Network Programming

Network Program Design

Note: **This class lecture will be recorded!**
If you do not consent to this recording, please do not ask questions via your video, audio or public chat; send your question to the instructor using the private chat.

Lisa Frye, Instructor
frye@Kutztown.edu
Kutztown University

Copyright Lisa Frye 2017

Network Program Design

- Previous experience – test a single program
- Network programming
  - Interaction between multiple programs

Design a Protocol

- Develop a new protocol
- How do the programs interact?
  - Message sent
  - Responses to specific messages
  - How to indicate end of message or end of data?
    - How does other host know it has received all the data sent?

Use Case Diagram

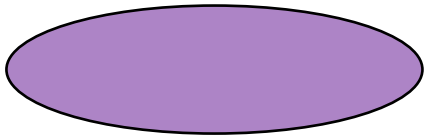
- Drives development process
- Typical interaction between user and system
- External view of system

- Scenario

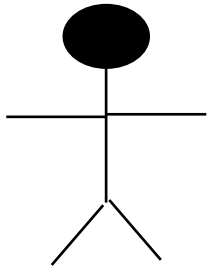
Use Case Characteristics

- Understand system requirements
- Show system behavior
- Depict activities performed
- Describe external users triggering an event
- Useful brainstorming technique to develop requirements analysis
- Understand user requirements
- Help develop tests
- Descriptive answers

