object flows. ■ Is labeled by its class name.	
A control flow: ■ Shows the sequence of execution.	
An object flow: ■ Shows the flow of an object from one activity (or action) to another activity (or action).	
An initial node: ■ Portrays the beginning of a set of actions or activities.	
A final-activity node: ■ Is used to stop all control flows and object flows in an activity (or action).	
A final-flow node: ■ Is used to stop a specific control flow or object flow.	
A decision node: ■ Is used to represent a test condition to ensure that the control flow or object flow only goes down one path. ■ Is labeled with the decision criteria to continue down the specific path.	
A merge node: ■ Is used to bring back together different decision paths that were created using a decision node.	
A fork node: Is used to split behavior into a set of parallel or concurrent flows of activities (or actions)	
A join node: Is used to bring back together a set of parallel or concurrent flows of activities (or actions)	
A swimlane: Is used to break up an activity diagram into rows and columns to assign the individual activities (or actions) to the individuals or objects that are responsible for executing the activity (or action) Is labeled with the name of the individual or object responsible	

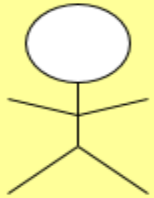
CONTOH ACTIVITY DIAGRAM

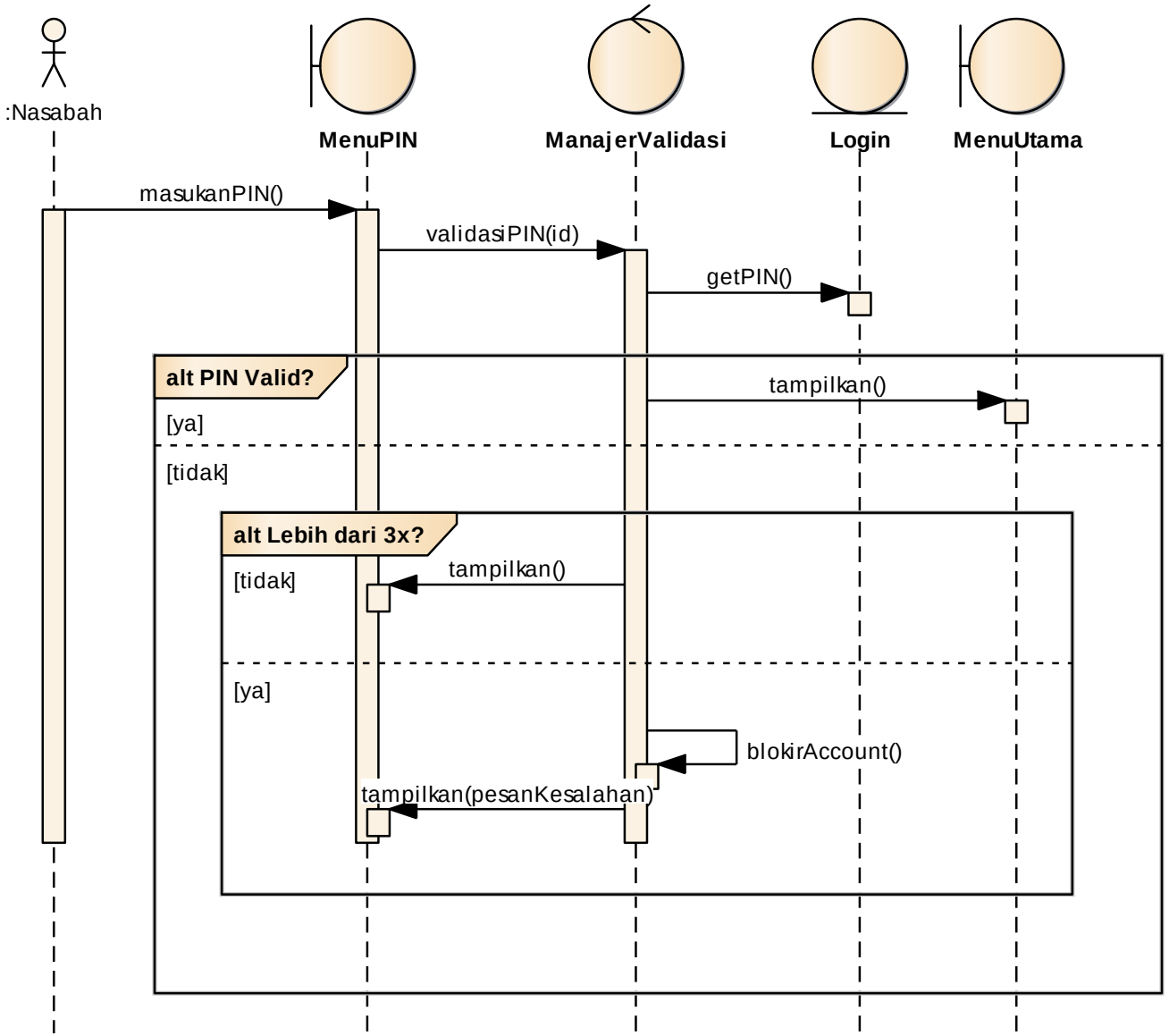


3. SEQUENCE DIAGRAM

- Diagram sequence merupakan salah satu diagram Interaction yang menjelaskan bagaimana suatu operasi itu dilakukan; message (pesan) apa yang dikirim dan kapan pelaksanaannya.
- Objek-objek yang berkaitan dengan proses berjalannya operasi diurutkan dari kiri ke kanan berdasarkan waktu terjadinya dalam pesan yang terurut.
- Sequence diagram adalah alat komunikasi System Analyst dengan Programmer, menggambarkan alur proses bekerjanya software sekaligus dengan komposisi software akan seperti apa.

SIMBOL SEQUENCE DIAGRAM

AN ACTOR	
AN OBJECT	anObject:aClass
A LIFELINE	
A FOCUS OF CONTROL	
A MESSAGE	aMessage()
OBJECT DESTRUCTION	X



CONTOH SEQUENCE DIAGRAM

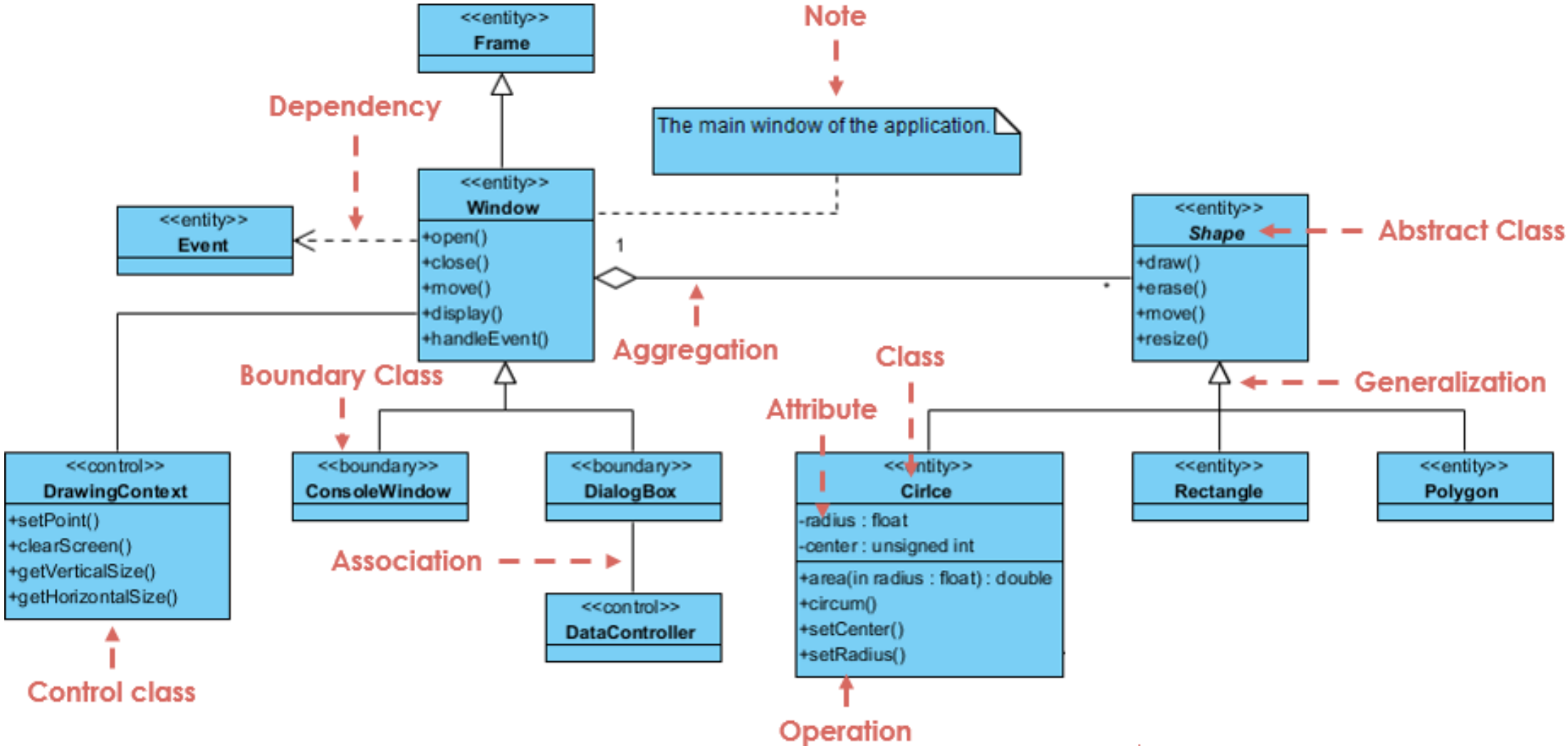
4. CLASS DIAGRAM

- Dalam rekayasa perangkat lunak, diagram kelas dalam Unified Modeling Language (UML) adalah jenis diagram struktur statis yang menggambarkan struktur sistem dengan menunjukkan kelas sistem, atributnya, operasi (atau metode), dan hubungan antar objek.
- Class Diagram terdiri dari satu set class dan seperangkat hubungan antar class
- Satu set class terdiri dari Nama Class, Atribut Class dan Methode Class.

