Software Engineering COMP1035

Lecture 05

Requirements Modelling

Requirements Engineering

- The Motivation of Requirements Engineering
- Why we need to learn requirements engineering?

1. Communication skills with end users （PPT , LEC3）
2. System design & analytics （UML）
3. Leadership and work with coders (coding)

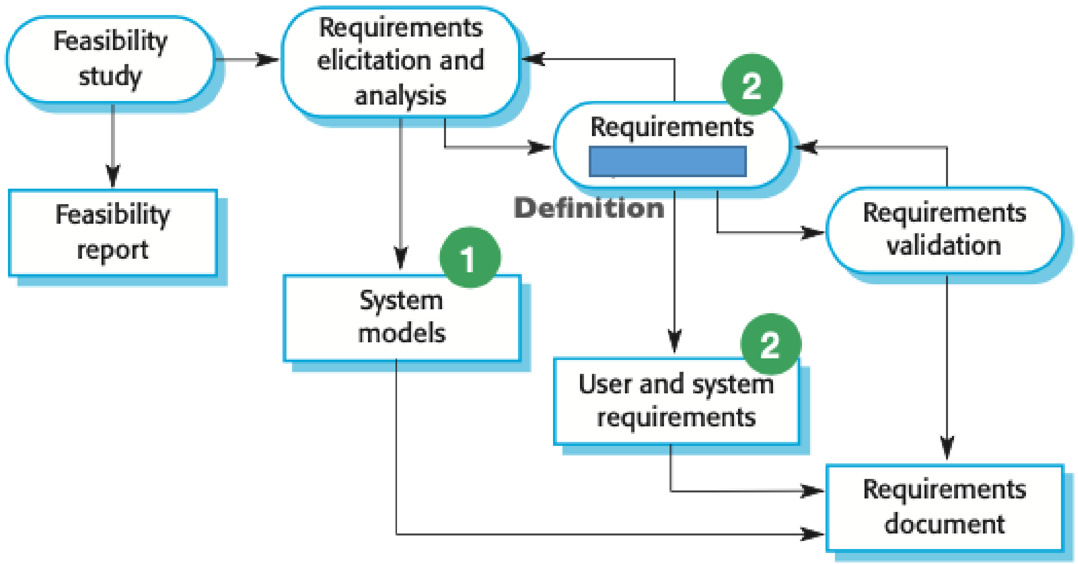


Figure 2.4 The requirements engineering process

Requirements Engineering

Today’s Learning Outcomes

1. What requirements models are, and why they are useful?
2. Different types of UML diagrams.
3. Models aren’t just for requirements.
    - This is a useful transition point to our next stage.

What Are Requirements Models For?

Requirements Models

“It is not a complete representation of the system.”
“It purposely leaves out details and picks out the most salient characteristics.”

- Ian Sommerville (2011) Chapter 5, Systems Modelling

- **Synthesising** all the requirements you collected into key requirements.
- You start making diagrams that represent how **requirements relate**.
- Then you have a **comprehensive set** of **integrated requirements**.

UML Diagrams

(In Microsoft Visio / Visual Paradigm)

Unified Modeling Language (UML)

- There used to be three popular approaches to producing models.
    - Ones specifically for modelling object-oriented software in software engineering.
- In 1990, they got “unified”.
- UML 2 includes 13 different diagrams.
    - Each for different purposes.
- It’s possible to automatically produce outline code from UML.
    - Called Model Driven Development.
    - But this isn’t entirely popular – the book outlines 4 reasons why in section 5.5.

https://www.uml-diagrams.org/uml-25-diagrams.html

UML – Key Diagram Types

- Context models
- Activity diagrams

Initial Requirements

Could be Req., Spec. or Doc.

- Use case diagrams
- Sequence diagrams

Refined Requirements

Could be Req., Spec. or Doc.

- Class diagrams
- Sequence diagrams
- State diagrams

Architecture Design

Of Code – thu more for Spec. and Doc.

Context Models

- During Requirements Gathering, you identified several systems you will have to interact with
  - Authentication systems, bluecastle, finance etc.
- You want to show how all all these systems inter-relate.
  - Use Case Diagram could have an ‘actor’ – but that’s not what UCDs are ‘for’.
  - You could use a flow diagram (Activity Diagram – see later) – but that’s not what they are ‘for’.
- Context Models are especially **for this problem**.

Context Diagram

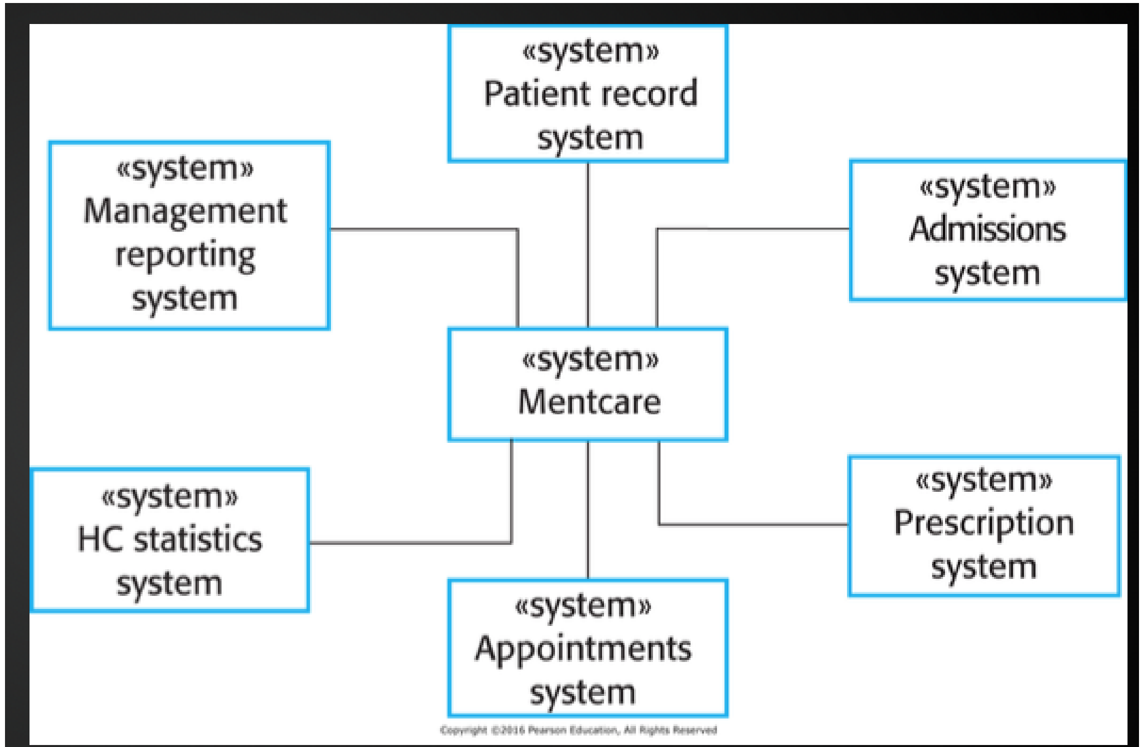

«system» Patient record system
«system» Management reporting system
«system» Admissions system
«system» Mentcare
«system» HC statistics system
«system» Prescription system
«system» Appointments system

