

# Introduction to UML Part 1

What is UML?

Hello

你好

Kaixo

こんにちは

নমস্কার

Hola

Алë

Sveiki

Halló

salve

chào

안녕하세요.

Ciao

ନମ୍ବାଣିଶଦା

a

Bonjour

Olá

cześć

ahoj

Γεια σας

Алло

Здравейте

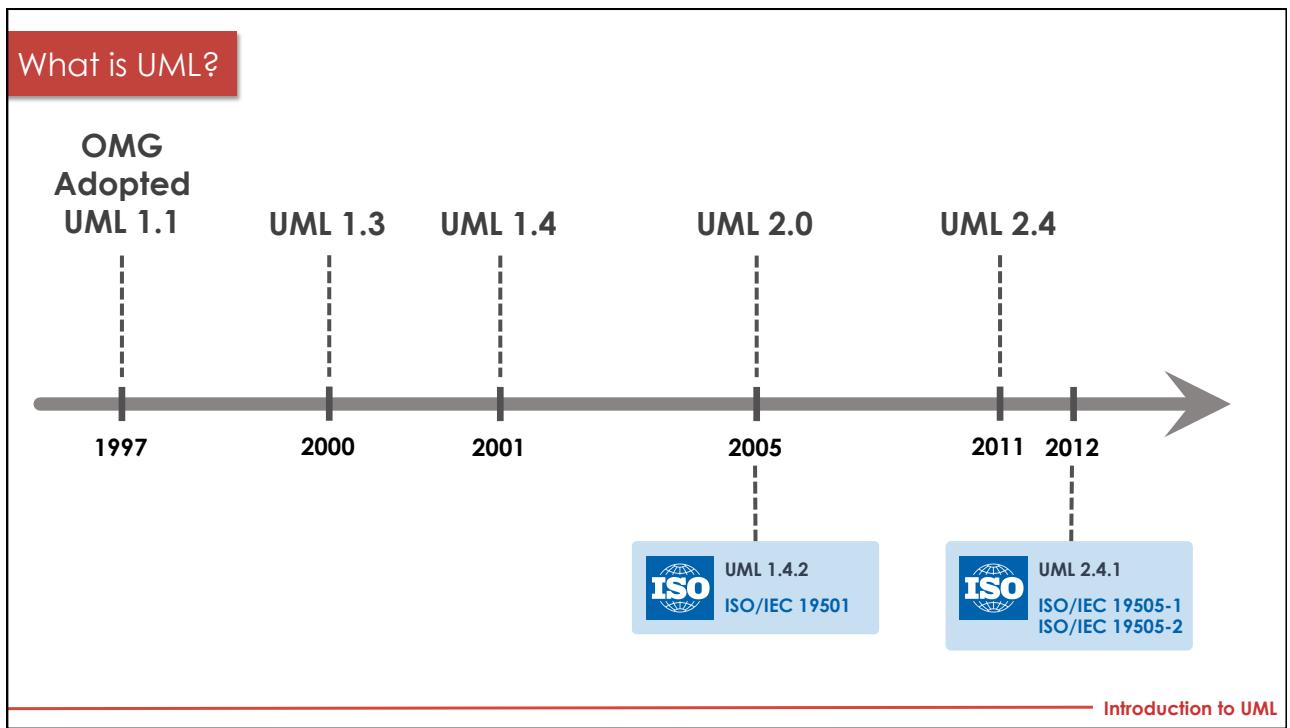
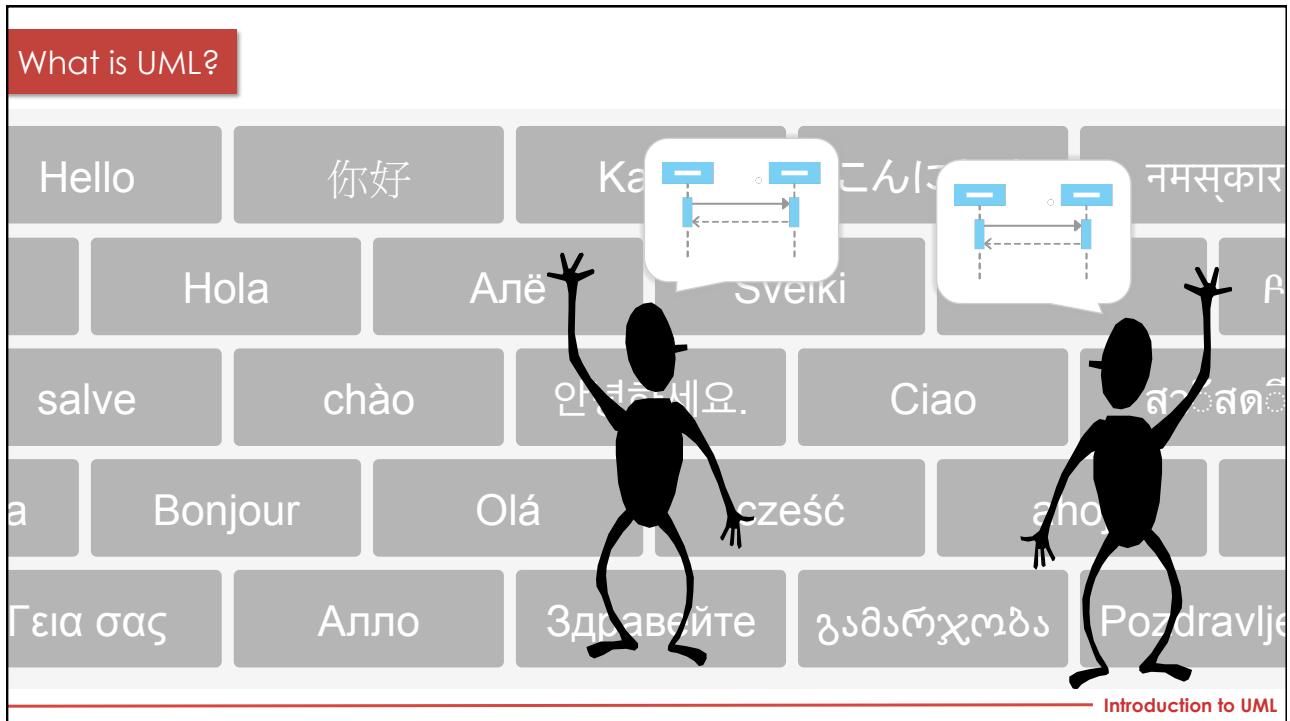
გამარჯობა