SIMBOL CLASS DIAGRAM

Nama Simbol	Simbol
Class / Kelas	
Asosiasi	
Asosiasi Berarah	
Generalisasi	
Agregasi	
Dependensi / Ketergantungan	

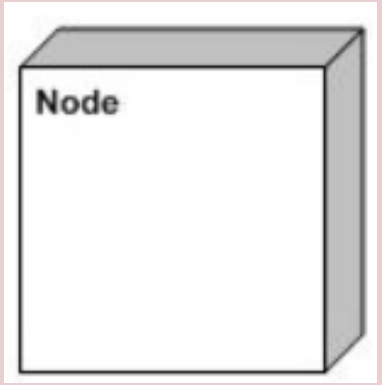
CONTOH CLASS DIAGRAM



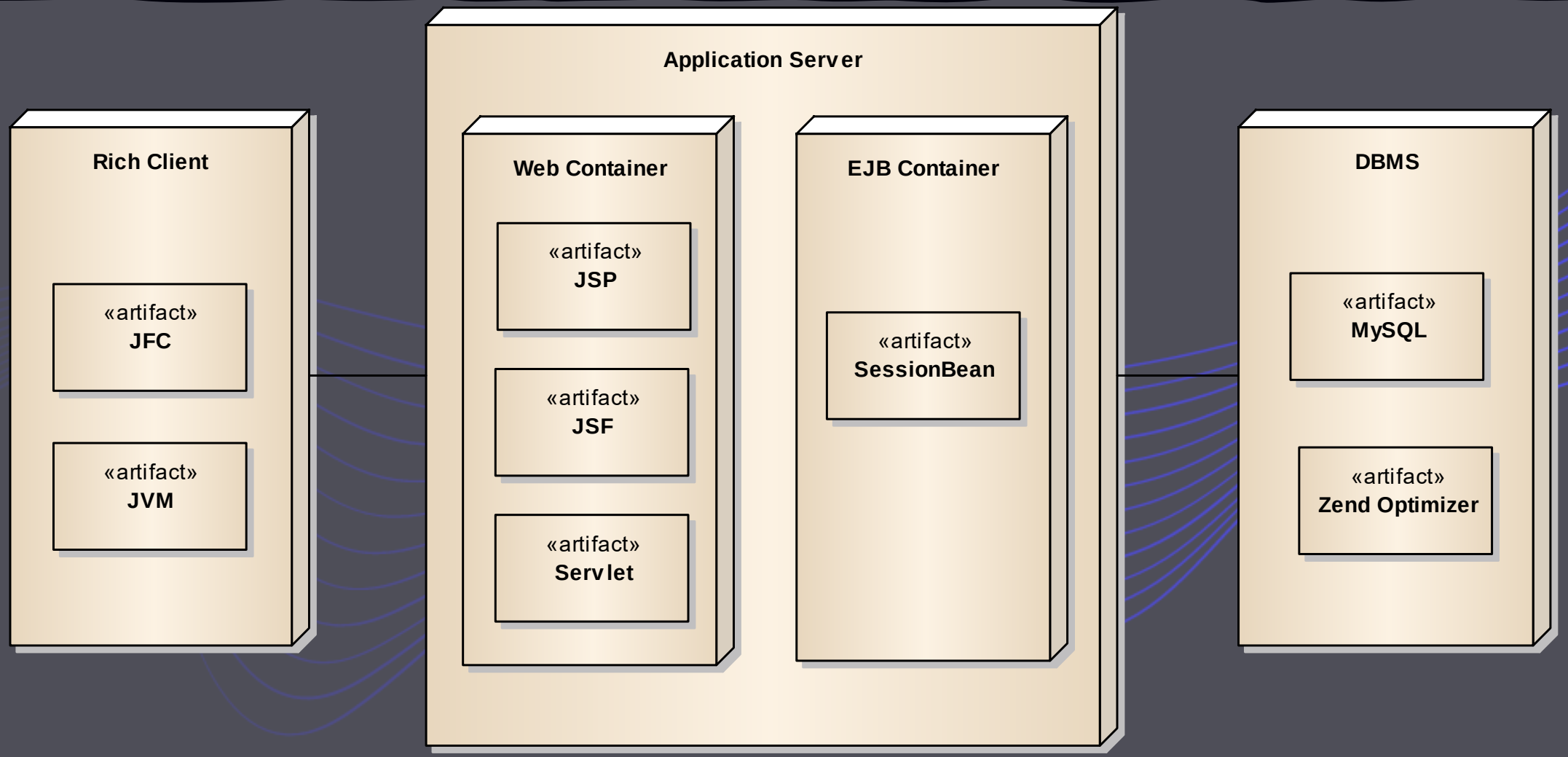
5. DEPLOYMENT DIAGRAM

- Merupakan diagram yang menunjukkan konfigurasi dari proses runtime nodes dan komponen yang terdapat didalamnya.
- Sejenis diagram struktur yang digunakan dalam pemodelan aspek fisik dari sistem berorientasi objek serta pemodelan tampilan penyebaran statis suatu sistem (Topologi Perangkat Keras)

SIMBOL DEPLOYMENT DIAGRAM

Nama Simbol	Simbol
Node	
Dependensi / Ketergantungan	
Link	

CONTOH DEPLOYMENT DIAGRAM



PERANCANGAN UML STUDI KASUS SISTEM ATM

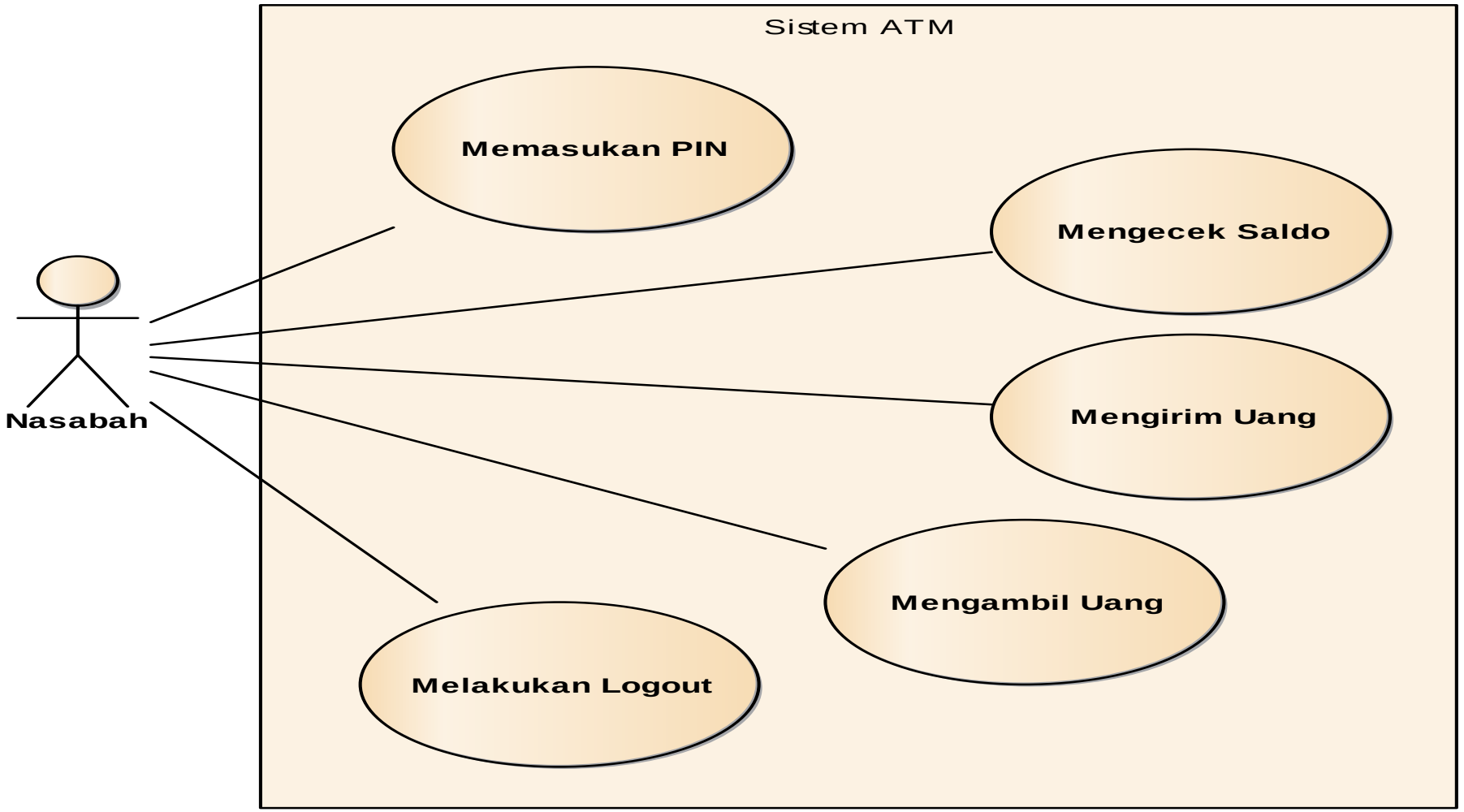
- Analisa Kebutuhan pada Sistem ATM
- Identifikasi Proses Bisnis dengan Use Case Diagram.
- Pemodelan Proses Bisnis dengan Activity Diagram
- Realisasi proses bisnis dengan Sequence Diagram
- Perancangan Class Diagram
- Perancangan Desain Data Model
- Perancangan Deployment Diagram

ANALISA KEBUTUHAN PADA SISTEM ATM

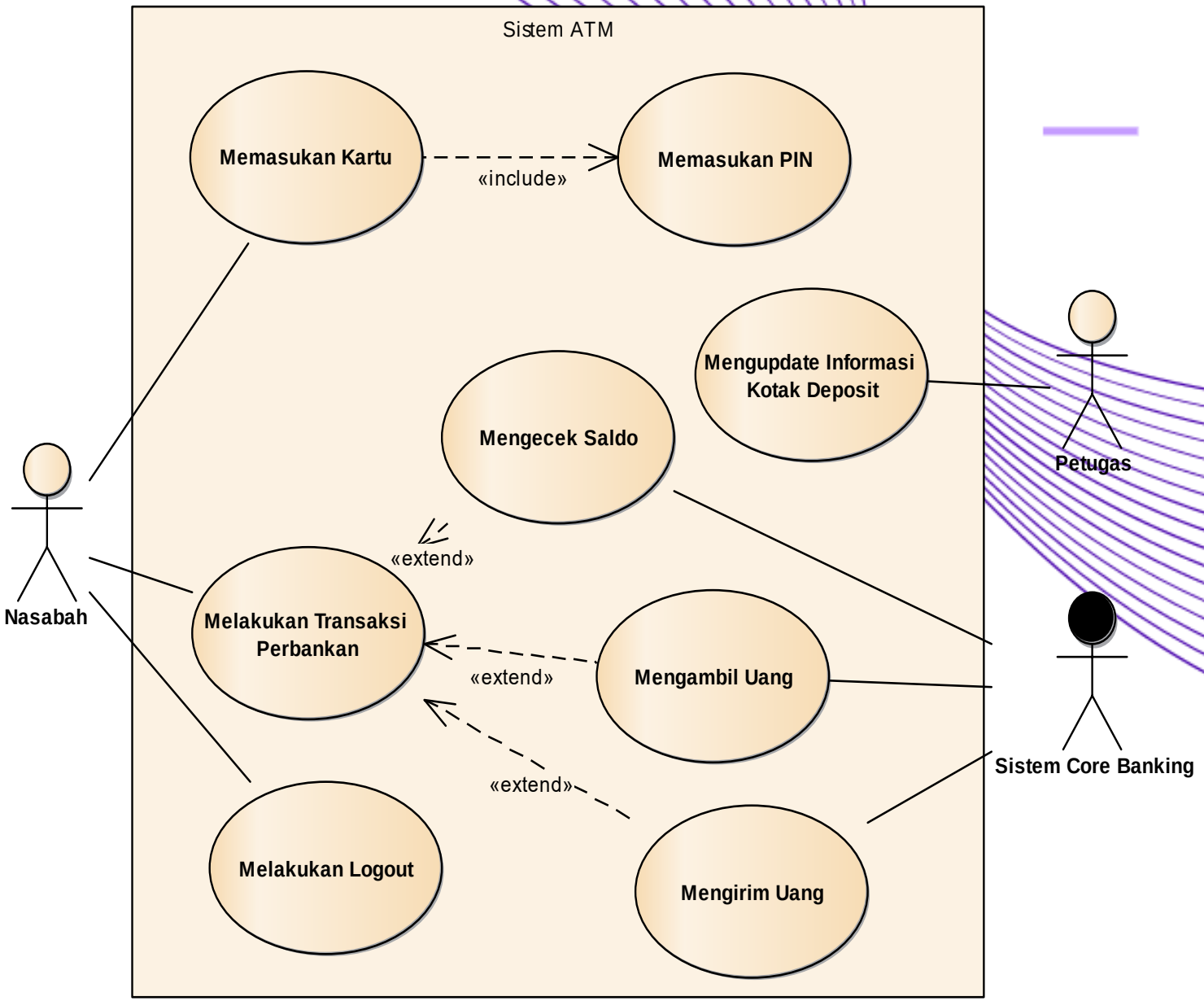
- Memasukan PIN
- Mengecek Saldo
- Mengirim Uang
- Mengambil Uang
- Melakukan LogOut

IDENTIFIKASI PROSES BISNIS DENGAN USE CASE DIAGRAM.