***Figure 5.1** The context of the Mentcare system*

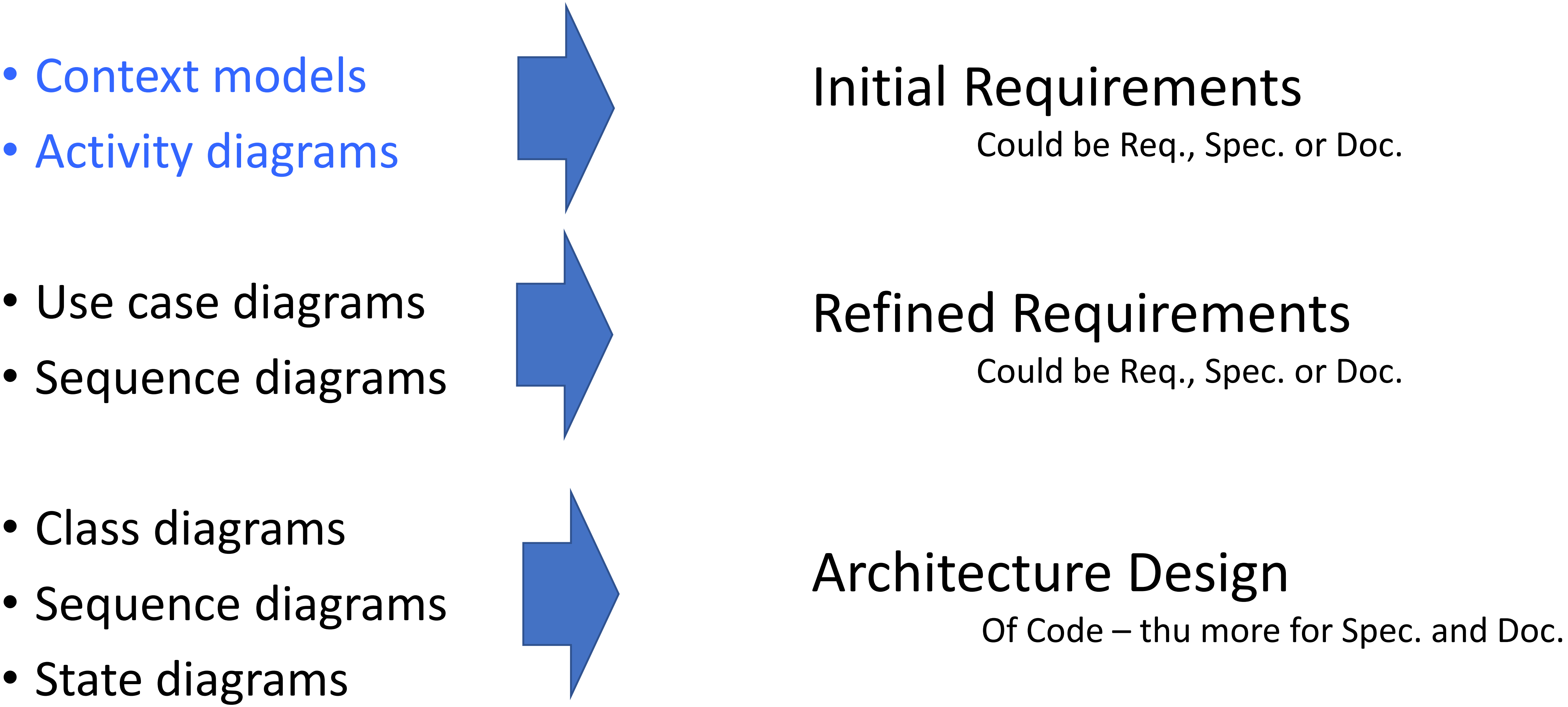
- Show related systems.
- These are often part of the Non-Functional (NF) requirements.
- Can be simple like this one.
- Even show components of a multi-part system (Twitter).

Context Diagram

- Context models define the **boundaries** of they system.
- They represent key systems that **need to be** developed.
- And **the relationship** to the other systems/components.

- Also – **what NOT to develop** (to focus investment).

UML – Key Diagram Types



- Context models
- Activity diagrams

Initial Requirements

Could be Req., Spec. or Doc.

- Use case diagrams
- Sequence diagrams

Refined Requirements

Could be Req., Spec. or Doc.

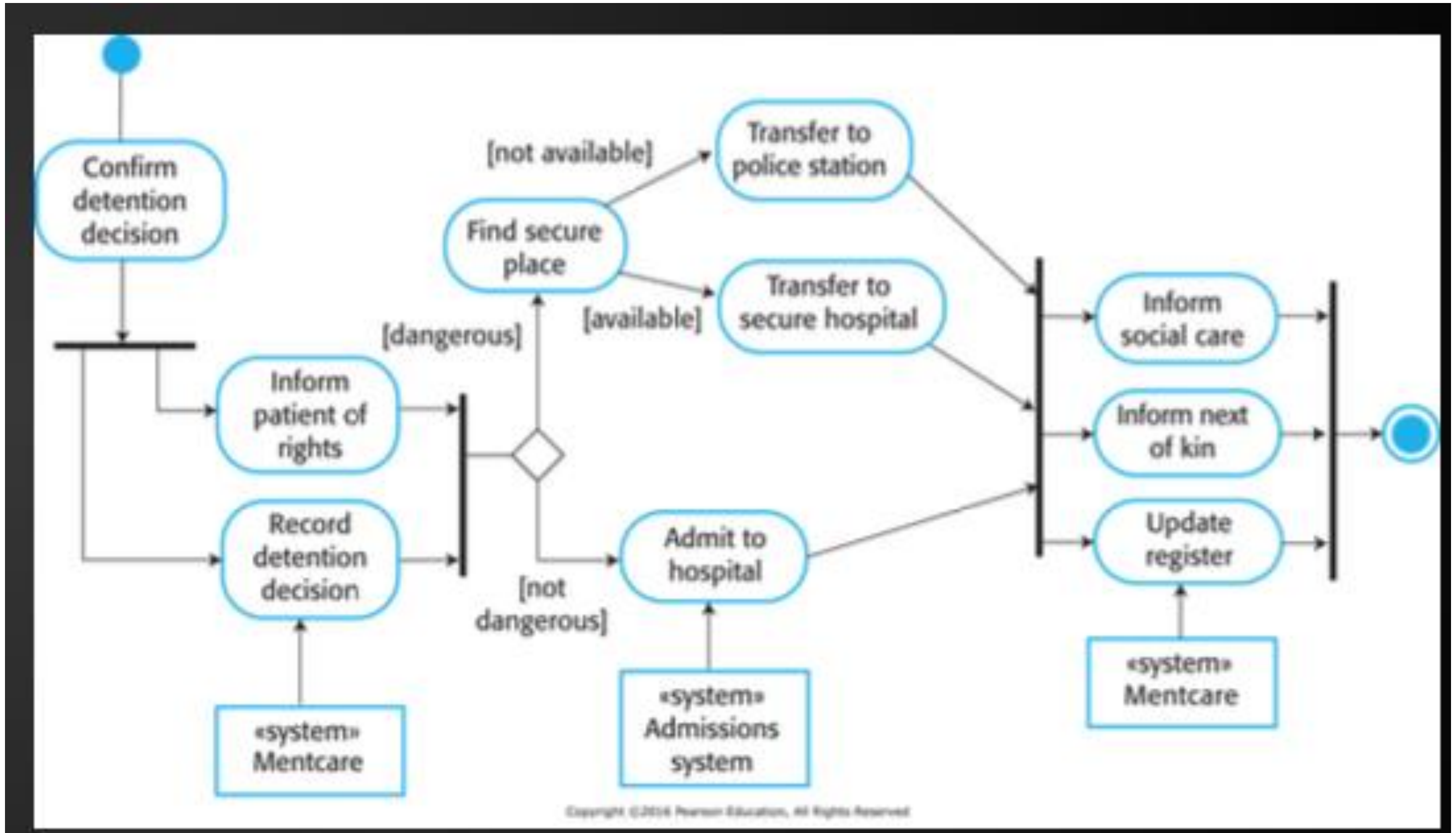
- Class diagrams
- Sequence diagrams
- State diagrams

Architecture Design

Of Code – thu more for Spec. and Doc.

Activity Diagrams

- Used to elaborate workflows for key activities – especially if they involve decisions.
- Explains the process, decision points, wait points & parallel work.
- To define one Use Case in more detail.
- May tie together several Use Cases for one bigger process.



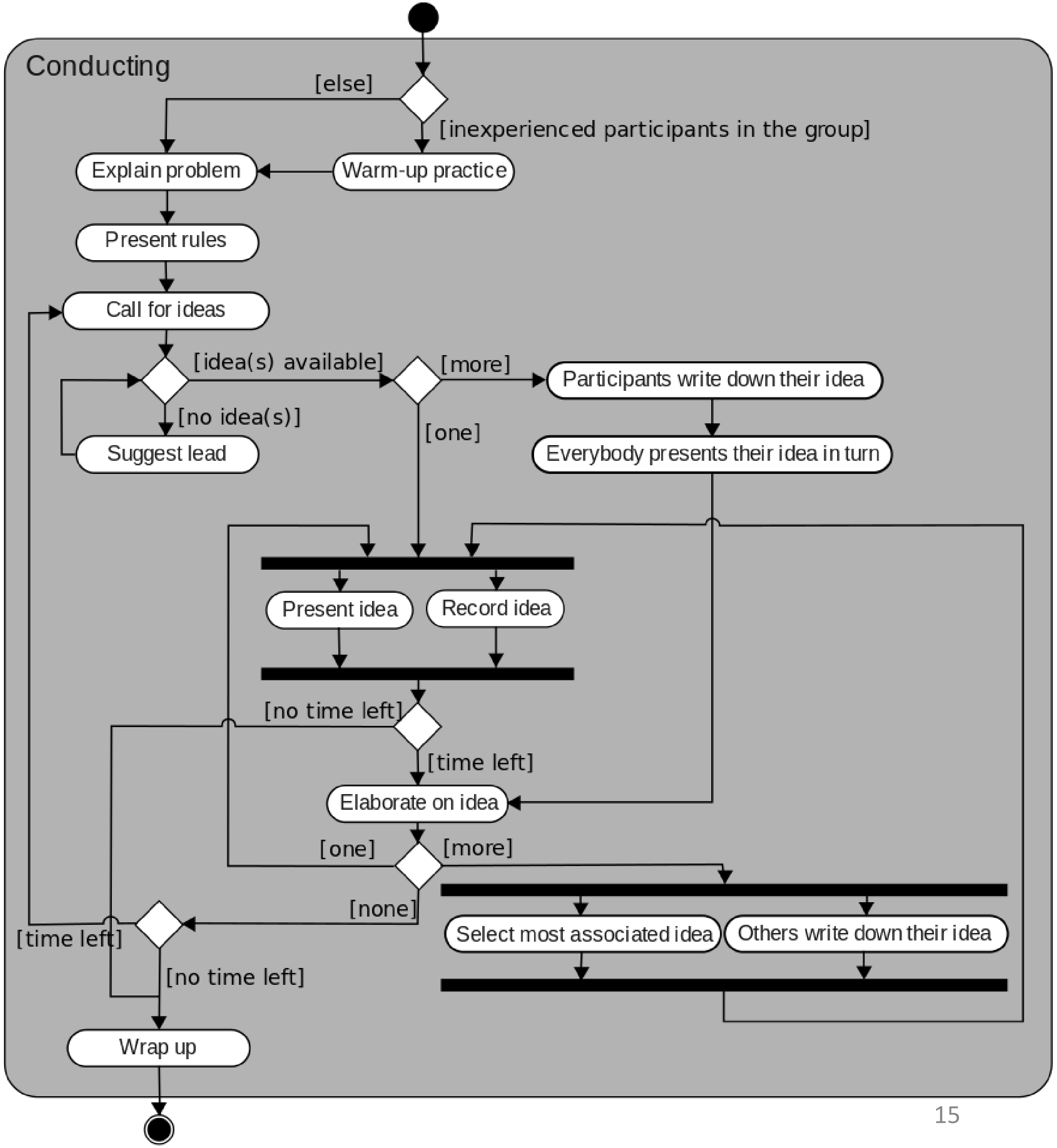
Copyright ©2014 Pearson Education, All Rights Reserved

