



# CSci363 User Interface Design

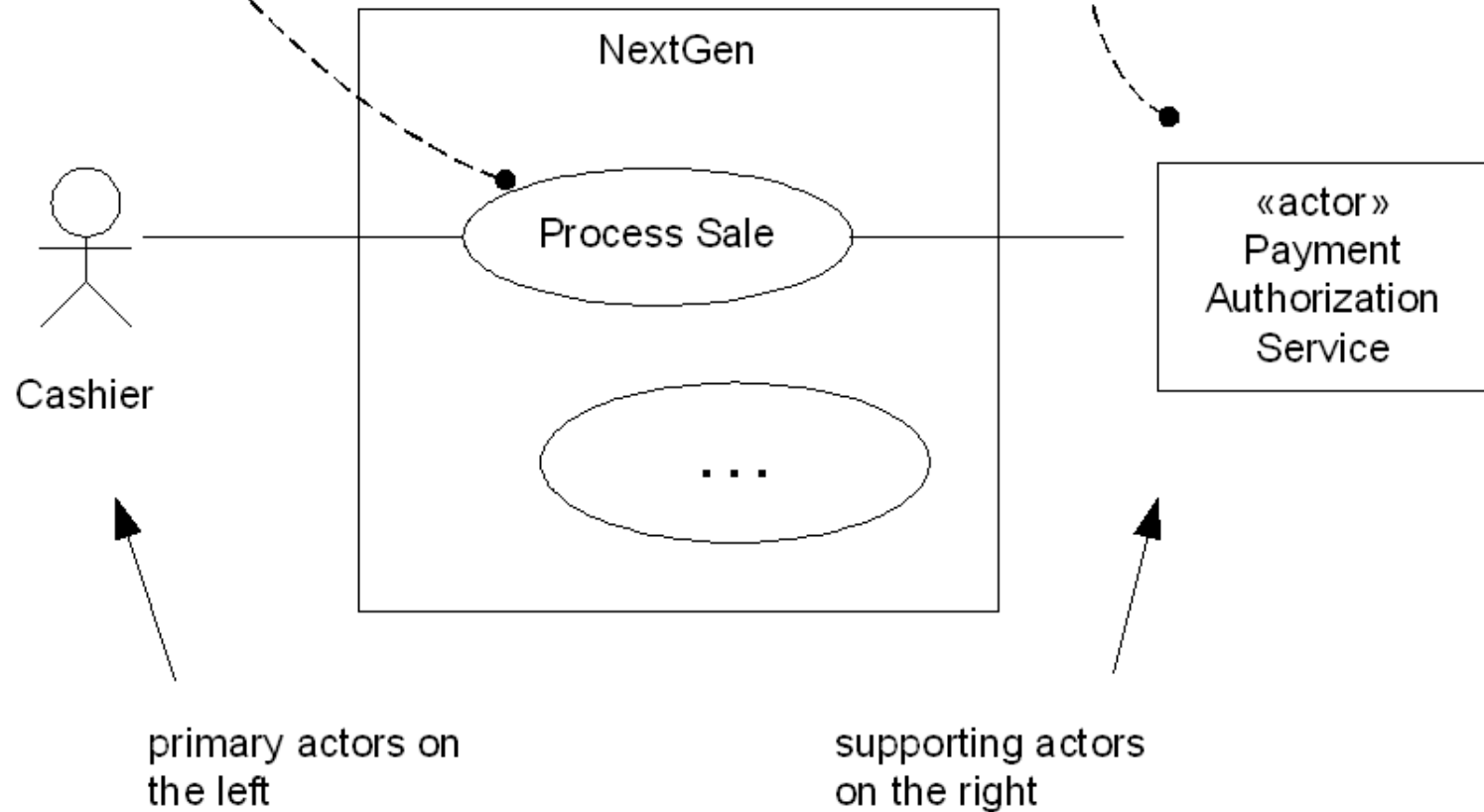
Monday, September 16, 2024

- **Today's session:**
- Software Engineering Models

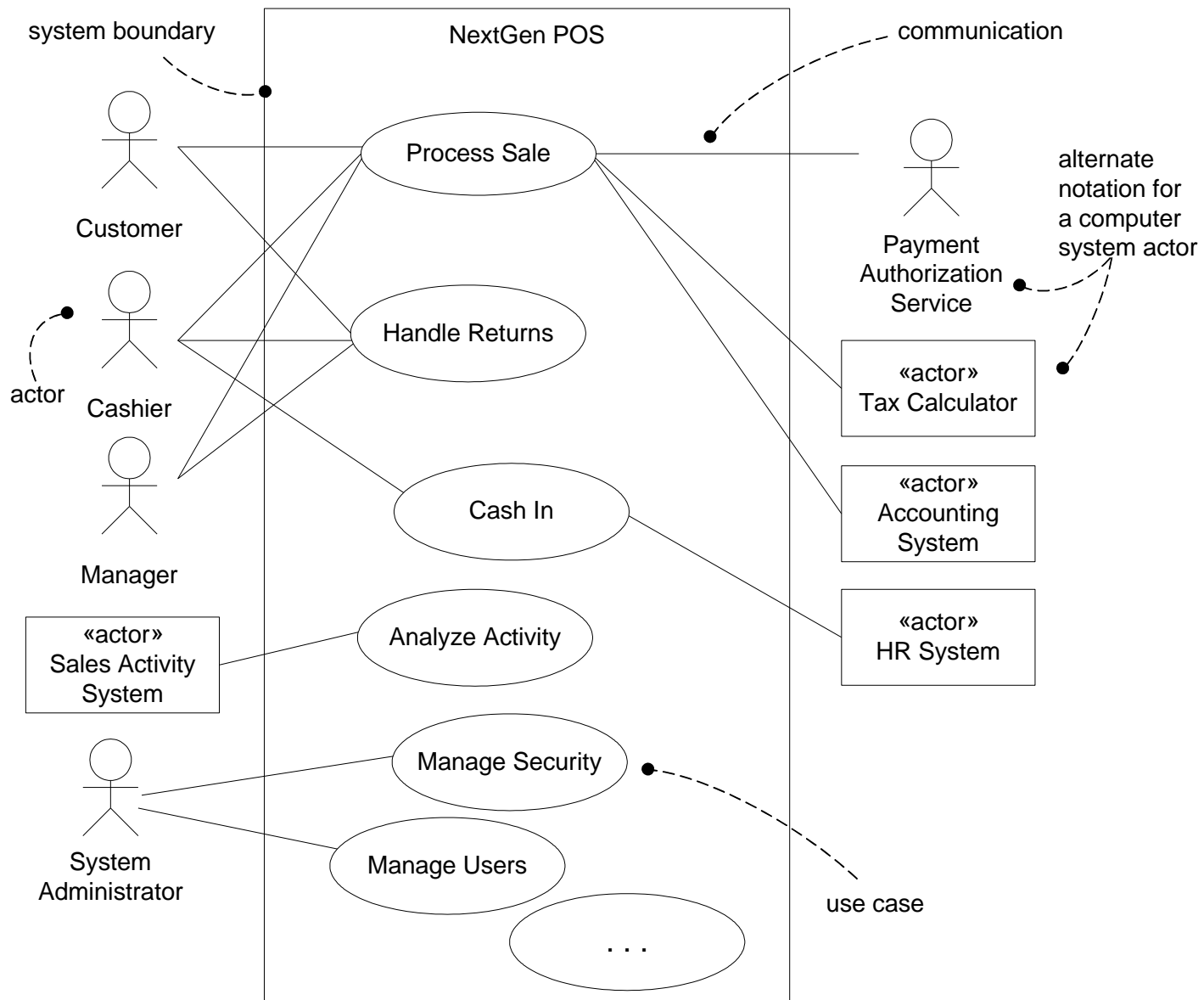
# Use Case Diagram

For a use case context diagram, limit the use cases to user-goal level use cases.

Show computer system actors with an alternate notation to human actors.



# Use Case Diagram



# Process for Creating Requirements Level Use Case Diagram

Extract verb/verb phrases from specification,

Identify candidate use cases,

- Is this entity relevant?
- Is this entity within the scope of the system?
- Is this entity at the highest level of abstraction?
- Decide name of use case

Identify candidate users,

Candidate users are taken from the Class Diagram.

Draw the use cases,

Identify relationships between candidate users and use cases,

Obtain from specification and customer.

Draw the relationships.

# Process for Creating Requirements Level CD



Extract noun/noun phrases from specification,



Identify candidate classes,

Is this entity relevant?

Can I uniquely identify this entity?

Is this entity within the scope of the system



From the remaining entities, identify candidate attributes,

Is this entity relevant?

Is this candidate a property of a candidate class?



Draw the classes,



Identify relationships between the candidate classes,

Obtain from specification and customer.



Draw the relationships.

# University Library Book Rental System (LibSys)

## Borrow a book Description:

A **user** of the library may log into the system to **borrow** a **book** from the **collection**. The user will **enter** his **user-name** and **password**. User-names must be at least **eight characters**, start with an **alpha character**, must contain at least **one digit**, and only the **special characters, underscore, and hyphen** are allowed. The **system** will **verify** the **user's login** and **access policy**. If the login is valid the user will be able to **select** a **book** to be **borrowed** from the **library**. If the login is invalid an **error message** is **reported** and the **system returns** to the **login state**.