Pozdravlje



Introduction to UML

## Introduction to UML Part 1



## Introduction to UML Part 1

Why UML?



Introduction to UML

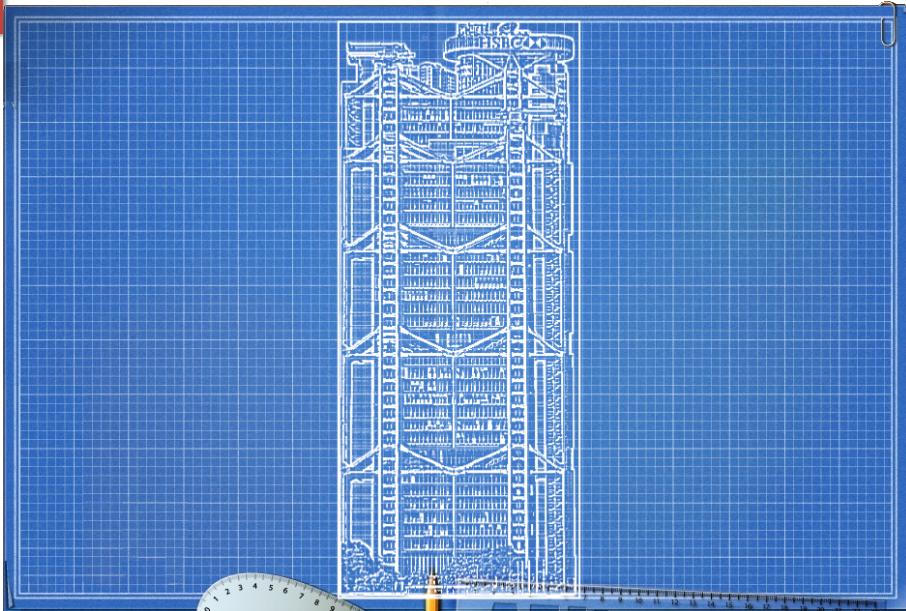
Why UML?