Figure 5.2 ***A process model of involuntary detention***

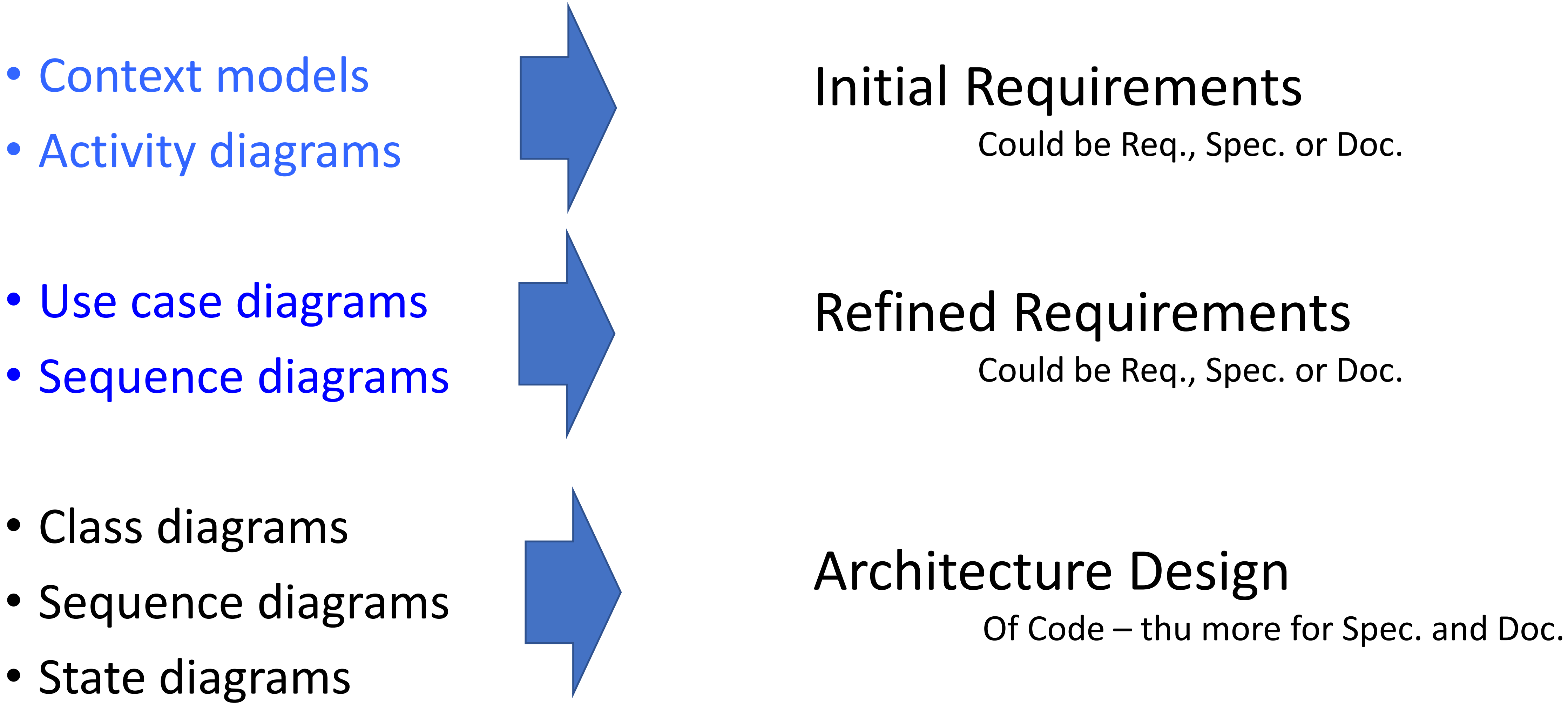
Activity Diagrams

- Rounded rectangles represent actions.
- Diamonds represent decisions.
- Bars represent the start (split) or end (join) of concurrent activities.
- The black circle represents the start.
- The encircled black circle represent the end.

https://en.wikipedia.org/wiki/Activity_diagram



UML – Key Diagram Types



- Context models
- Activity diagrams

Initial Requirements

Could be Req., Spec. or Doc.

- Use case diagrams
- Sequence diagrams

Refined Requirements

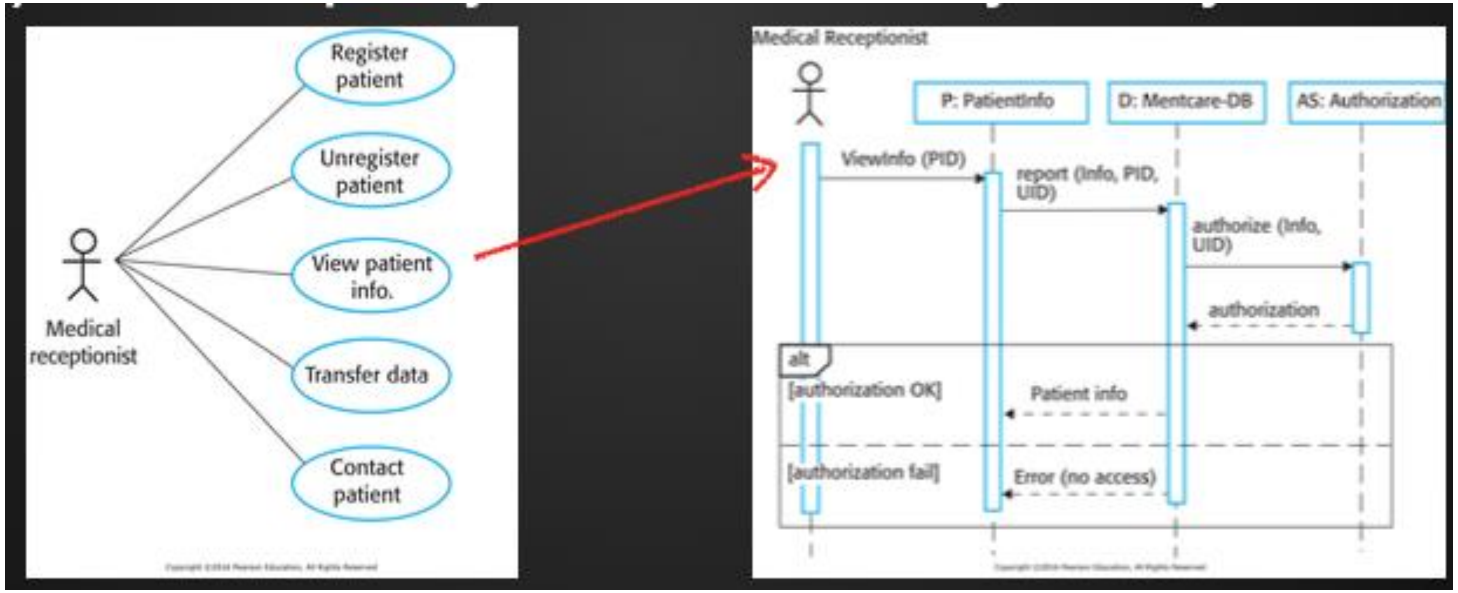
Could be Req., Spec. or Doc.

- Class diagrams
- Sequence diagrams
- State diagrams

Architecture Design

Of Code – thu more for Spec. and Doc.