Use Case Symbols



Use Case / System functionality



Actor

Boundary

Connector

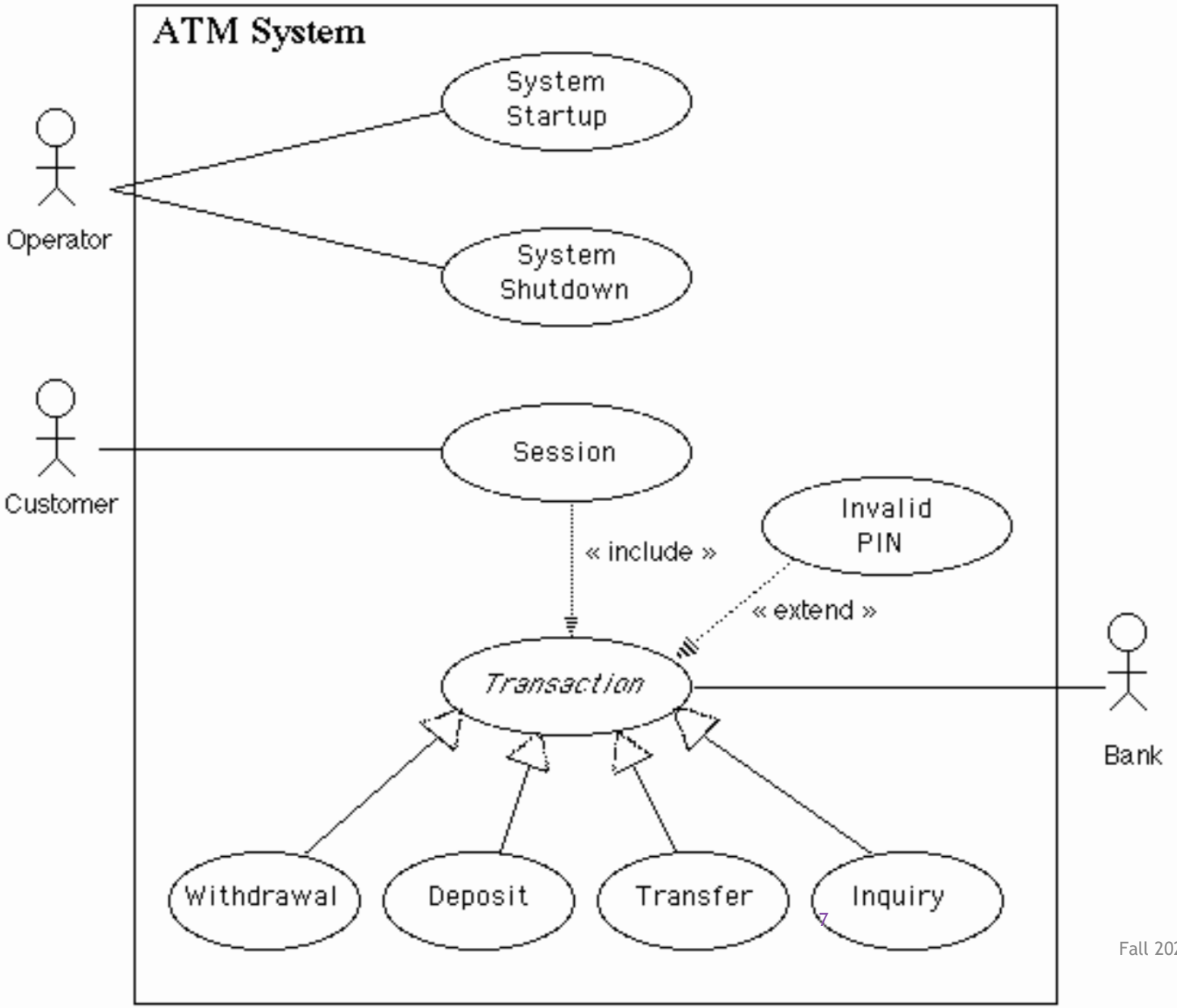
<<include>>

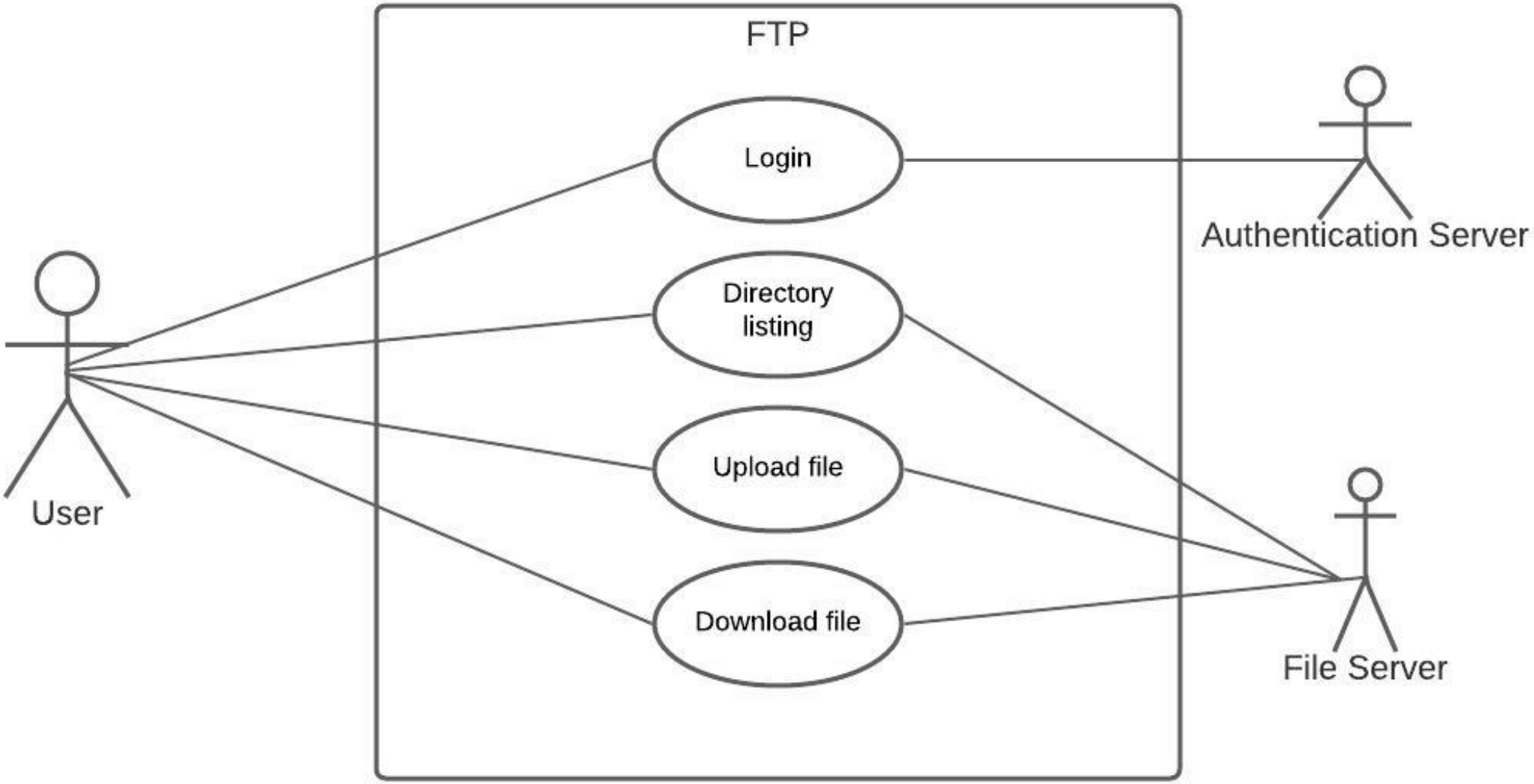
Include Relationship

<<extend>>

Exclude Relationship

Use Case Example: ATM




FTP
Login
Directory listing
Upload file
Download file
User
Authentication Server
File Server