USE CASE DIAGRAM VERSI SEDERHANA

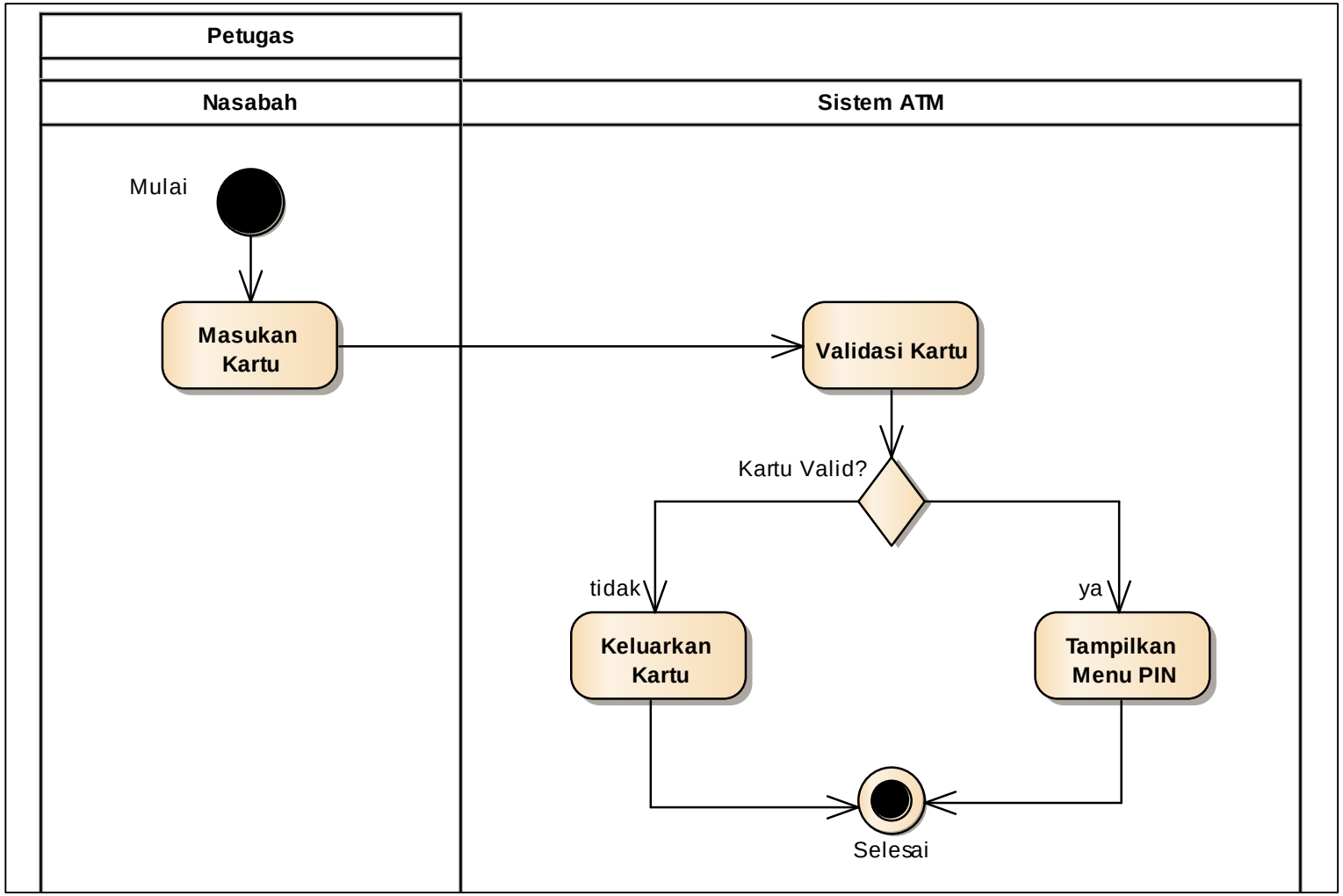


USE CASE DIAGRAM. (DENGAN INCLUDE DAN EXTEND)