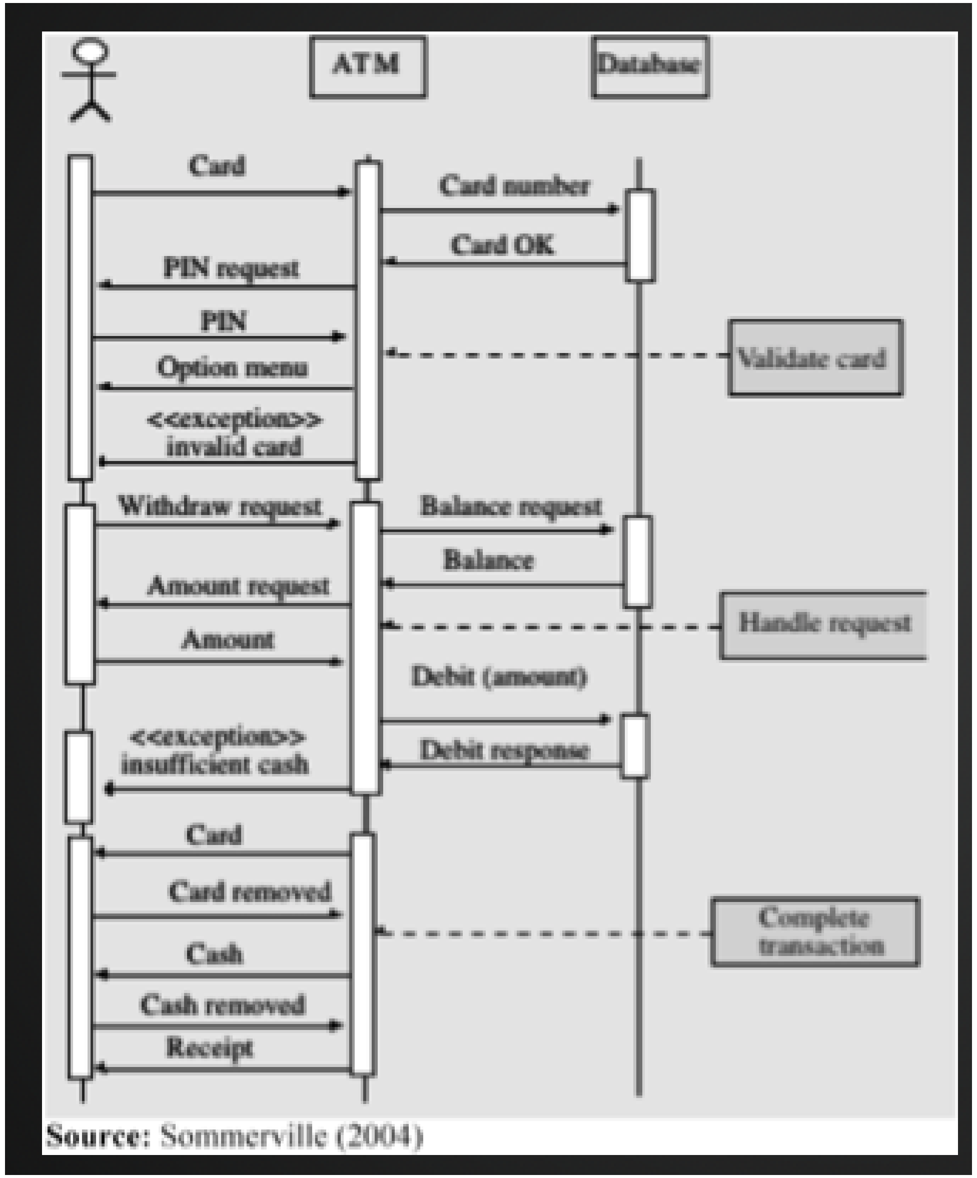
Use Case modeling & Sequence Diagrams



- Good for **complex sharing of information between people and systems**.
- Could be seen as a series of messages between key components.
- Can be (later) used to specify function calls in Java objects.

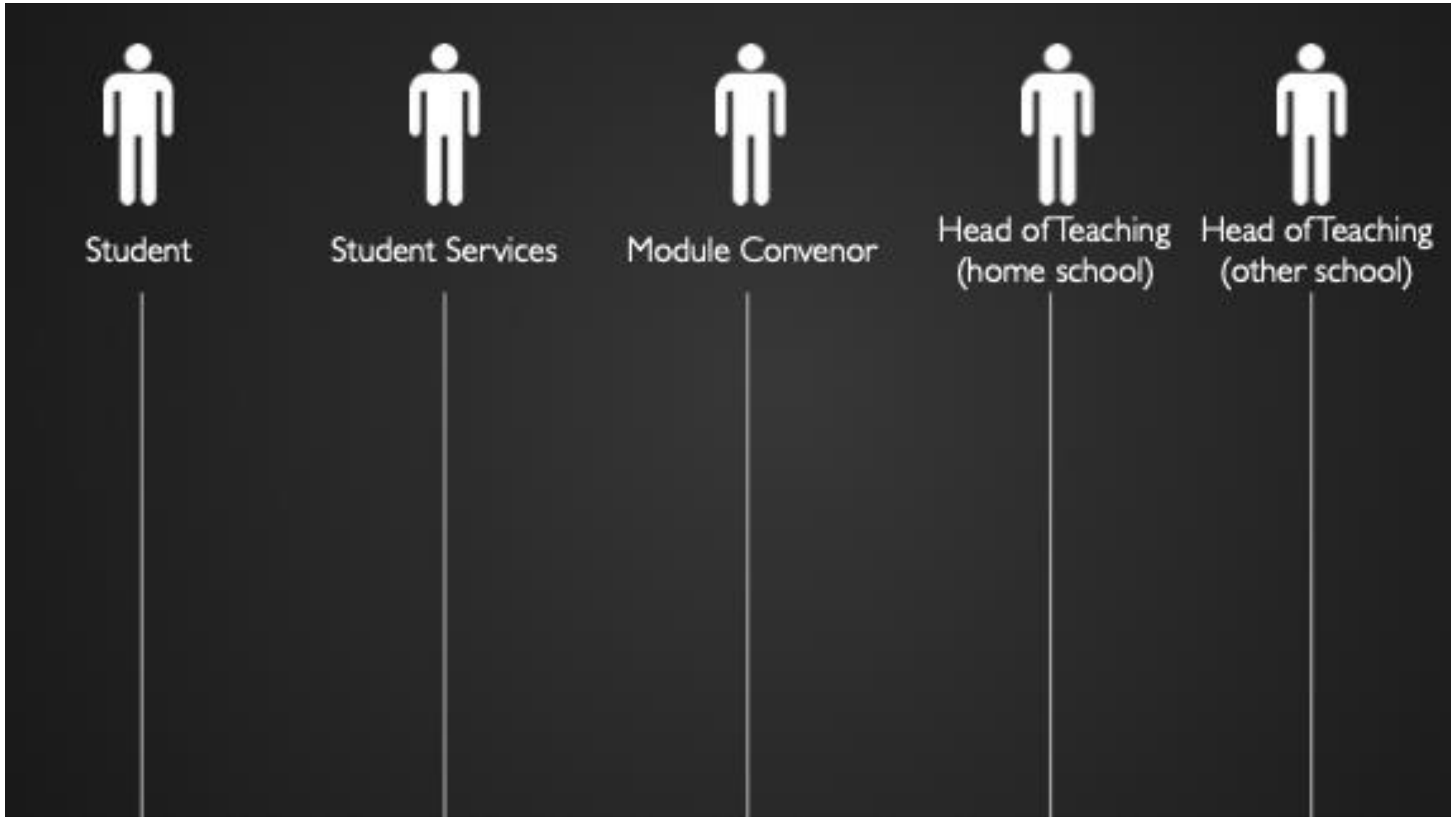
Sequence Diagrams

- Using an ATM:
  - Kevin Curran, David King, (2008) “Investigating the human computer interaction problems with automated teller machine navigation menus”, Interactive Technology and Smart Education,” vol. 5, Iss:1, pp. 59-79.



Source: Sommerville (2004)

Sequence Diagrams – Activity


Student
Student Services
Module Convenor
Head of Teaching (home school)
Head of Teaching (other school)

Other UML (mostly* for later stages, like specifications)

*but can help in Requirements

UML – Key Diagram Types

- Context models
- Activity diagrams

Initial Requirements

Could be Req., Spec. or Doc.

- Use case diagrams
- Sequence diagrams

Refined Requirements

Could be Req., Spec. or Doc.

- Class diagrams
- Sequence diagrams
- State diagrams

Architecture Design

Of Code – thu more for Spec. and Doc.

Class Diagrams

- Specify the classes in Object-Oriented Code.
- Done well – this is obviously more like specification.
- Could be done at Requirements time.
    - a kind of prototype of objects that Requirements Engineers have found.