Sequence Diagrams

- Illustrates process interaction
  - Interaction Diagrams
- Show a number of objects and the messages that are passed between the objects
- Help understand the flow of the program

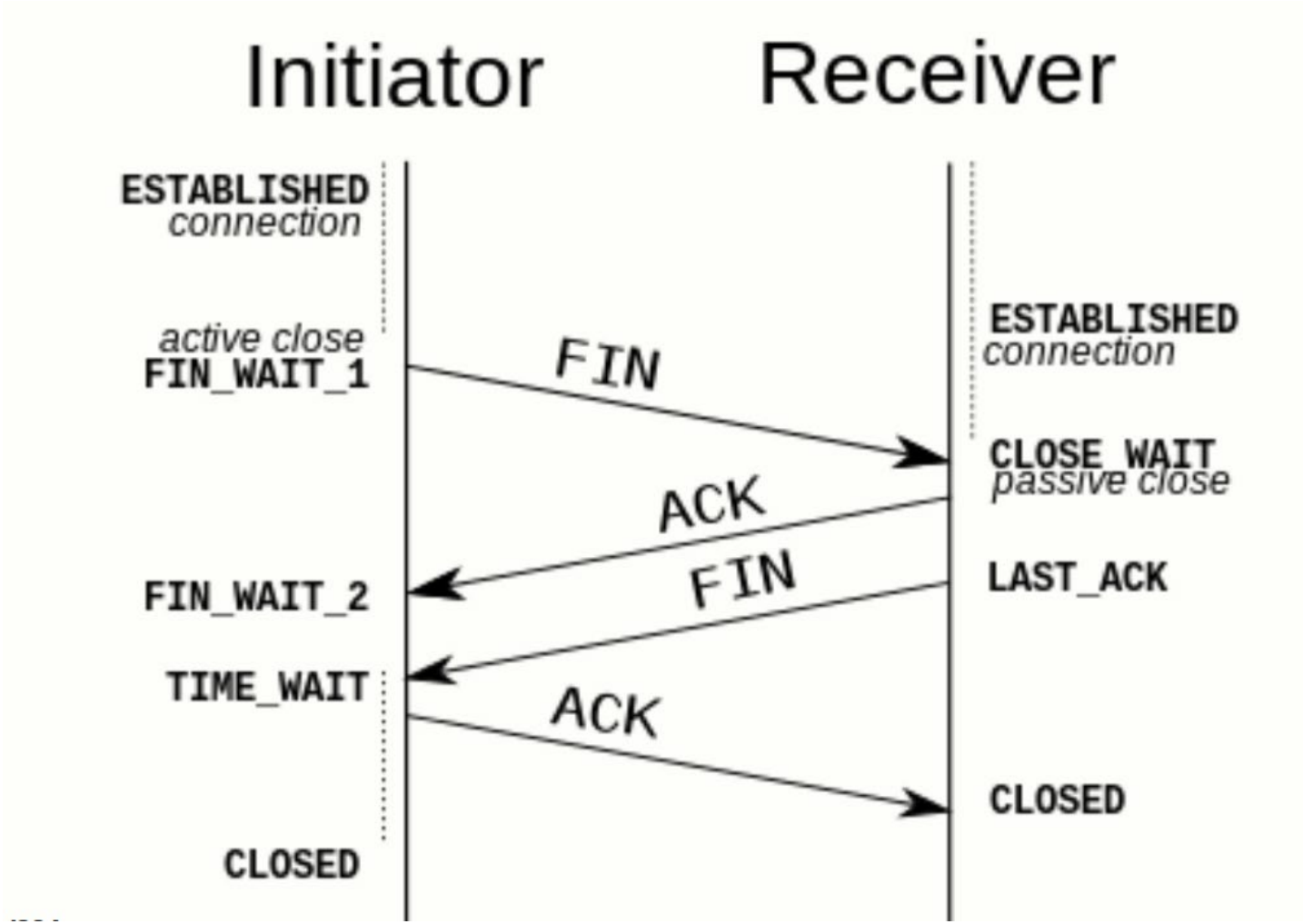
Sequence Diagram Symbols

- Objects / Participants – rectangle (across top)
- Lifeline – dashed line
- Messages – arrows
  - Message – solid line
  - Return / response – dotted line
- Activation Box – rectangle (vertical)