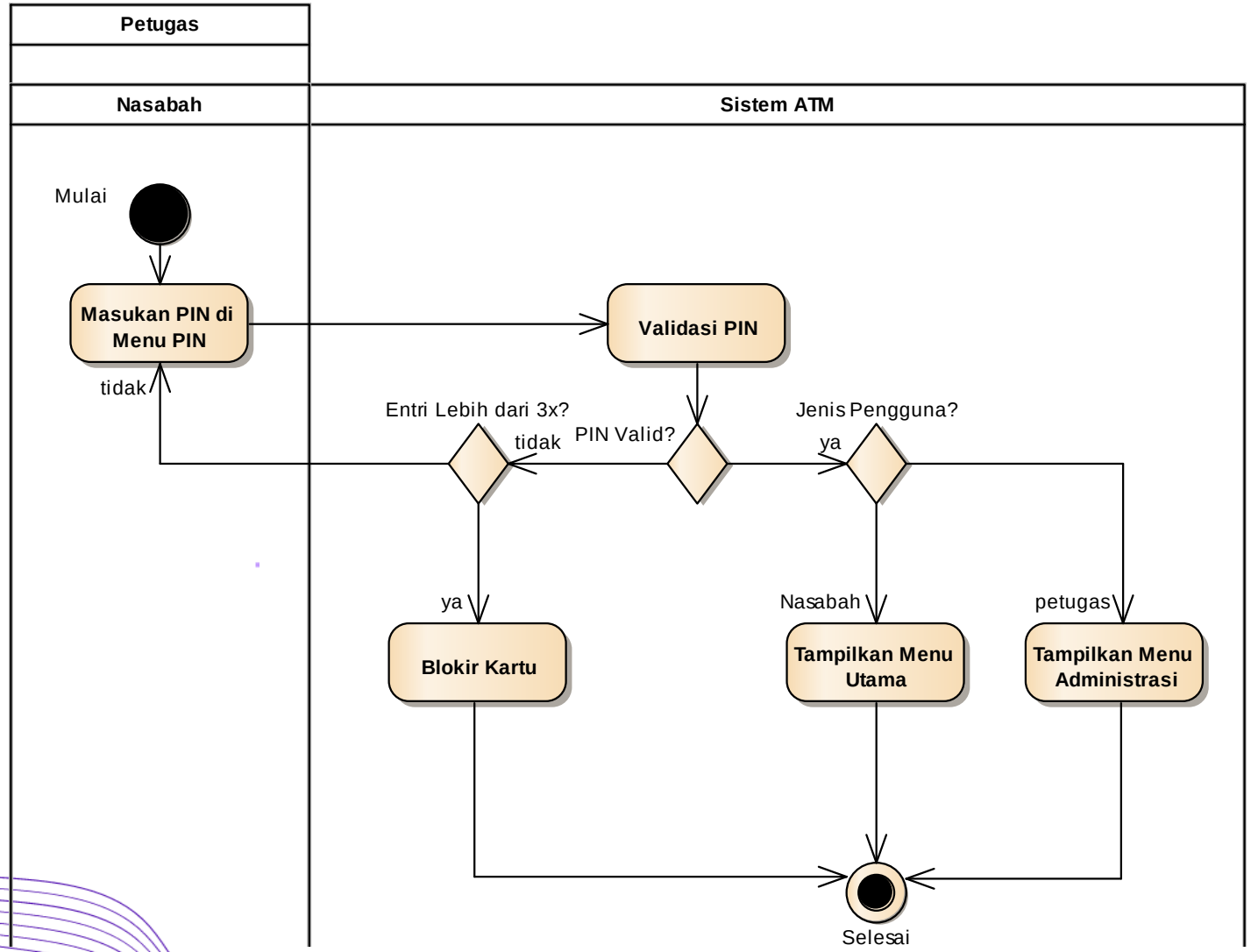


PEMODELAN PROSES BISNIS DENGAN ACTIVITY DIAGRAM

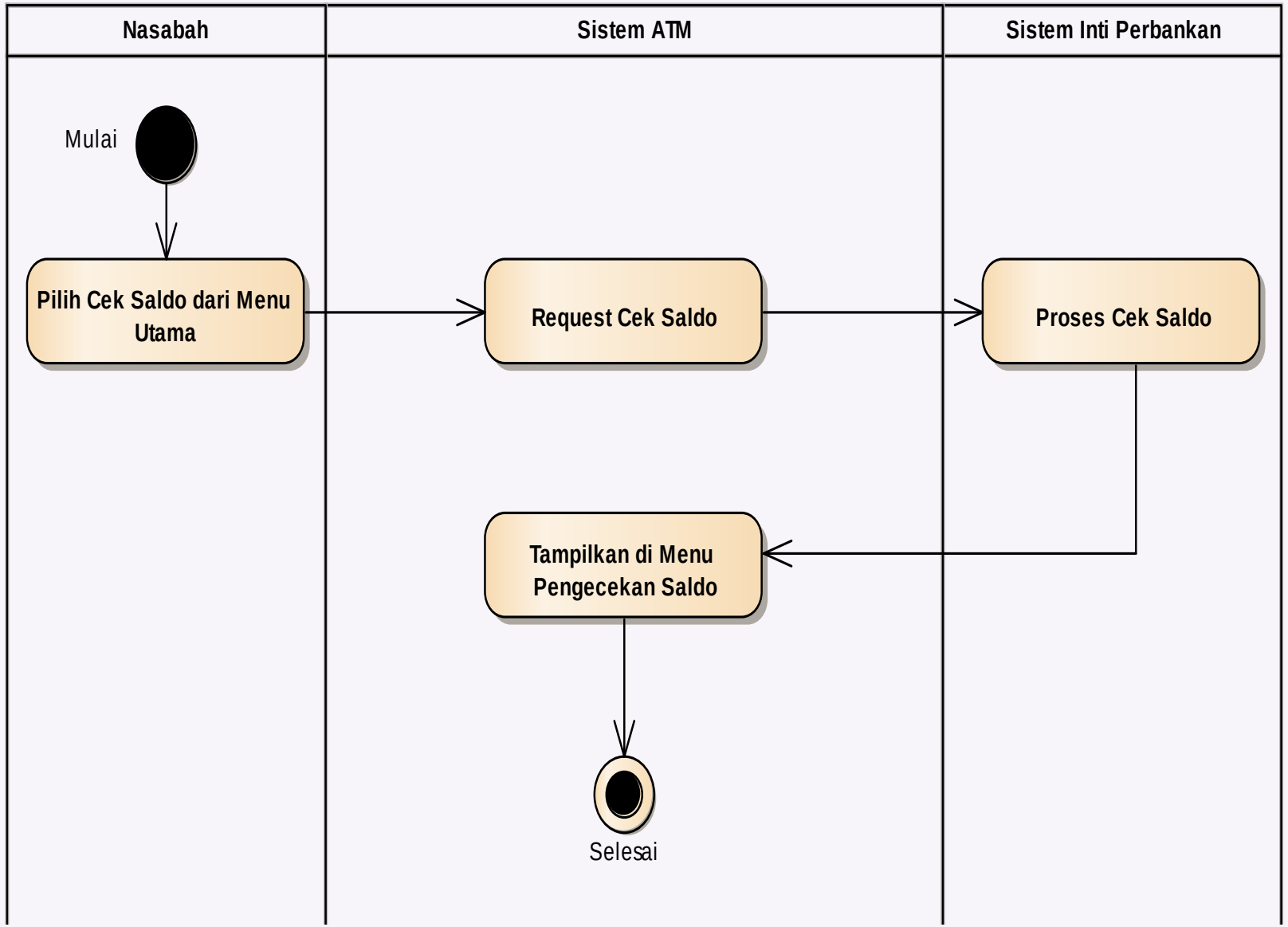
ACTIVITY DIAGRAM : MEMASUKKAN KARTU



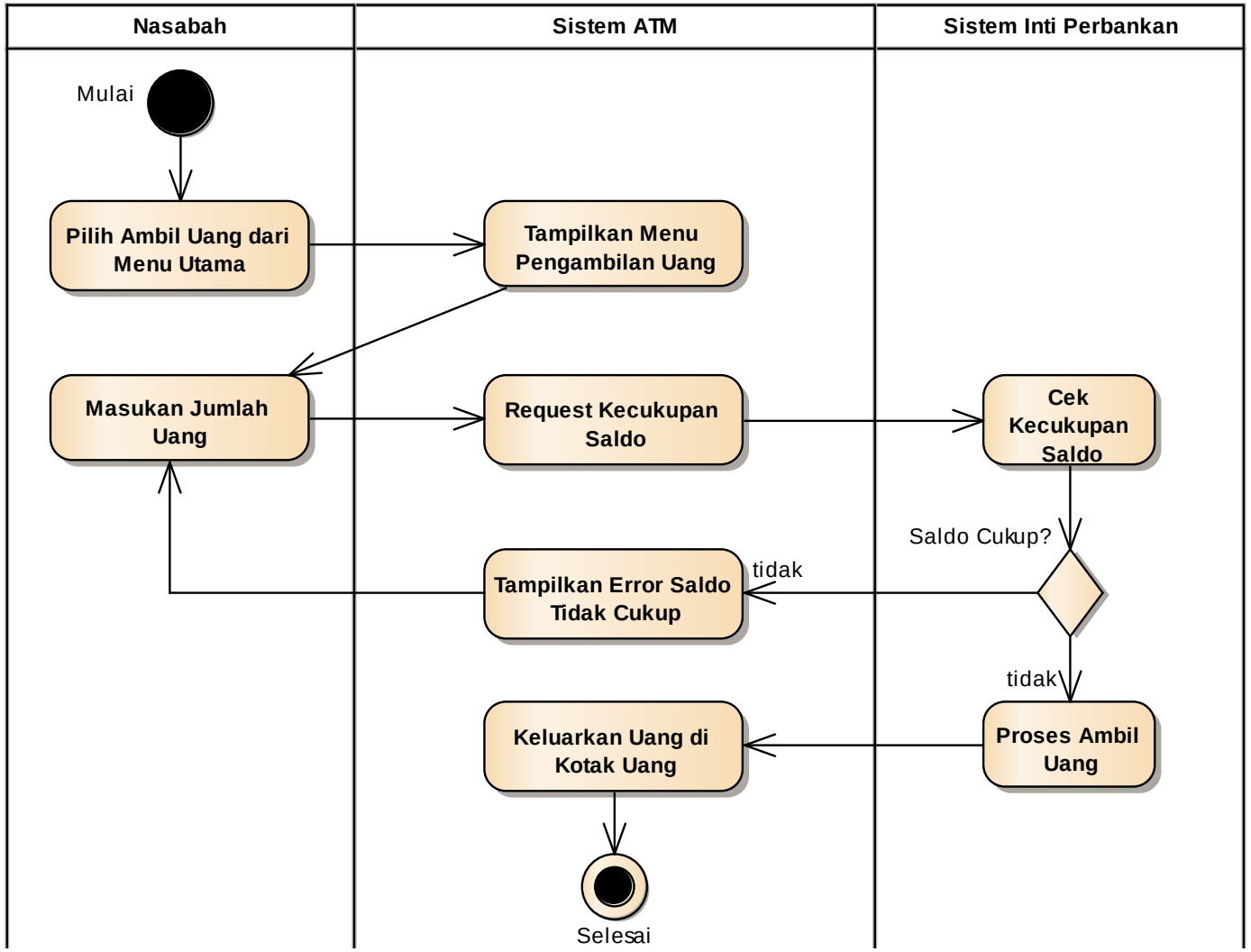
ACTIVITY DIAGRAM : MEMASUKKAN PIN

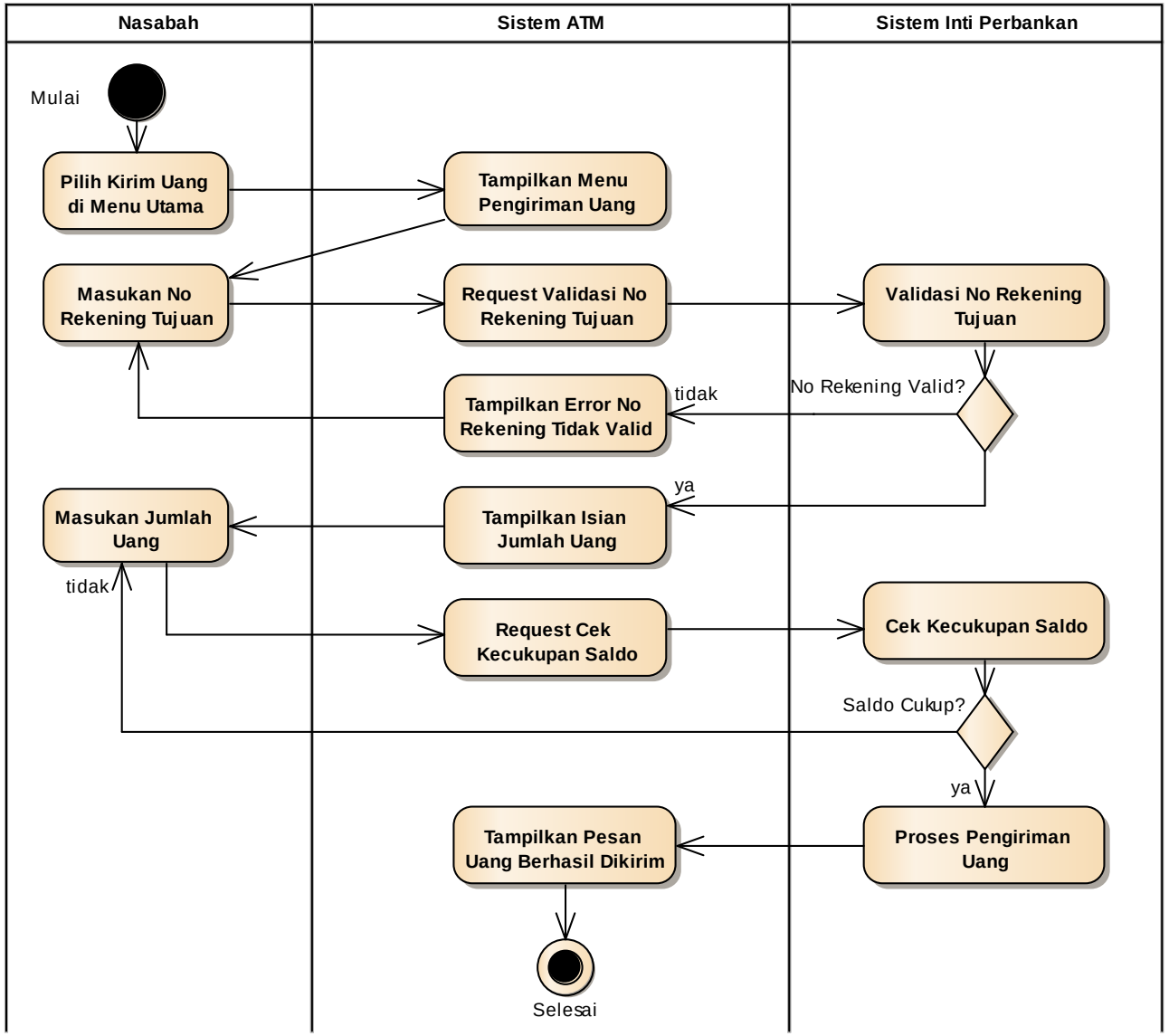