Introduction to UML

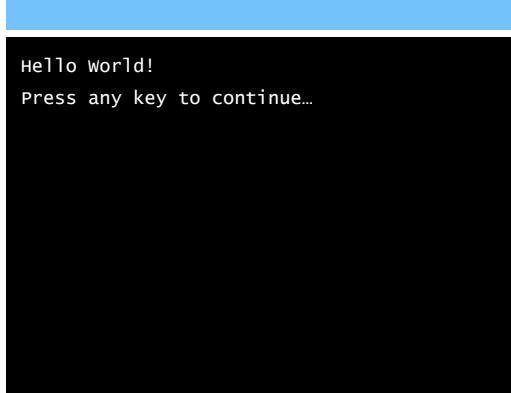
## Introduction to UML Part 1

Why UML?



Introduction to UML

Why UML?



Introduction to UML

## Introduction to UML Part 1

Why UML?

The image consists of two main parts. On the left, there is a screenshot of a Facebook profile for 'Mark Zuckerberg'. The profile picture is of a young Mark Zuckerberg smiling. The bio text includes: 'Has worked at Facebook since 2004', 'Lives in Palo Alto, California', 'Studied Computer Science at Harvard University', 'From Dobbs Ferry, New York', and 'Born on May 14, 1984'. Below the bio are several small thumbnail images of him. On the right, there is a 3x3 grid of nine identical illustrations. Each illustration shows a man with dark skin and short hair, wearing a dark suit and tie, sitting at a desk and working on a computer. The illustrations are arranged in three rows and three columns.

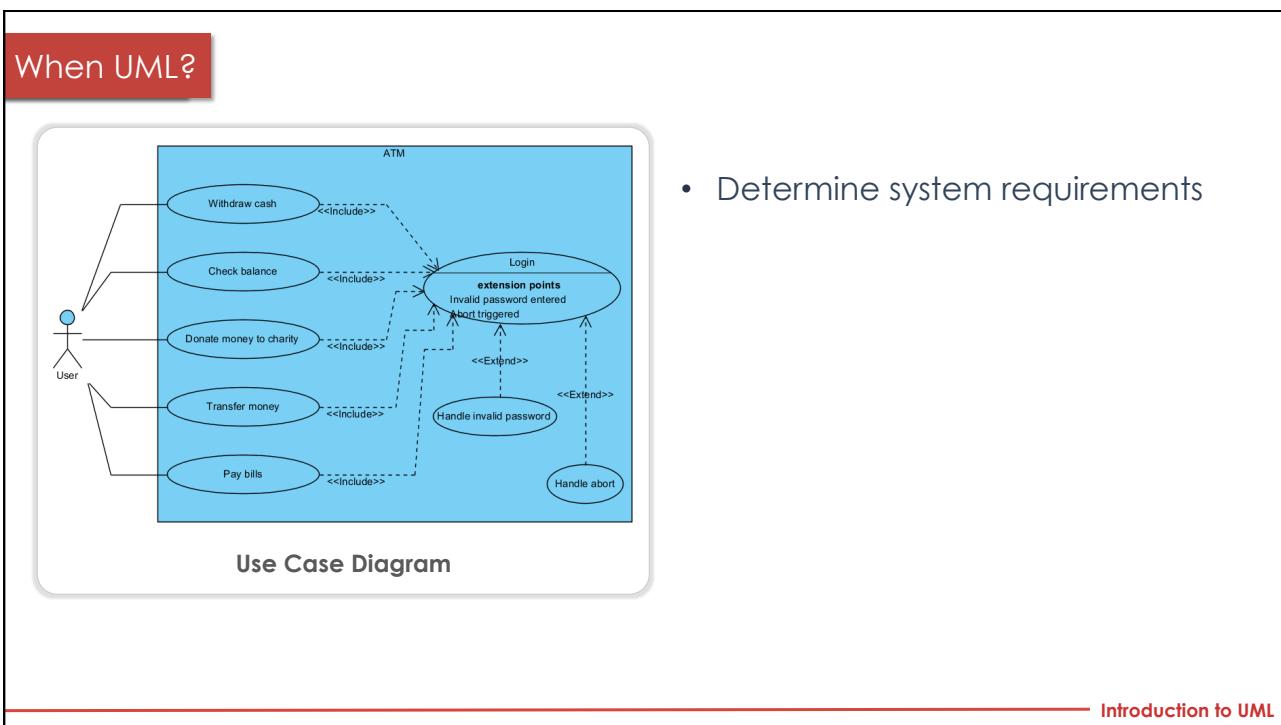
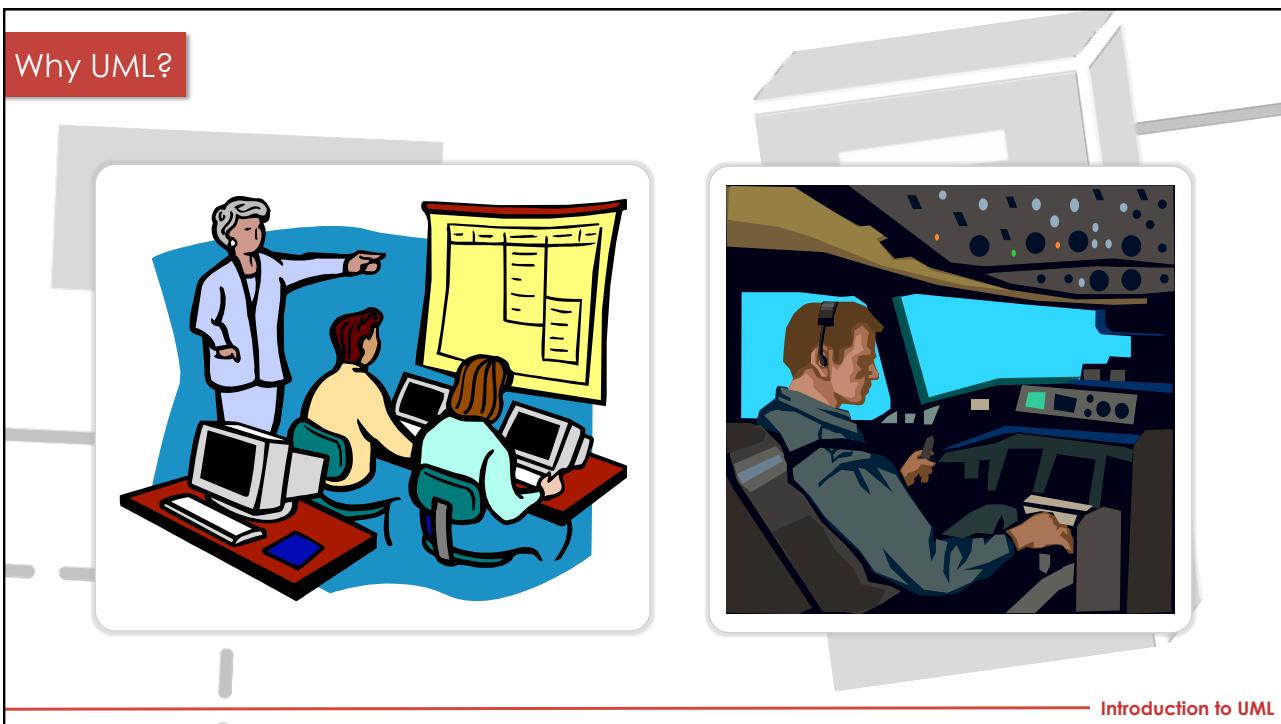
Introduction to UML