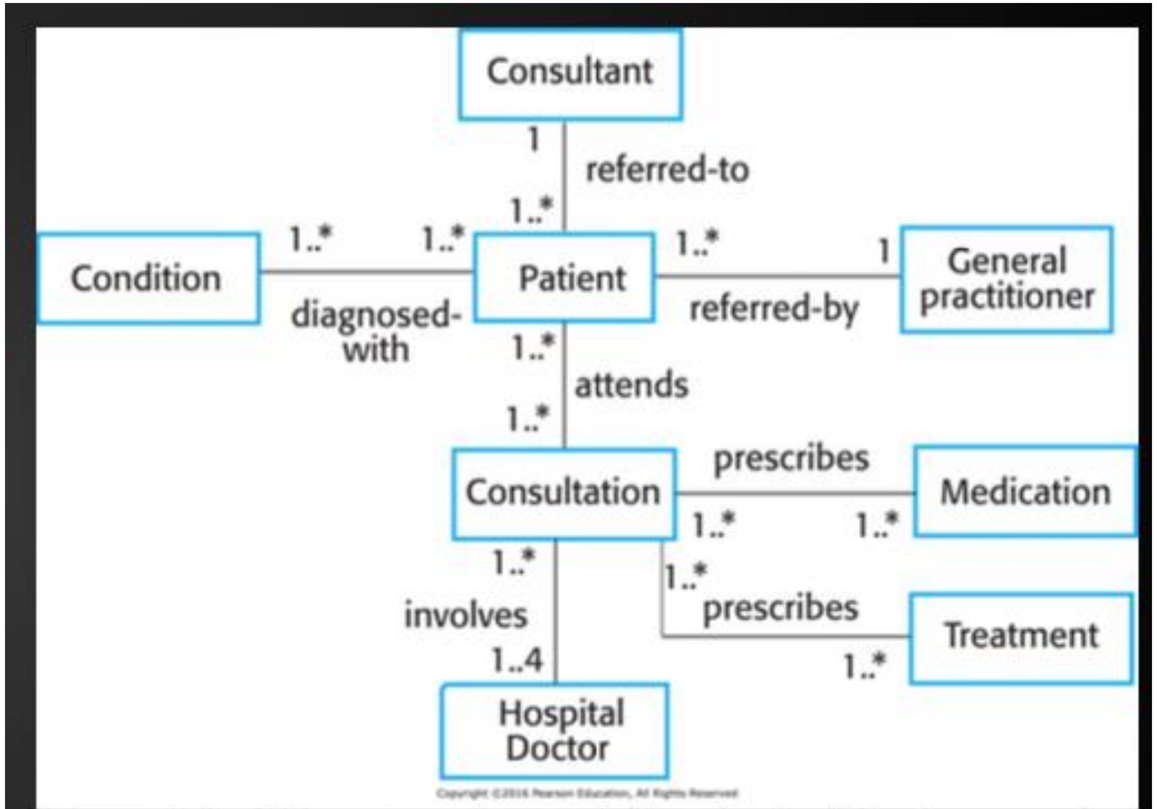


***Figure 5.9** Classes and associations in the Mentcare system*

Class Hierarchy Diagram

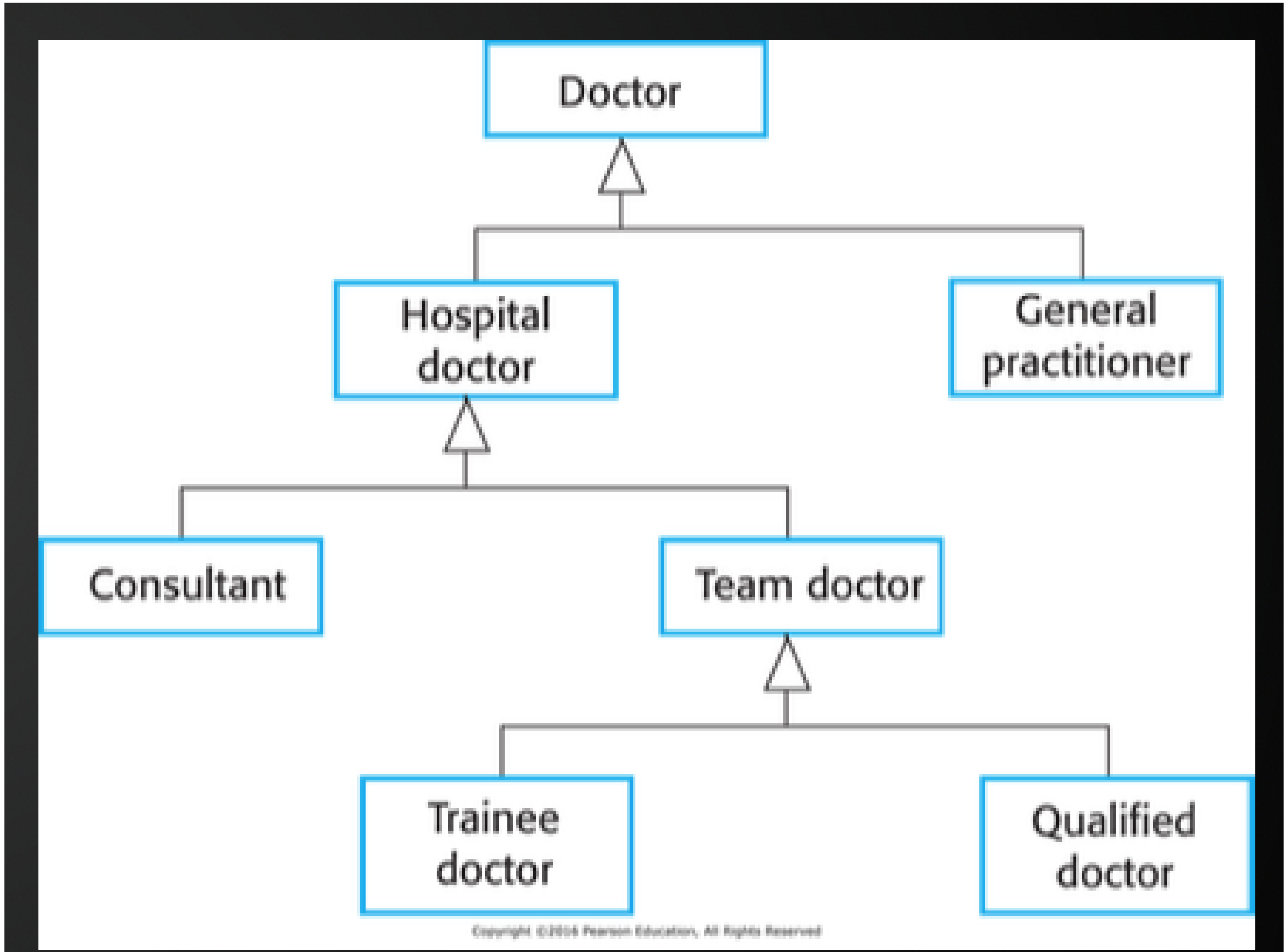
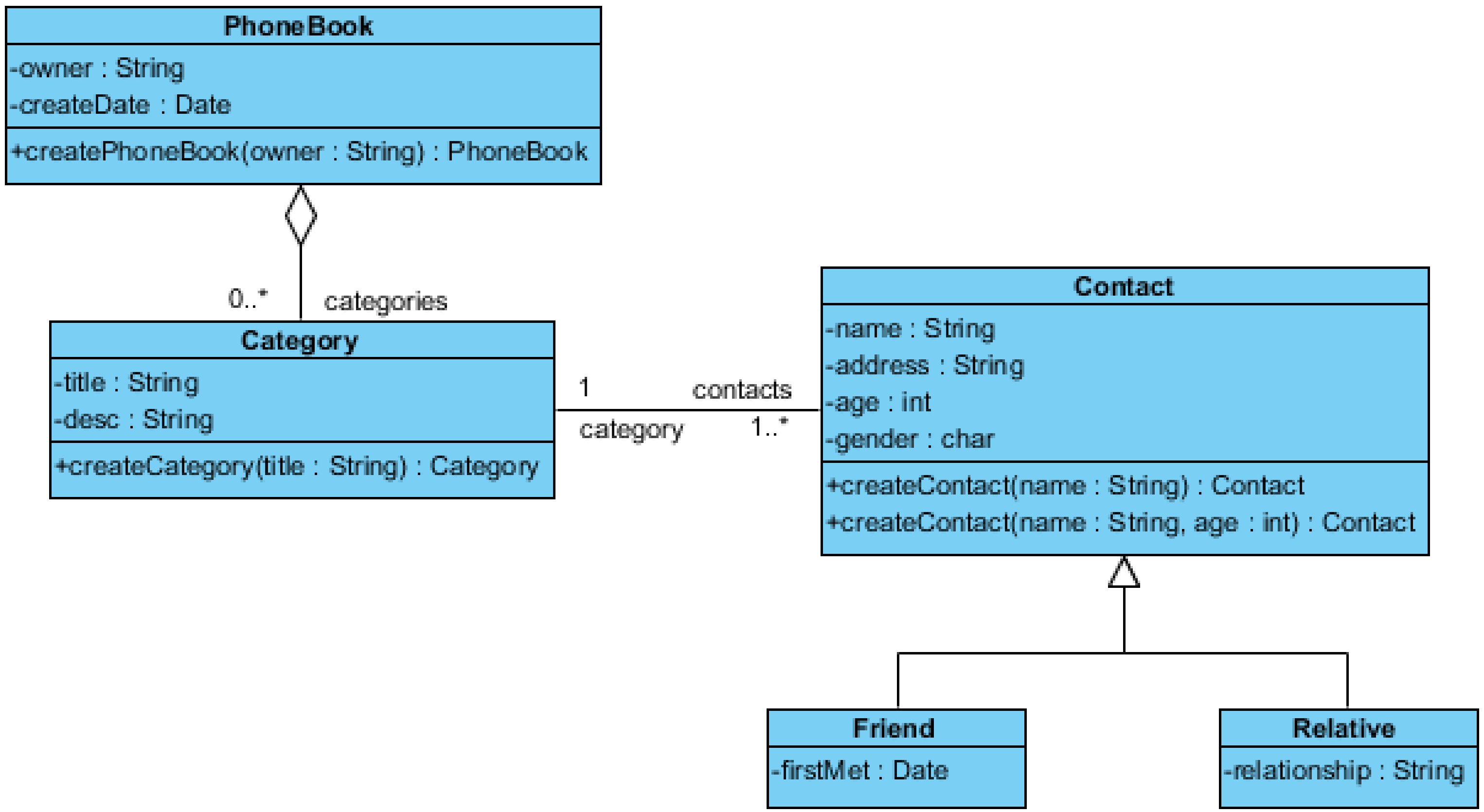
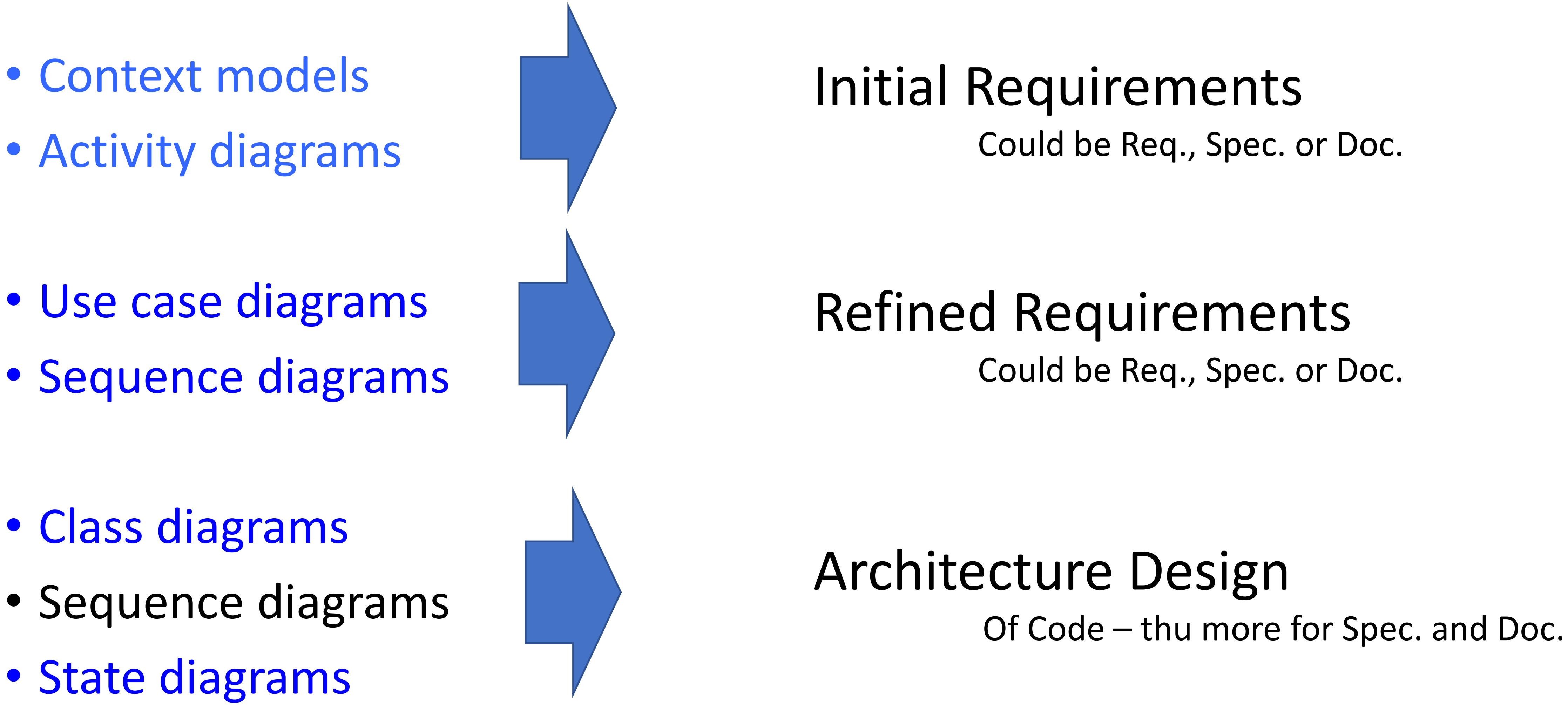