ACTIVITY DIAGRAM: MENGECEK SALDO

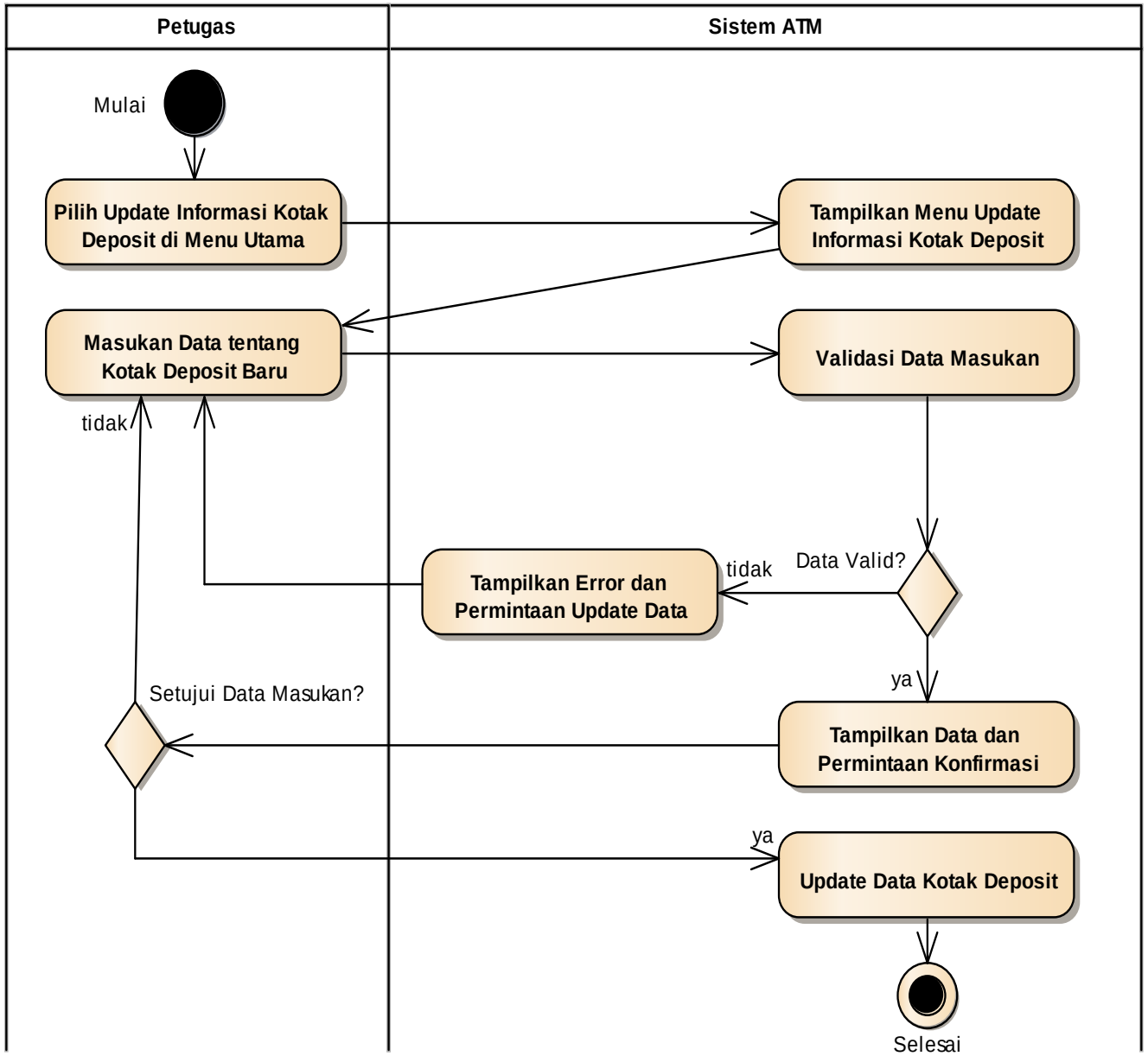


ACTIVITY DIAGRAM: MENGAMBIL UANG



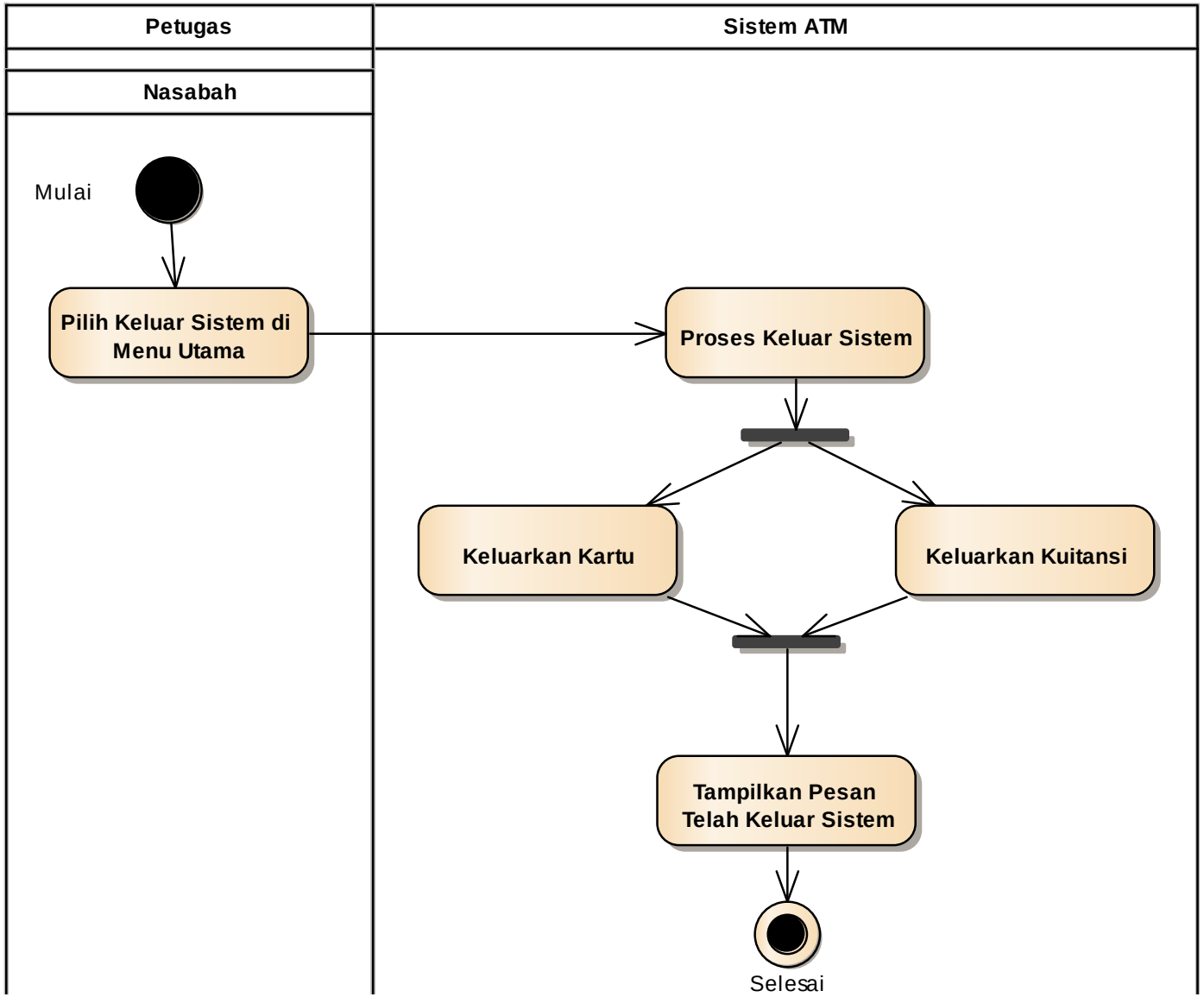


ACTIVITY DIAGRAM: MENGIRIM UANG



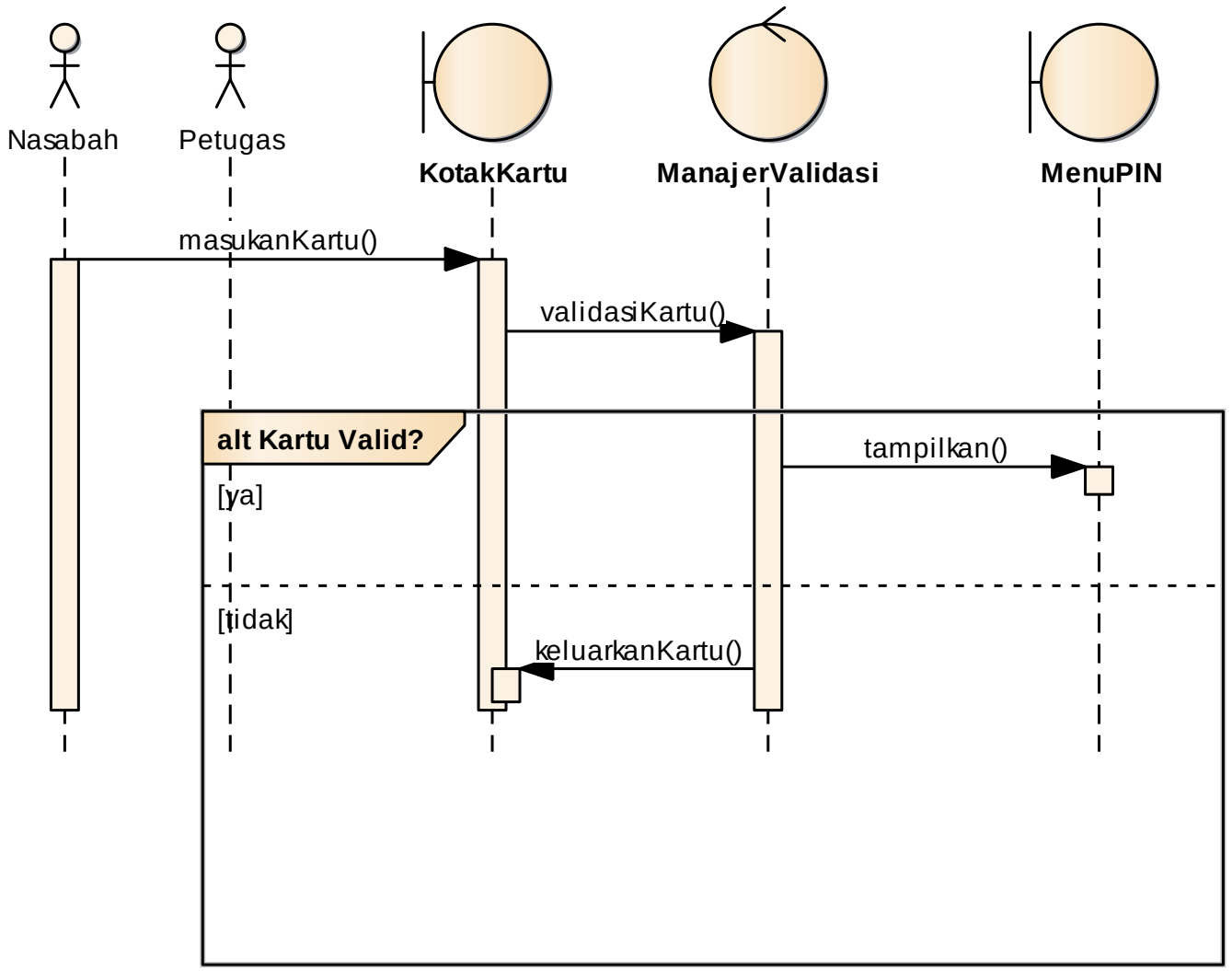
ACTIVITY DIAGRAM: MENGUPDATE INFORMASI KOTAK DEPOSIT