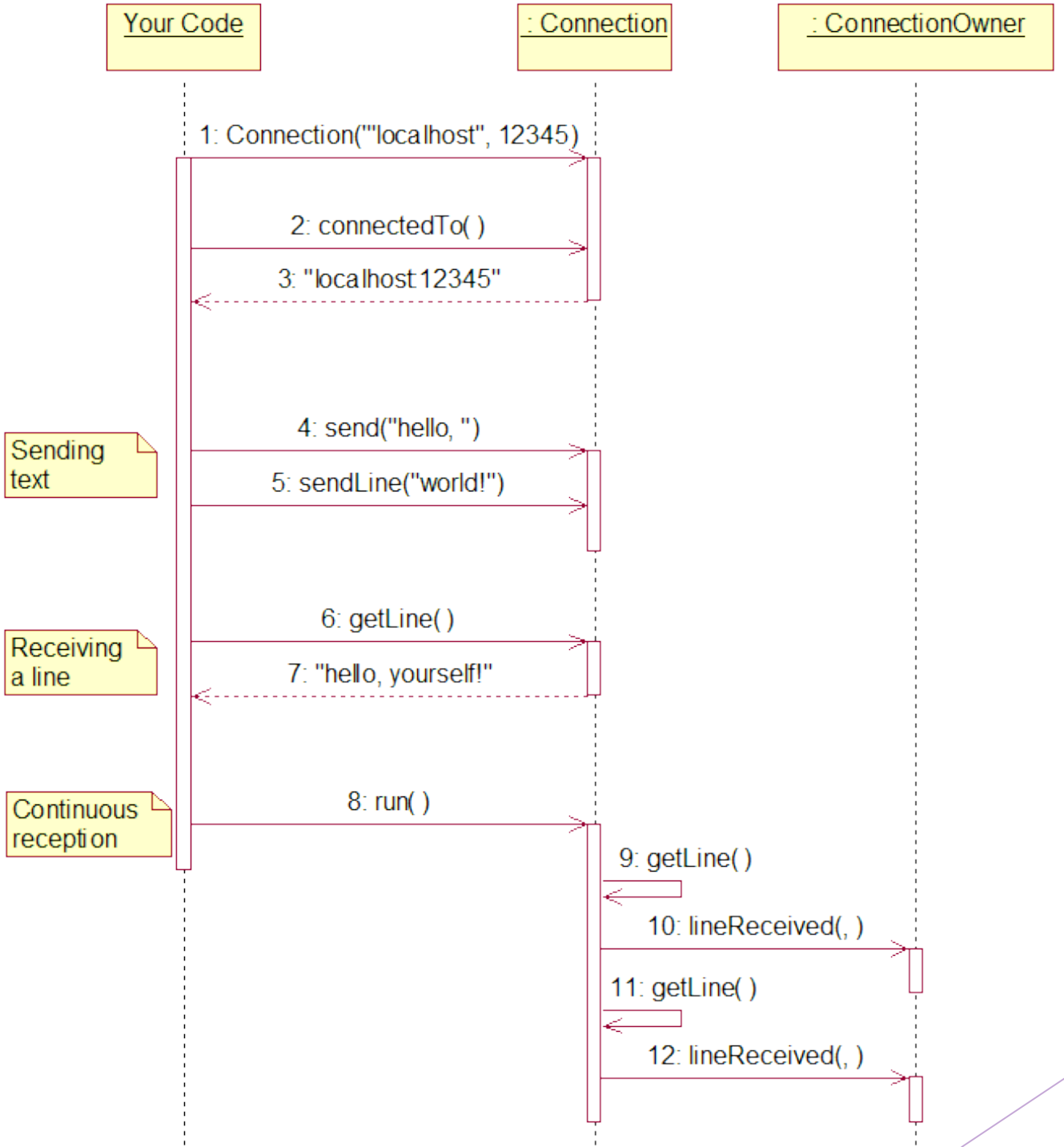
Messages

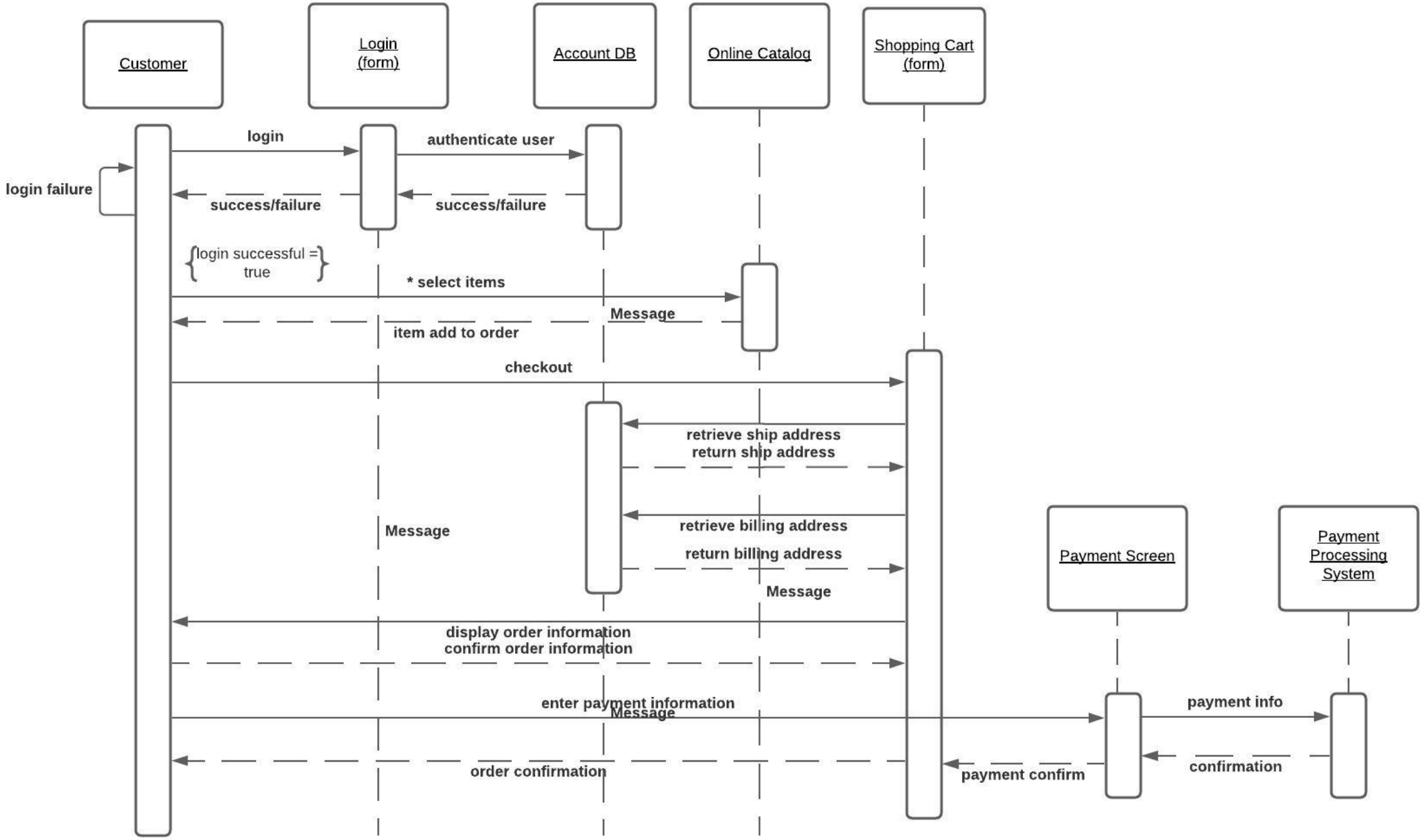
- Solid arrow from sender to receiver
- Self-call
  - Object sends message to itself
- Return
  - Dashed line
  - Not a new message

Sequence Diagram Example



TCP Modified Three-way Handshake (close connection)


Your Code
: Connection
: ConnectionOwner
1: Connection("'localhost", 12345)
2: connectedTo( )
3: "localhost 12345"
4: send("hello, ")
Sending text
5: sendLine("world!")
6: getLine( )
Receiving a line
7: "hello, yourself!"
8: run( )
Continuous reception
9: getLine( )
10: lineReceived(,)
11: getLine( )
12: lineReceived(,)


Customer
Login (form)
Account DB
Online Catalog
Shopping Cart (form)
Payment Screen
Payment Processing System
login
authenticate user
login failure
success/failure
success/failure
login successful = true
* select items
Message
item add to order
checkout
retrieve ship address
return ship address
Message
retrieve billing address
return billing address
Message
display order information
confirm order information
enter payment information
Message
payment info
confirmation
payment confirm
order confirmation

Testing

- Inspections / Code walkthroughs
- Desk checking
- Unit testing
- Integration testing
- System testing
- Stub testing
- Performance / Stress Testing
- User Acceptance Test (UAT)

- What are some things you should look for when testing?
- Test plans