The **user** will be **prompted** to enter **identification information** for the requested **book**. The **system** will **verify** the presence of the **identified book** in the library's **collection**, the users **authorization to borrow** the **identified book**, and the availability of a **copy of the book** for lending for the user's associated **policy borrowing period**. If the **verification** fails at any point the **system** will **report** the failure and return to the state of awaiting input of a book's **identification data**.

Upon **passing** lending verification the **system** will **generate** a **lending receipt** and **record** all **information** associated with this **transaction** then **return** to the state awaiting another **transaction request**.



# Process for Creating Models

1. Read the assignment and the specification,
  - Do I understand what is required?
  - Do I understand the notation?
2. Need help?
  - Ask questions in class,
  - Review relevant recorded lecture,
  - Do some research,
  - Ask questions during instructor's office hours,
  - Do not send email query.
3. Attempt the assignment,
  - Go to 2. if necessary.
4. Complete assignment,
  - Submit via Blackboard,
  - Do not email assignments.

## Process for Grading Modeling Assignments

- All correctly submitted assignments start out with 100%,
- 1 – 5 percentage point deduction,
  - Each error in model,
  - Non-executable program,
  - Missing header page information (student name, assignment number, date)
- Assignment emailed without permission will be ignored,
- Late assignment, as per syllabus statement,
- Assignments submission deadline is end of day on the due date.



# UML Textbook