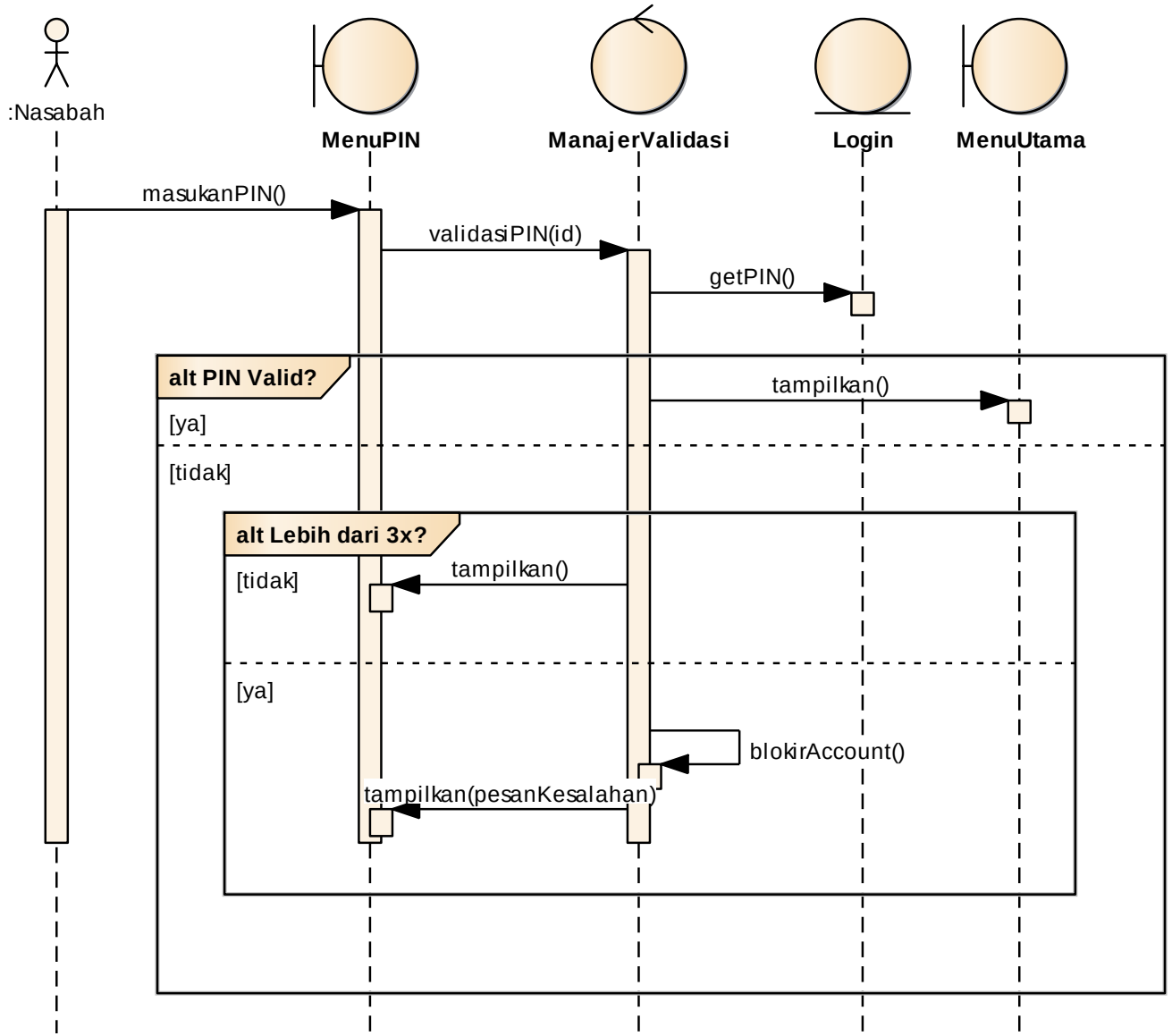
ACTIVITY DIAGRAM: KELUAR SISTEM



REALISASI PROSES BISNIS DENGAN SEQUENCE DIAGRAM

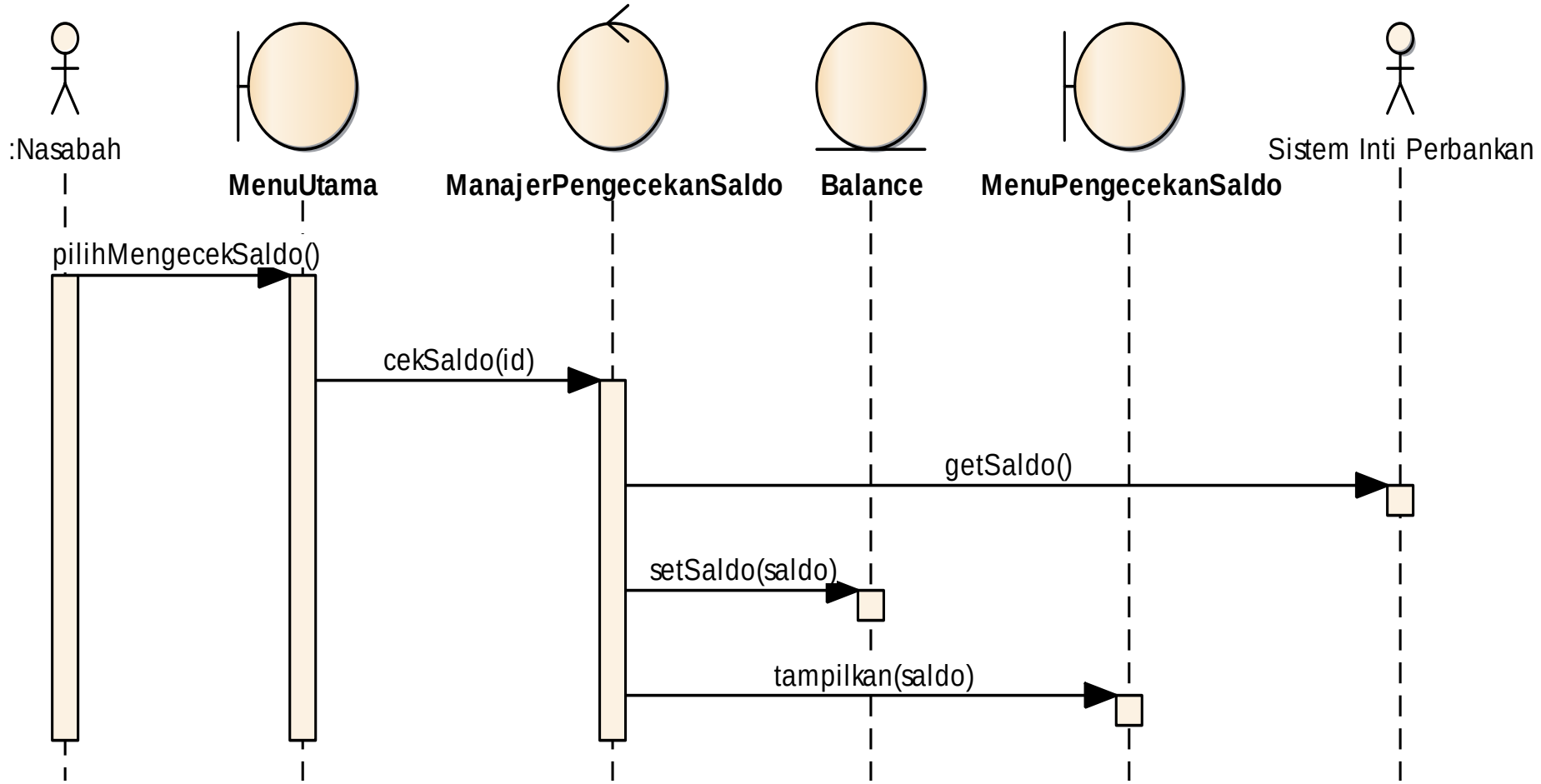
SEQUENCE DIAGRAM: MEMASUKKAN KARTU



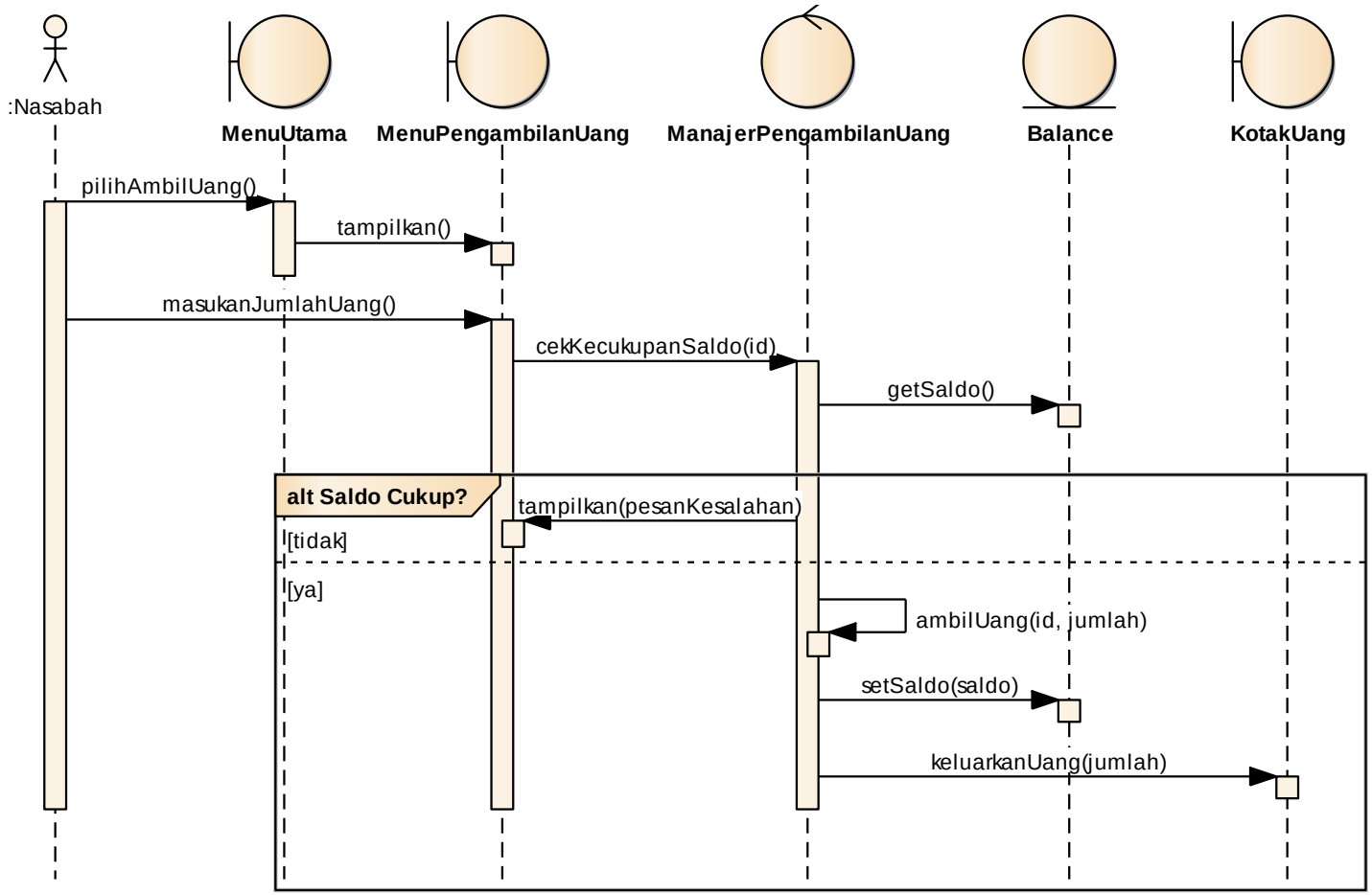


SEQUENCE DIAGRAM: MEMASUKKAN PIN

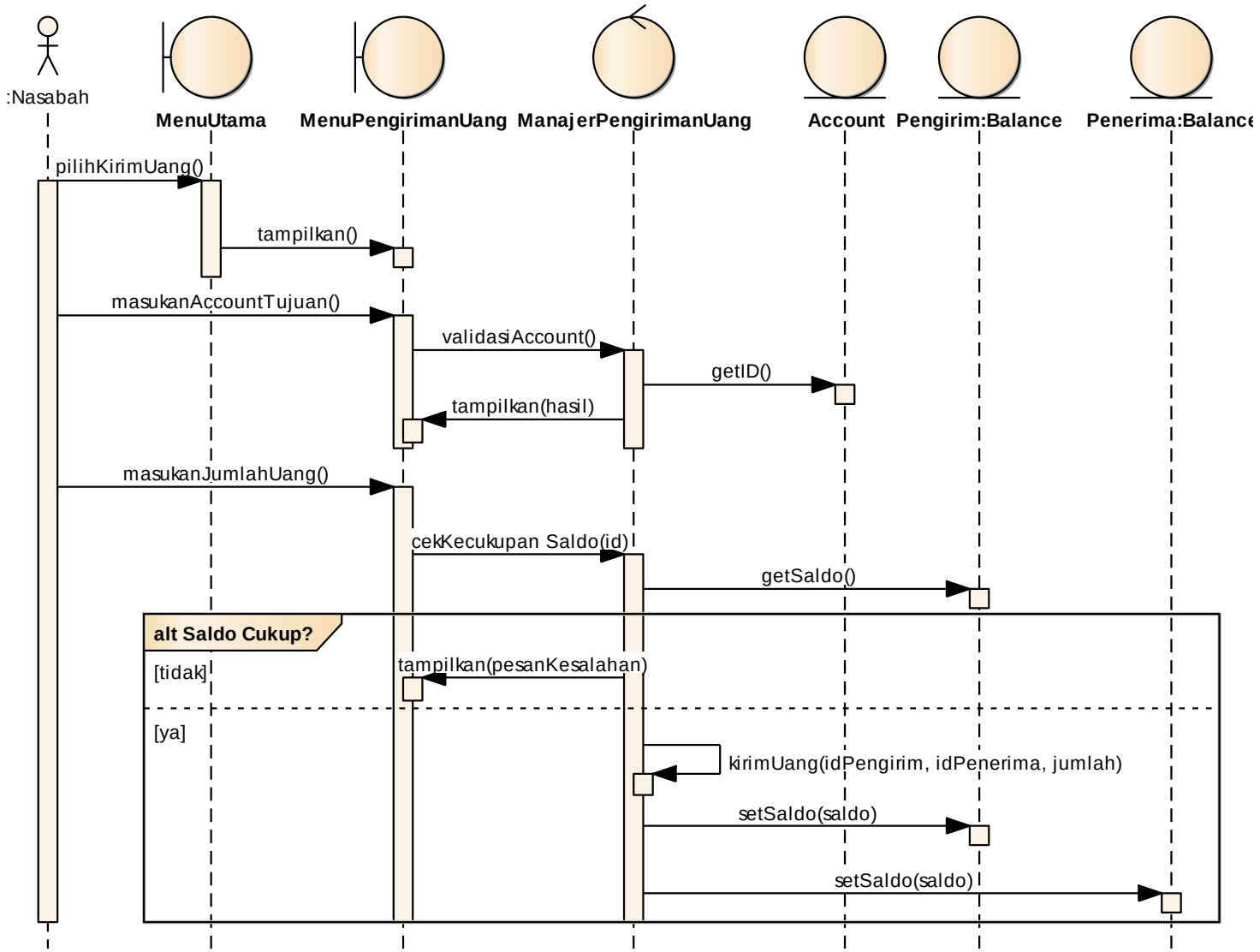
SEQUENCE DIAGRAM: MENGECEK SALDO

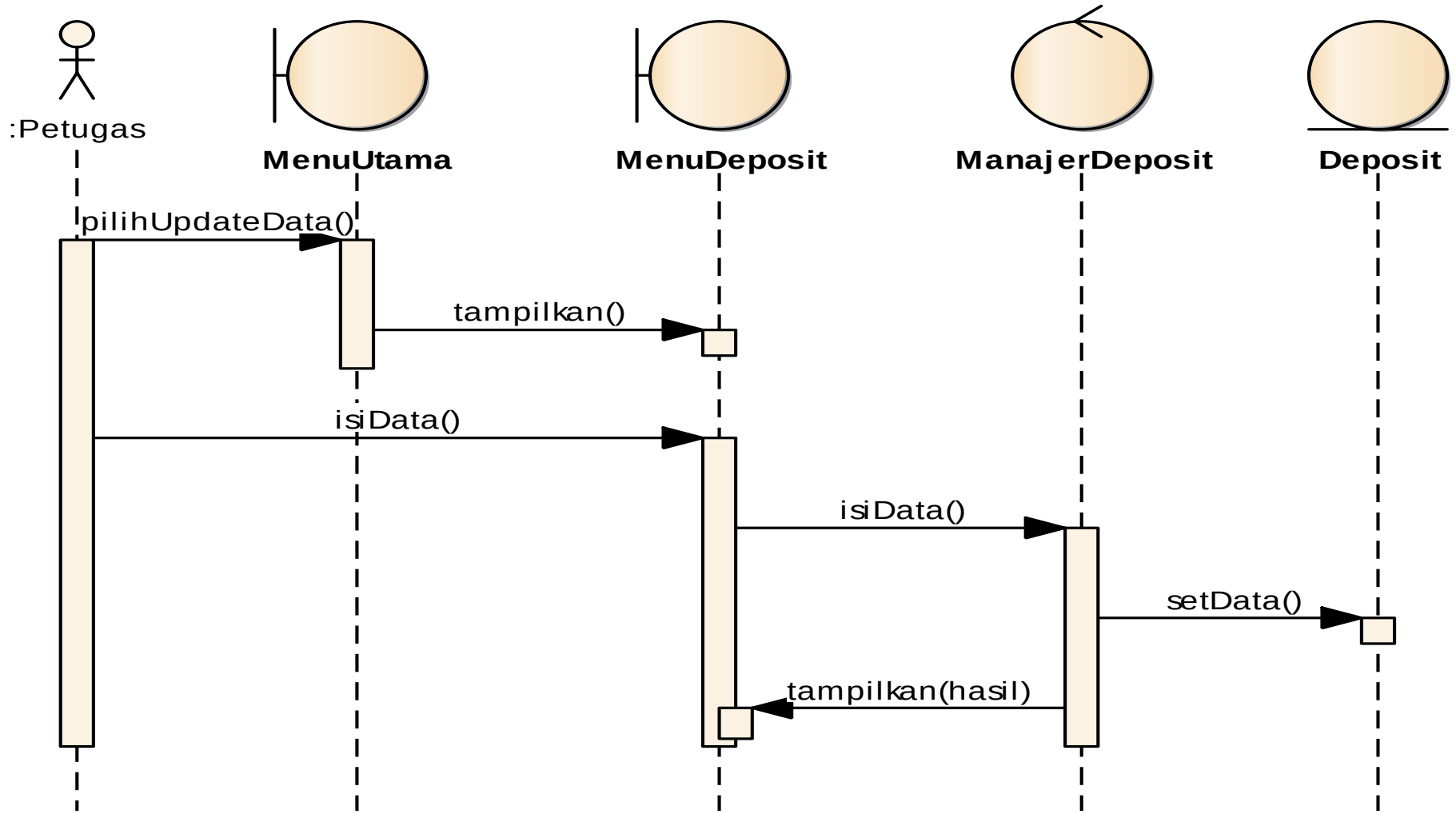


