# Secure Software Engineering Lab05: UML Diagrams

Mohammed Alqmase 25/12/2024



# Subsystem Name: Administrative Tools Subsystem in a Judicial Management System

#### GROUP NAMES:

- 1. Mohammed Qasabah
- 2. Mazen Al-ammari
- 3. Mohanned Shaaf
- 4. Mohammed Al-howshabi

# Secure Software Engineering Labo5: UML Diagrams



Mohammed Alqmase 25/12/2024

#### **Table of Contents**

Too	Tools Required:		
	Task 1: Create Use Case Diagram (30 minutes)		
	Task 2: Create an Entity-Relationship Diagram (ERD) (40 minutes)		
	Task 3: Design a Class Diagram (30 minutes)		
4	Task 4: Create a Sequence Diagram (30 minutes)	7	

# Secure Software Engineering Labo5: UML Diagrams

Mohammed Alqmase 25/12/2024



# **Creating UML and ERD Diagrams**

This lab provides hands-on experience in creating UML and ERD diagrams for subsystem modeling. It ensures students can visualize and design the architecture of their assigned subsystem effectively, bridging the gap between analysis and implementation.

#### **Tools Required:**

	draw.io	(Diagrams.net)
--	---------	----------------

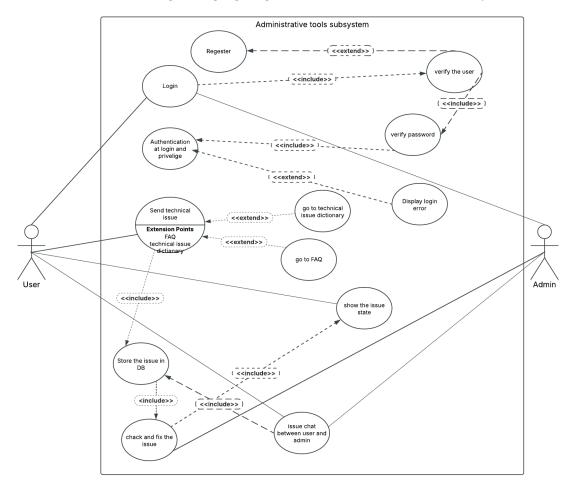
## Secure Software Engineering Lab05: UML Diagrams

Mohammed Alqmase 25/12/2024



#### 1 Task 1: Create Use Case Diagram (30 minutes)

- 1. Identify the core use cases for the subsystem.
  - Example: For "Litigant Portal," use cases may include "Check Case Status," "File Complaint," and "Receive Notifications."
- 2. Draw a Use Case Diagram showing the relationships between actors and use cases.
- Include system boundaries and extend/include relationships where applicable. **Deliverable:** A Use Case Diagram highlighting the core functionalities of the subsystem.



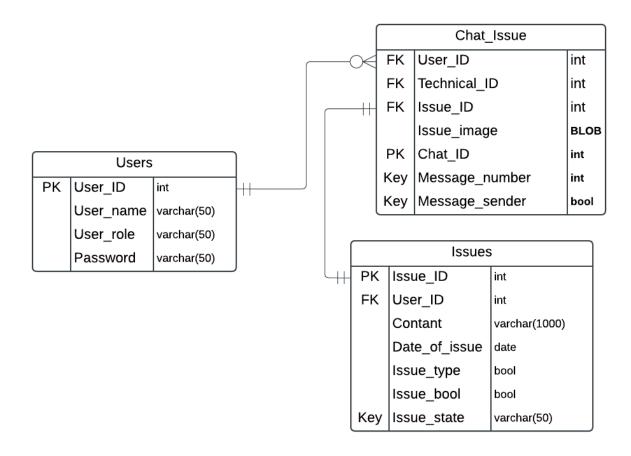
### Secure Software Engineering Labo5: UML Diagrams

Mohammed Alqmase 25/12/2024



#### 2 Task 2: Create an Entity-Relationship Diagram (ERD) (40 minutes)

- 1. Identify the key entities within the subsystem and their relationships.
  - Example: For "Document Management," entities may include Document, Case, User, and Access Log.
- 2. Define attributes for each entity and primary/foreign keys for relationships.
- 3. Ensure cardinality (1:1, 1:N, N:M) is correctly represented.



**Deliverable:** A complete **ERD** showing the relationships between entities in the assigned subsystem.

#### Secure Software Engineering Labo5: UML Diagrams

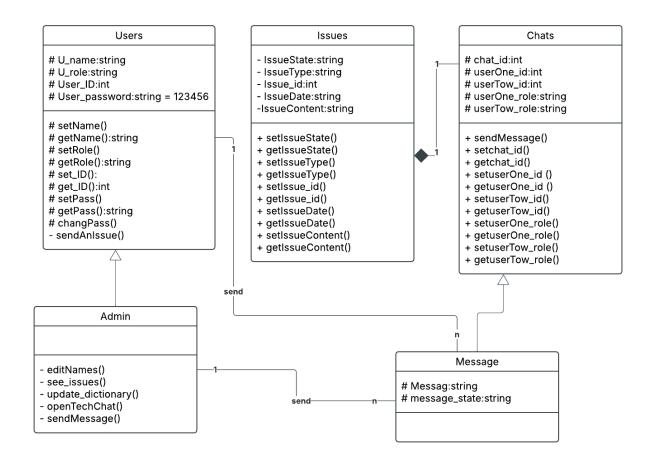
Mohammed Alqmase 25/12/2024



#### 3 Task 3: Design a Class Diagram (30 minutes)

- 1. Identify key classes, their attributes, and methods for the subsystem.
  - Example: For "Security and Role Management," classes may include User, Role, and Permission.
- 2. Define relationships such as inheritance, aggregation, and composition.
- 3. Include visibility markers (public, private, protected) for attributes and methods.

**Deliverable:** A **Class Diagram** detailing the structure of the subsystem.



## Secure Software Engineering Labo5: UML Diagrams

Mohammed Alqmase 25/12/2024



#### 4 Task 4: Create a Sequence Diagram (30 minutes)

- 1. Select one core functionality of the subsystem.
  - Example: For "Courtroom Scheduling," the functionality could be "Assign Judge to a Hearing."
- 2. Illustrate the interaction between objects/actors over time.
  - Include lifelines, messages, and activation boxes.

**Deliverable:** A **Sequence Diagram** showing the flow of actions for the selected functionality.