Why UML?

The image shows three UML-related diagrams. At the top is a class diagram with a generalization relationship. It features a blue rounded rectangle at the top representing a general class, with two smaller blue rectangles below it representing specific subclasses. An upward-pointing arrow connects the general class to the specific ones. Below this are two more diagrams: one on the left showing a sequence diagram with lifelines and messages, and one on the right showing a state transition diagram with states and transitions. To the right of these diagrams is the 'UNIFIED MODELING LANGUAGE' logo, which consists of the words 'UNIFIED MODELING LANGUAGE' in a sans-serif font next to a stylized 'UML' where the 'U' is red and the 'ML' is purple.

Introduction to UML

## Introduction to UML Part 1



## Introduction to UML Part 1

### When UML?

The diagram illustrates a UML Class Diagram divided into three main horizontal sections: payment, writing, and presentation.

- payment:** Contains classes `PayController` and `Transaction`. `PayController` has methods `pay(writer : Writer)` and `recordPayout(writer : Writer)`. `Transaction` has attributes `writer : Writer` and `date : Date`.
- writing:** Contains classes `Writer`, `Article`, and `Category`. `Writer` has attributes `loginID : String`, `name : String`, `age : int`, `country : String`, and `remarks : String`. `Article` has attributes `title : String`, `author : String`, `body : String`, `dom : Date`, and `rating : int`. `Category` has attributes `id : String`, `creatable : Date`, `desc : String`, and `logoPath : String`. There are associations between `Writer` and `Article` (1..1), `Article` and `Category` (1..\*), and `Category` and `SubscriptionController` (1..\*).
- presentation:** Contains classes `Template` and `Theme`. `Template` has attributes `id : String` and `title : String`. `Theme` has attributes `id : String` and `zone : String`. `Template` has a dependency on `Theme`.