SEQUENCE DIAGRAM: MENGAMBIL UANG



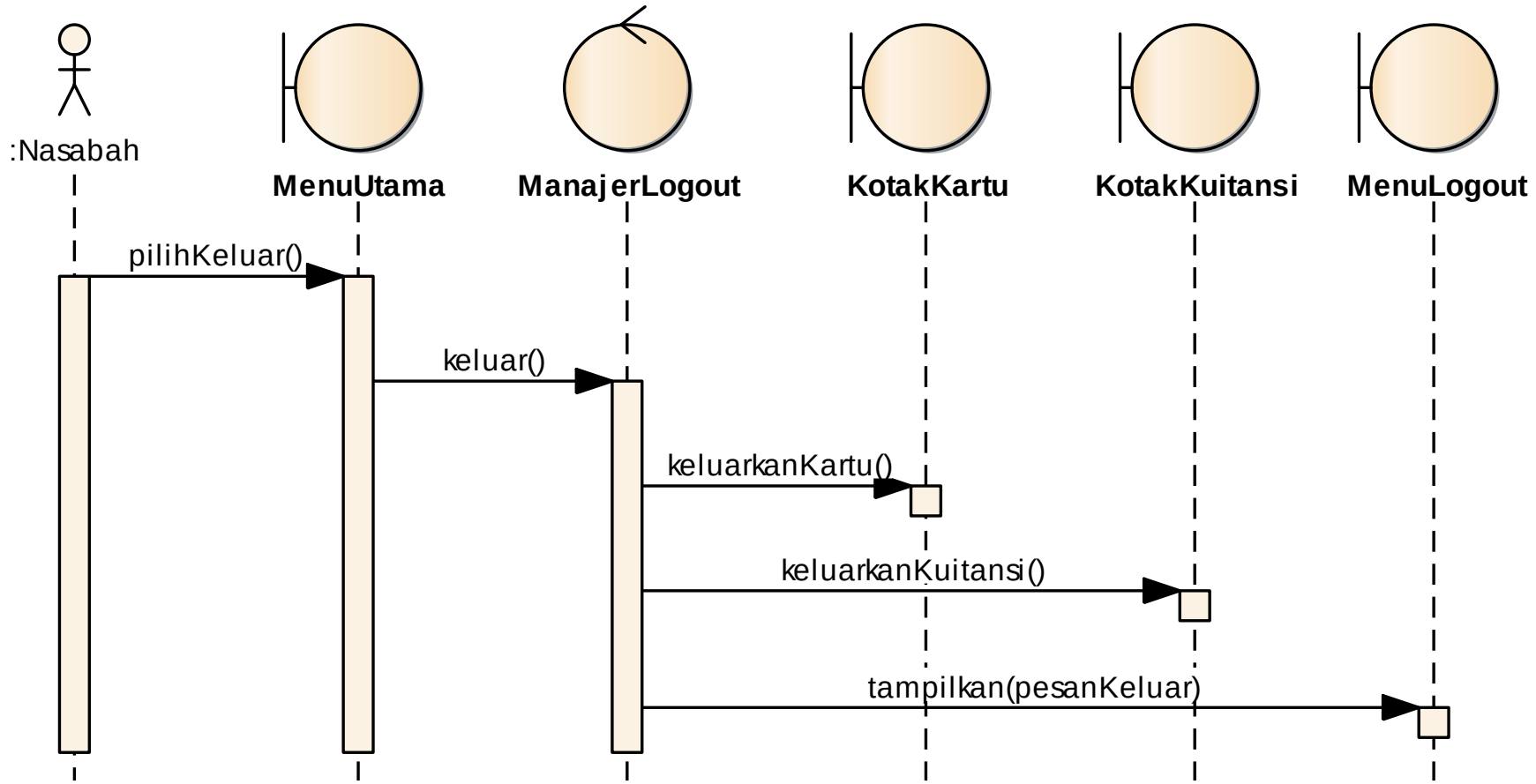
SEQUENCE DIAGRAM: MENGIRIM UANG





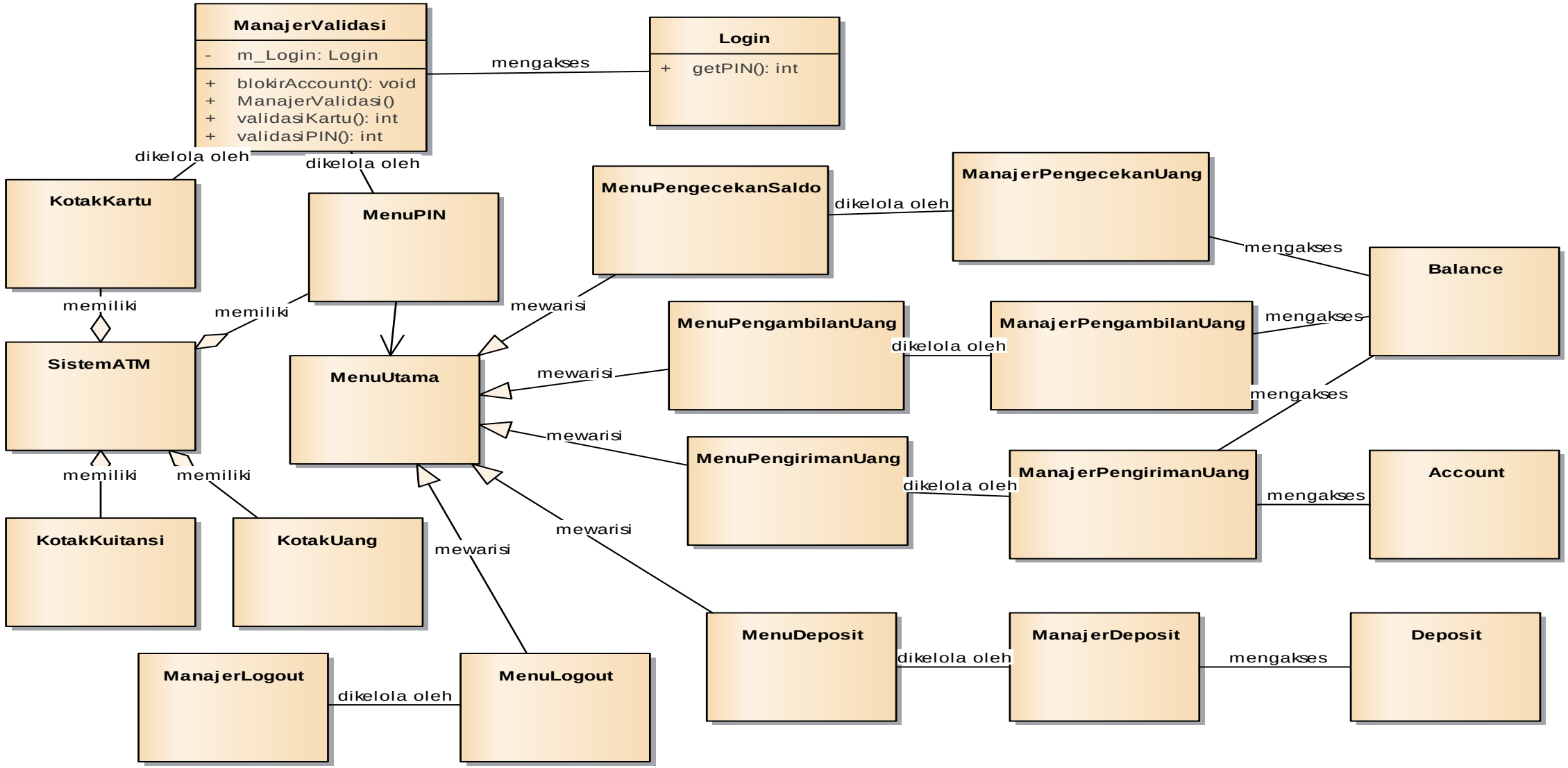
SEQUENCE DIAGRAM: MENGUPDATE INFORMASI KOTAK DEPOSIT

SEQUENCE DIAGRAM: KELUAR SISTEM



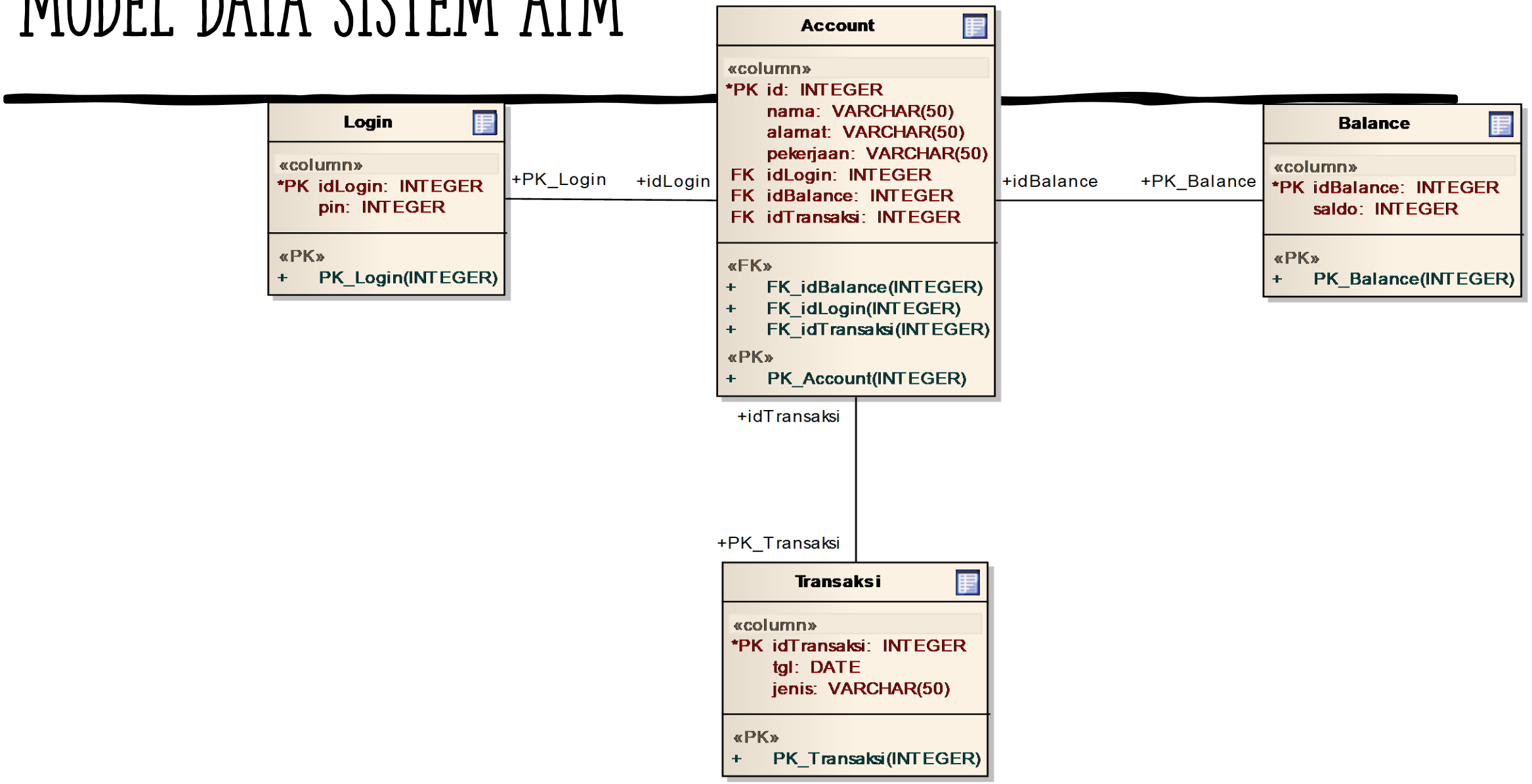
PERANCANGAN CLASS DIAGRAM

CLASS DIAGRAM SISTEM ATM



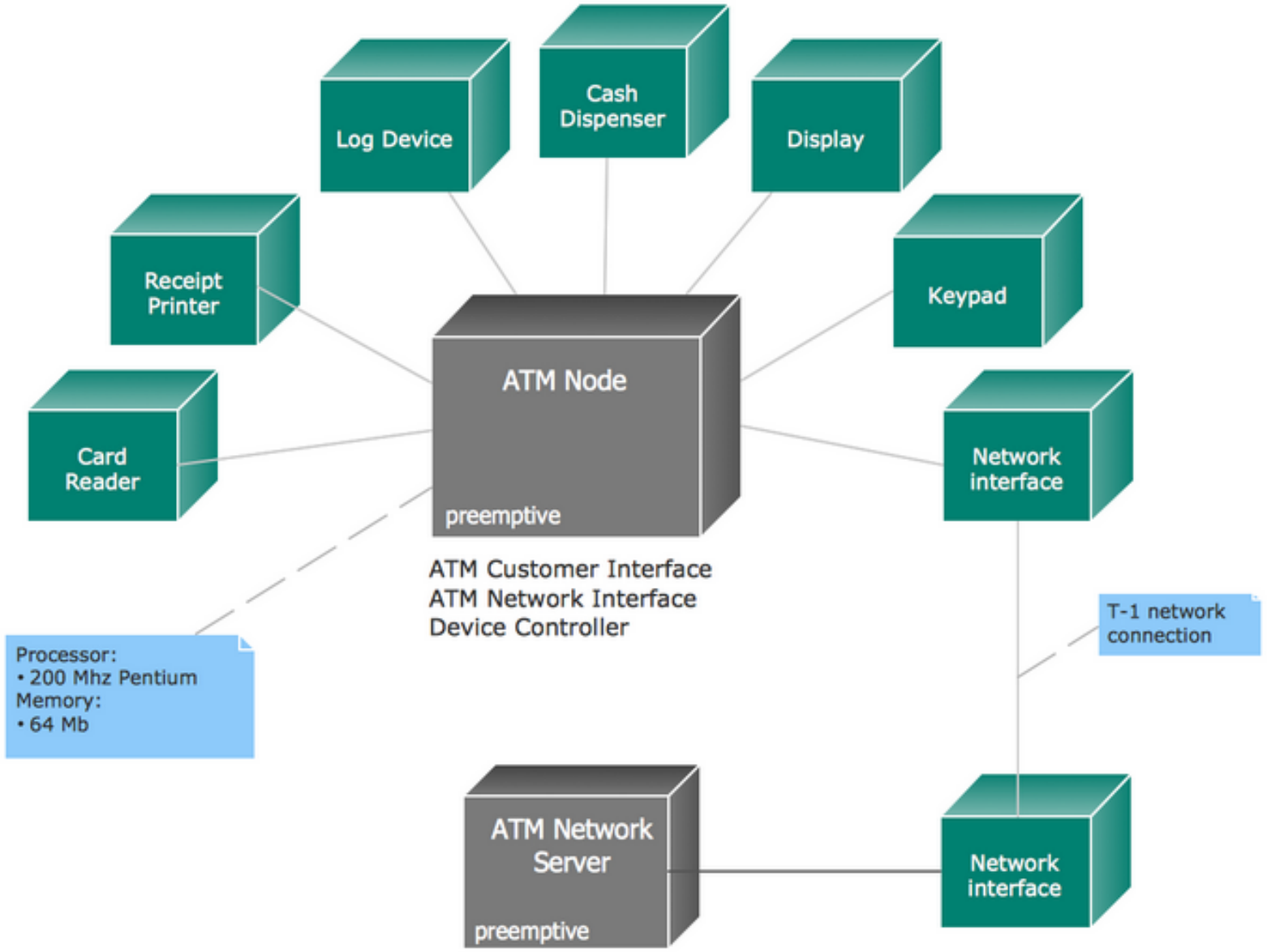
PERANCANGAN DESAIN DATA MODEL

MODEL DATA SISTEM ATM



PERANCANGAN DEPLOYMENT DIAGRAM

DEPLOYMENT DIAGRAM SYSTEM ATM



RUJUKAN

- https://romisatriawahono.net/sad/
- https://www.youtube.com/watch?v=THv-aaLQg04
- https://www.visual-paradigm.com/guide/uml-unified-modeling-language
- https://proyekjava.wordpress.com/tag/sistem-atm-automated-teller-machine/

BE FIRST
BE UNIQUE
BE DIFFERENT