Figure 5.11 A generalization hierarchy

Class Diagrams



UML – Key Diagram Types



- Context models
- Activity diagrams

Initial Requirements

Could be Req., Spec. or Doc.

- Use case diagrams
- Sequence diagrams

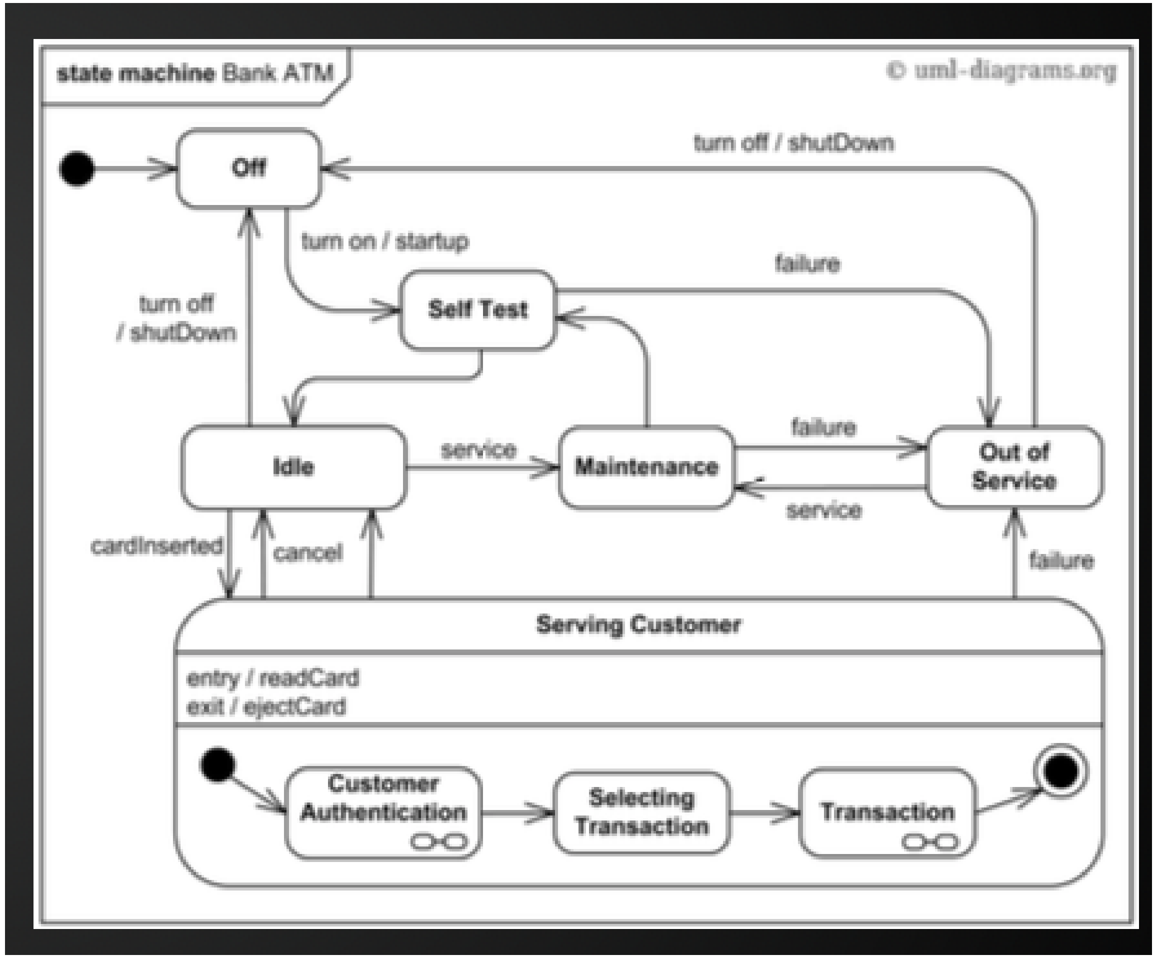
Refined Requirements

Could be Req., Spec. or Doc.