**Object Structure:** The diagram also shows the implementation of controllers: `PayController`, `SubmitArticleController`, and `SubscriptionController`. `SubmitArticleController` has methods `update(article)`, `confirm()`, and `init()`. `SubscriptionController` has methods `subscribeCategory(cat : Category)` and `unsubscribeEmail(email : String)`.

**Class Diagram**

Introduction to UML

### When UML?

The diagram illustrates a UML Sequence Diagram involving three systems: `Invoice System`, `Inventory System`, and `CRM`.

The sequence of interactions is as follows:

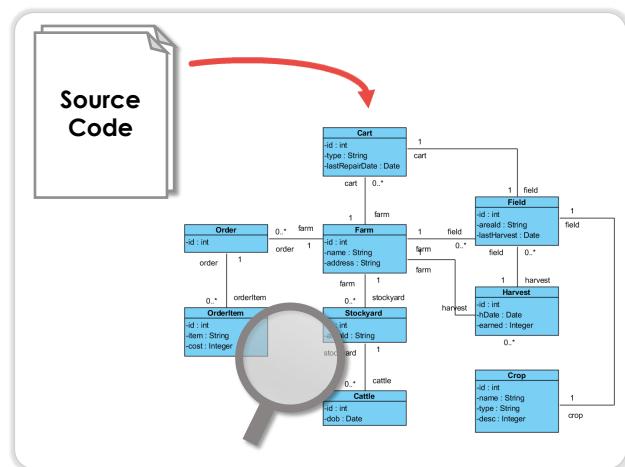
- 1: `create invoice` (from `Invoice System` to `CRM`)
- 1.1: `reserve stock` (from `CRM` to `Inventory System`)
- 1.2: `update credit limit` (from `CRM` to `Inventory System`)
- 2: `request deposit` (from `CRM` to `Invoice System`)
- 2.1: `deposit received` (from `Invoice System` to `CRM`)
- 2.2: `request delivery` (from `CRM` to `Inventory System`)
- 3: `goods delivered` (from `Inventory System` to `CRM`)
- 3.1: `issue demand note` (from `CRM` to `Invoice System`)
- 4: `payment received` (from `Invoice System` to `CRM`)
- 4.1: `update credit limit` (from `CRM` to `Inventory System`)

**Sequence Diagram**

Introduction to UML

## Introduction to UML Part 1

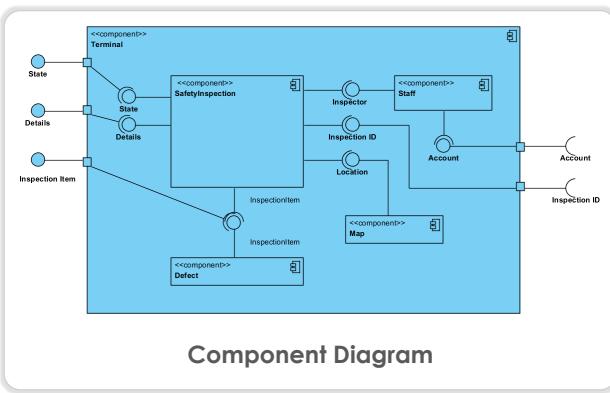
### When UML?



- Determine system requirements
- Design system implementation
  - Object structure
  - Interaction between systems
- Re-engineering software system

Introduction to UML

### When UML?



- Determine system requirements
- Design system implementation
  - Object structure
  - Interaction between systems
- Re-engineering software system
- Design deployment architecture

Introduction to UML

## Introduction to UML Part 1

When UML?

A Deployment Diagram illustrating the deployment of artifacts onto nodes. The diagram shows five nodes: Web Server, Application Server, DB Server, PDA, and Terminal. The Web Server node contains a 'Web Site' artifact. The Application Server node contains 'IMS.jar' and 'ORM.jar' artifacts. The DB Server node contains a 'MySQL Server' artifact. The PDA node contains a 'IMSClient.jar' artifact. The Terminal node also contains a 'IMSClient.jar' artifact. Interactions are shown as lines connecting the Application Server to the DB Server, the Application Server to the PDA, and the Application Server to the Terminal.