- Class diagrams
- Sequence diagrams
- State diagrams

Architecture Design

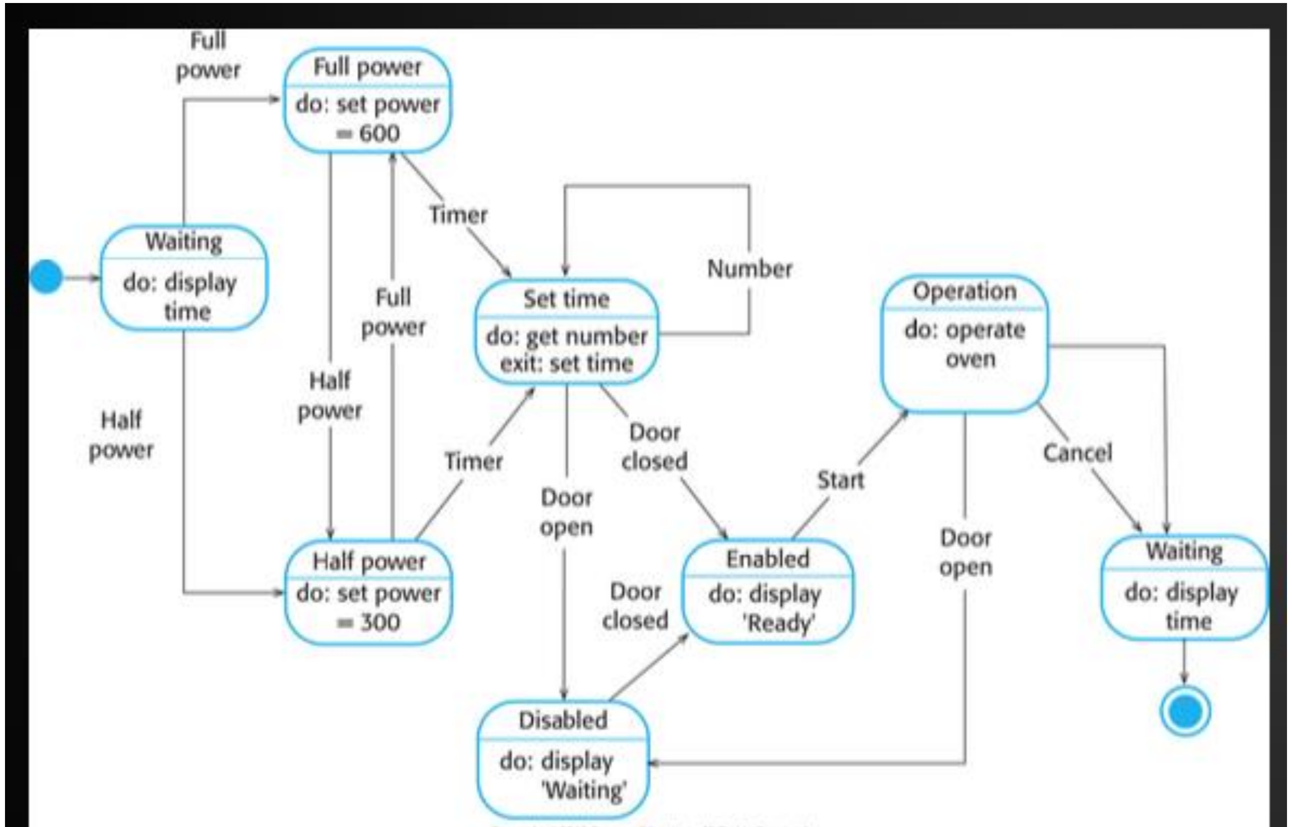
Of Code – thu more for Spec. and Doc.



State Diagrams

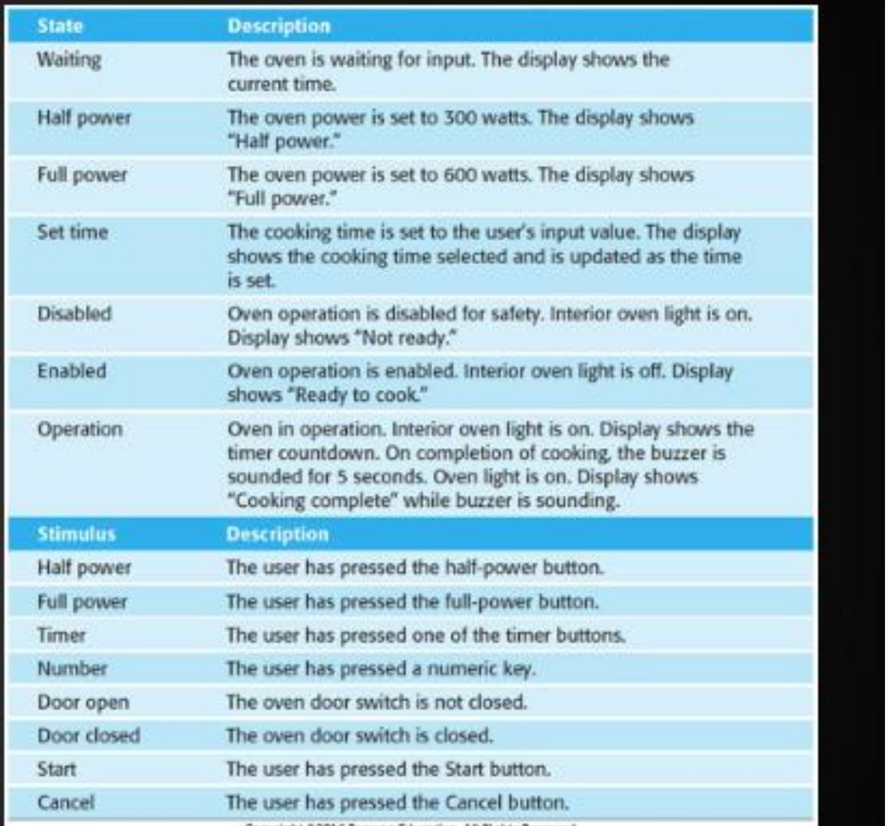
- Bank Example again.
- All the states that "**something**" can be in.
- And what actions change it.
- Could do this with a "module choice form".
- Or "student" going through university:
  - Applicant
  - Successful applicant
  - Studying
  - Suspended
  - Completed

State Diagrams



Copyright ©2016 Pearson Education, All Rights Reserved

Figure 5.16 A state diagram of a microwave oven



State	Description
Waiting	The oven is waiting for input. The display shows the current time.
Half power	The oven power is set to 300 watts. The display shows "Half power."
Full power	The oven power is set to 600 watts. The display shows "Full power."
Set time	The cooking time is set to the user's input value. The display shows the cooking time selected and is updated as the time is set.
Disabled	Oven operation is disabled for safety. Interior oven light is on. Display shows "Not ready."
Enabled	Oven operation is enabled. Interior oven light is off. Display shows "Ready to cook."
Operation	Oven in operation. Interior oven light is on. Display shows the timer countdown. On completion of cooking, the buzzer is sounded for 5 seconds. Oven light is on. Display shows "Cooking complete" while buzzer is sounding.
Stimulus	**Description**
Half power	The user has pressed the half-power button.
Full power	The user has pressed the full-power button.
Timer	The user has pressed one of the timer buttons.
Number	The user has pressed a numeric key.
Door open	The oven door switch is not closed.
Door closed	The oven door switch is closed.
Start	The user has pressed the Start button.
Cancel	The user has pressed the Cancel button.

Copyright ©2016 Pearson Education, All Rights Reserved

Figure 5.18 States and stimuli for the microwave oven

Most Important Things:

1. What do you want to show?
    - A complex decision process.
    - Which stage are you located ?
2. Which diagram or model is best for this?
3. Avoid ”trying to show everything in one diagram”.
    - For a specific diagram, leave out the details and pick out the most salient characteristics.

How to Choose Different Diagrams?

1. At the initial step, we need to define the **system boundaries**.
   - What is or is not part of the system? Context models.
   - How are the systems used? Activity diagrams.

2. Next, how to model the **interactions** between users, internal and external systems?
   - Use case diagrams are drafts, sequence diagrams are applied to explain the specific use case.

3. Most importantly, how to design the **system architecture**?
   - Static view of the software system can be modeled as class diagrams.
   - Dynamic view can be modeled as sequence diagrams if the system are driven by data. It can be modeled as state diagrams if the systems are driven by events.

Scenarios

(Not UML)

Bringing It Together

Scenarios

- Adding Context/Detail to Models/Diagrams.
- Not just a story, though.
- But a structured descripton of process.

“envision and document typical and significant user activities.”