Deployment Diagram

- Determine system requirements
- Design system implementation
  - Object structure
  - Interaction between systems
- Re-engineering software system
- Design deployment architecture

Introduction to UML

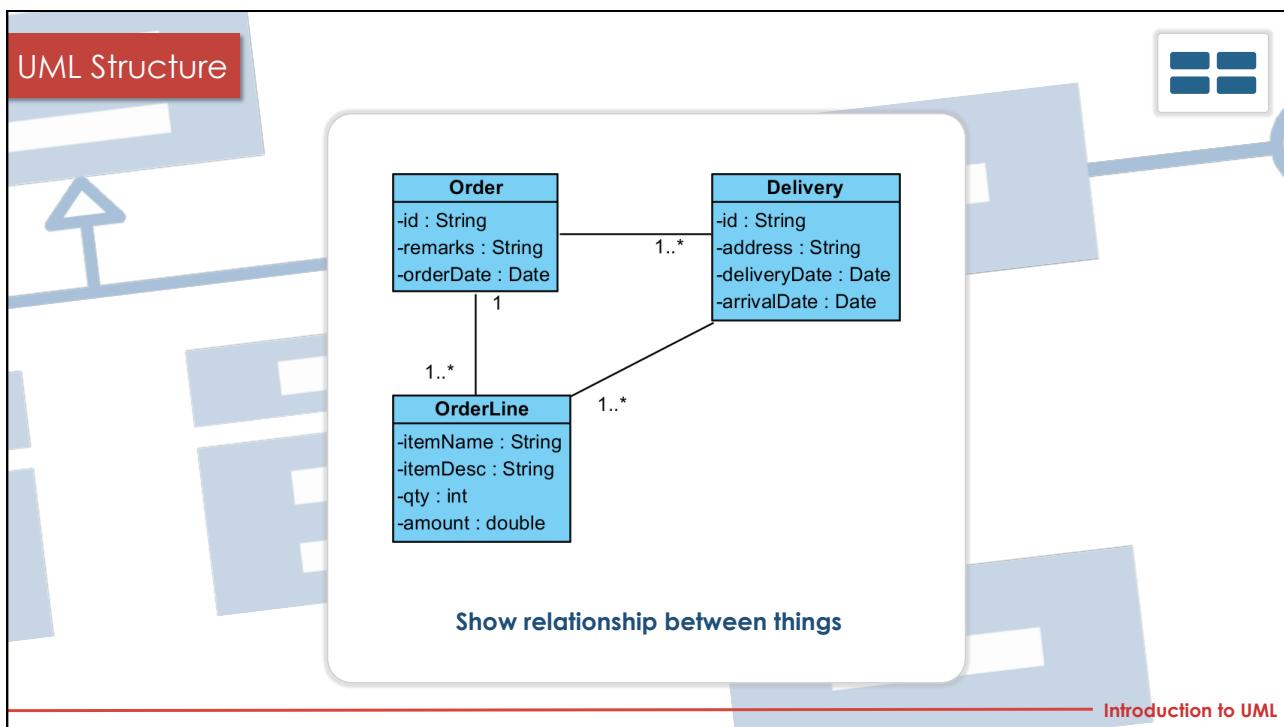
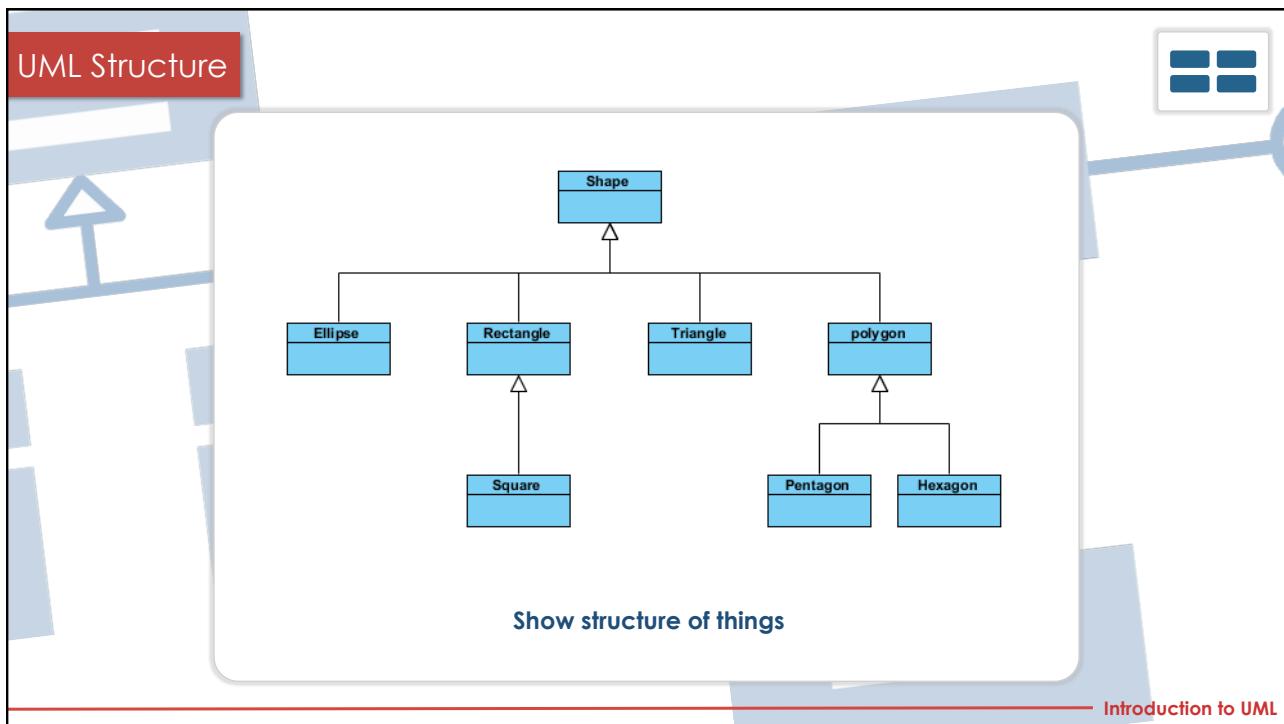
UML Structure

The diagram illustrates the three main types of UML diagrams:

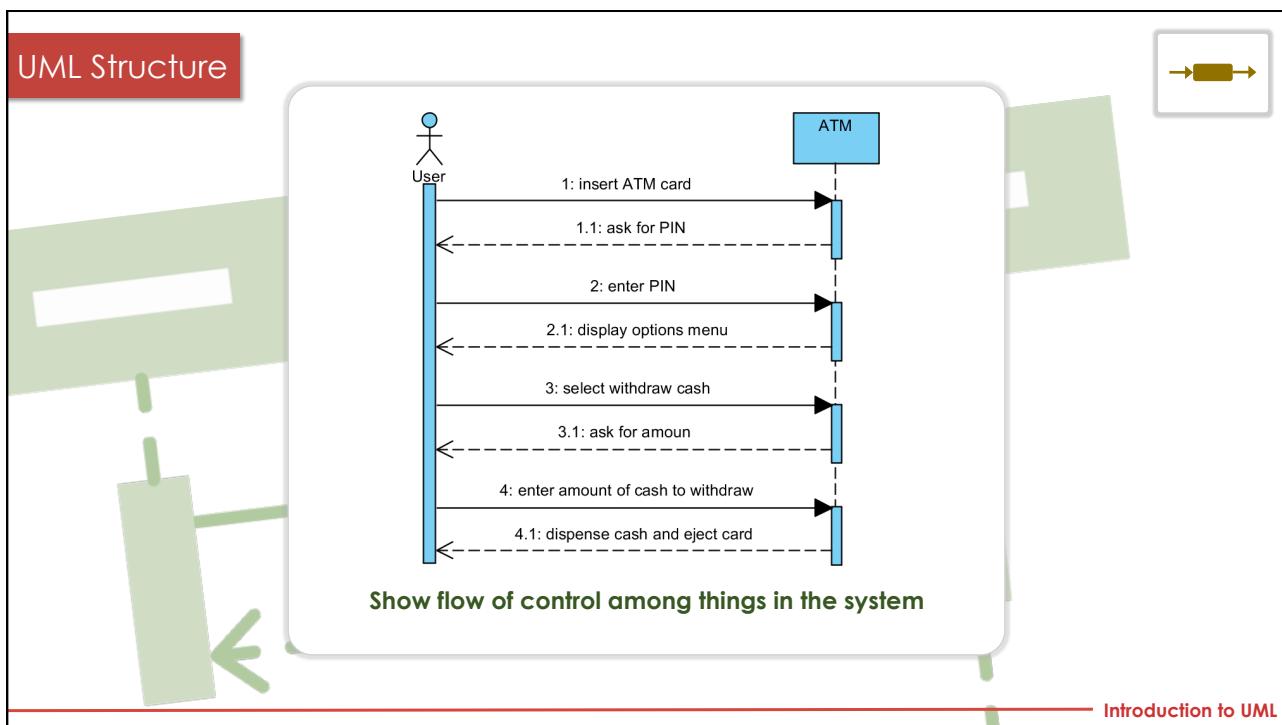
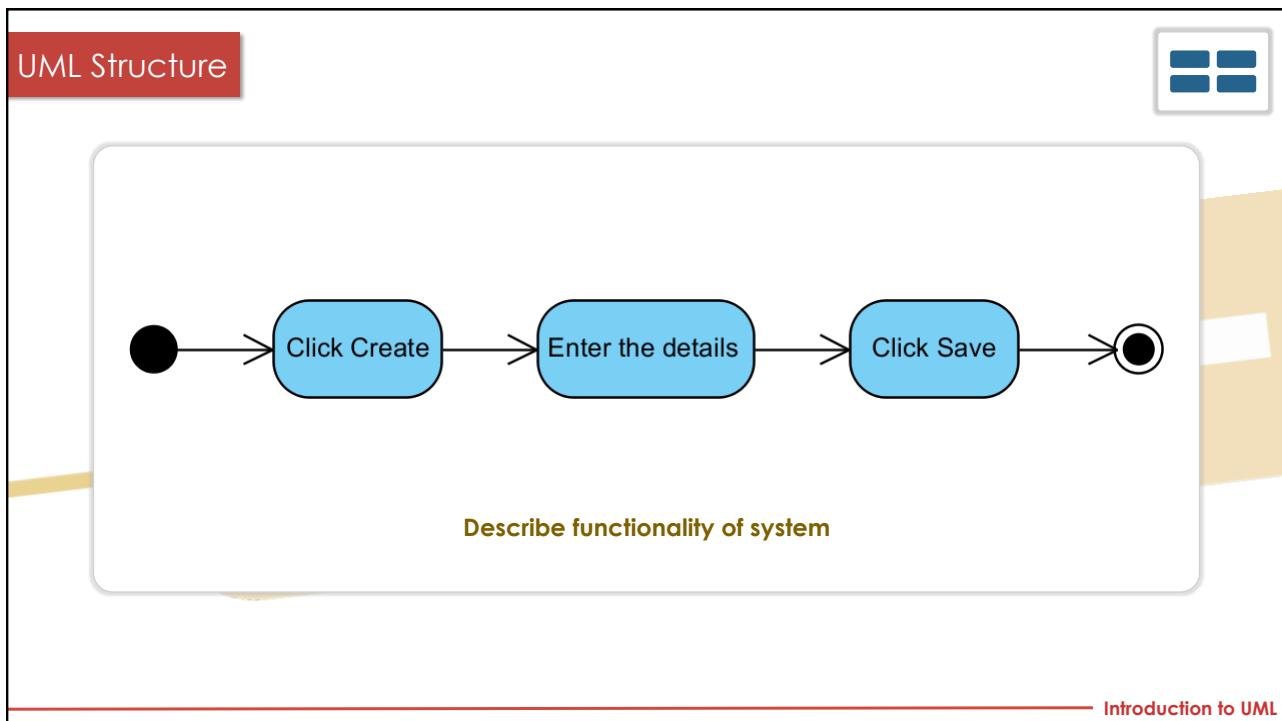
- Structure Diagrams:** Represent the static structure of a system, showing objects and their relationships. It features four blue rounded rectangles arranged in a 2x2 grid.
- Behavior Diagrams:** Represent the dynamic behavior of objects over time. It features a single yellow rounded rectangle with two horizontal arrows pointing in opposite directions.
- Interaction Diagrams:** Represent interactions between objects. It features two green rounded rectangles connected by a vertical double-headed arrow.

Introduction to UML

## Introduction to UML Part 1



## Introduction to UML Part 1



## Summary

- What is UML?
- Why do we need UML?
- When to use UML?
- Structure of UML

---

Introduction to UML