- John Carroll, Five Reasons for Scenario-Based Design

Scenarios

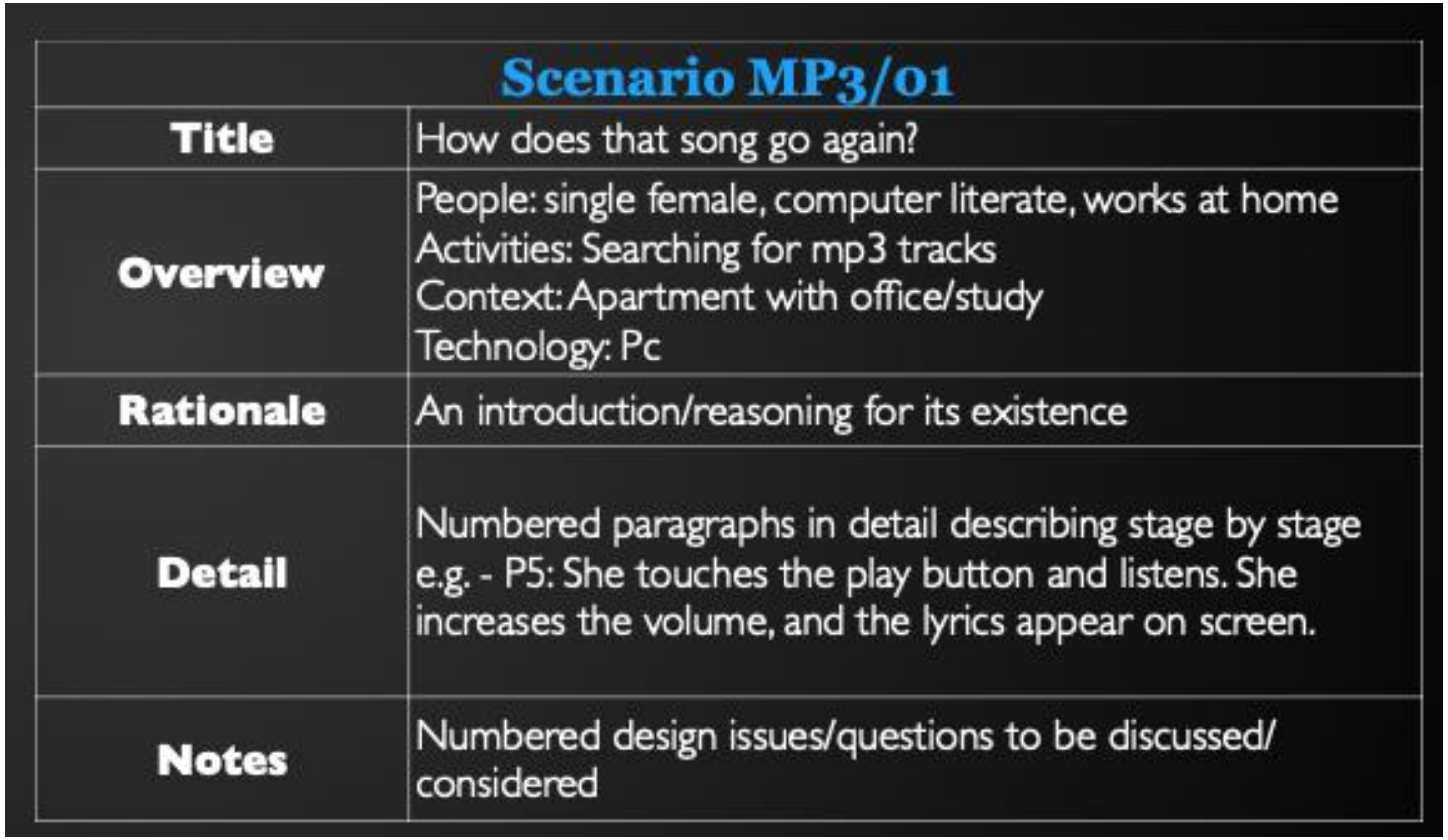
- Must **define a setting** or context.
  - Might come from a technology tour (for example).
  - The environment, the things in it, etc.
- Must define one or more **actors or users**.
  - Perhaps a persona.
- Must define **goals or objectives**.
  - Perhaps from a task analysis or user story.

Scenarios

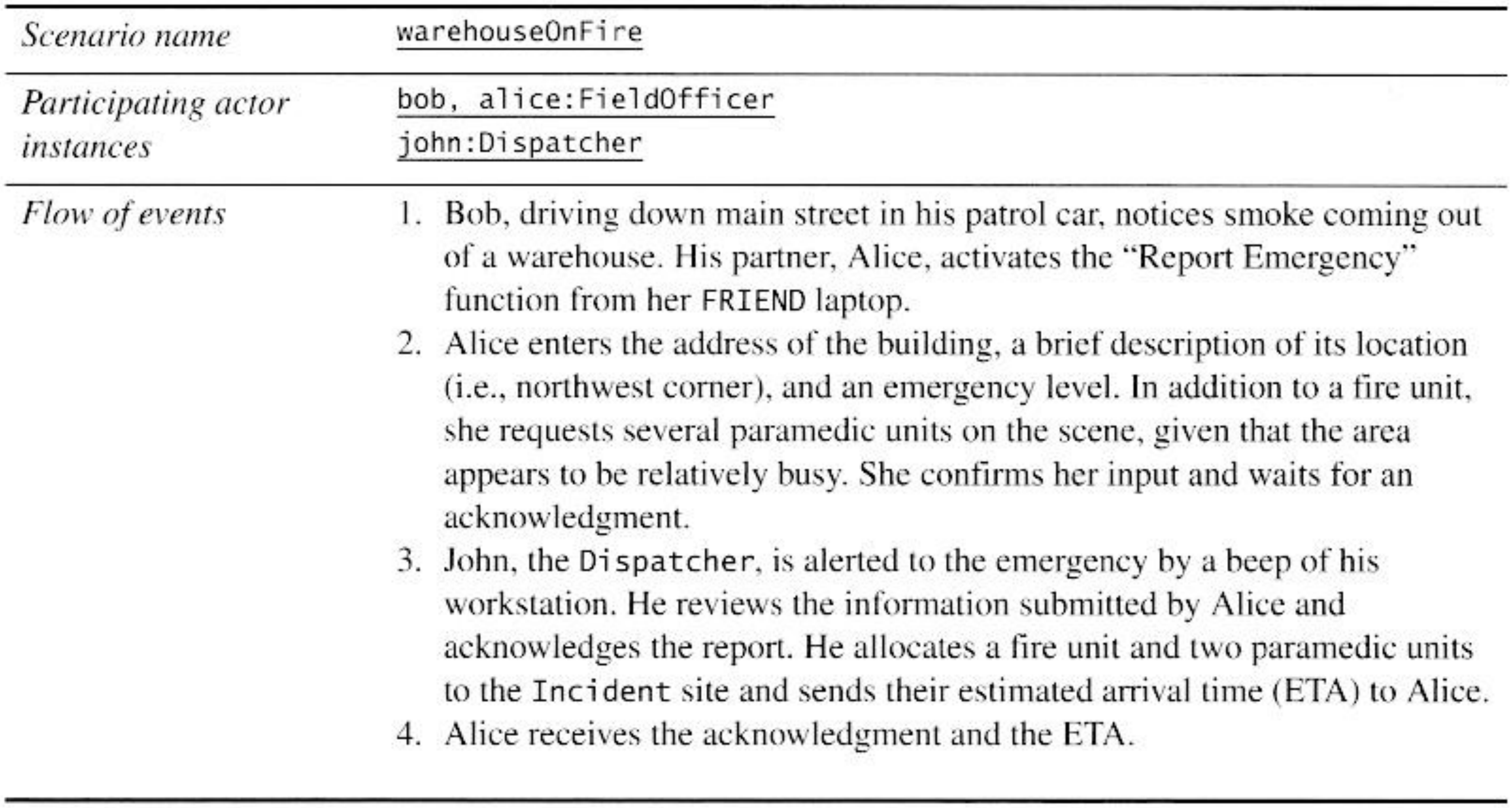
- Must **describe a plot**.
  - As a sequence of events.
  - Perhaps from a task analysis or activity diagram.
- **The plot describes how a user, in a context, achieves a goal**.
- Scenarios are written text descriptions.
  - Describing current events.
  - OR describing ideal or possible events in design.
- Pick ones that *would benefit* from being “illustrated” with detail.
  - Scenario for student picking the simple modules?
  - Scenario for student trying to do a 70/50, including a language from another department?
  - Scenario for a module convenor, considering unusual enrolment requests?
  - Scenario for a student services representative processing a series of standard requests?

Scenarios Example

- Book recommendations for detail section:
  - The “normal” flow of events for the scenario.
- Book recommendations for notes section:
  - What can go wrong and how to fix.
  - Other activities happening in parallel.
  - What happens after.



Scenario MP3/01	
Title	How does that song go again?
Overview	People: single female, computer literate, works at home Activities: Searching for mp3 tracks Context: Apartment with office/study Technology: Pc
Rationale	An introduction/reasoning for its existence
Detail	Numbered paragraphs in detail describing stage by stage e.g. - P5: She touches the play button and listens. She increases the volume, and the lyrics appear on screen.
Notes	Numbered design issues/questions to be discussed/ considered



Scenario name	warehouseOnFire
Participating actor instances	bob, alice:FieldOfficer john:Dispatcher
Flow of events	1. Bob, driving down main street in his patrol car, notices smoke coming out of a warehouse. His partner, Alice, activates the "Report Emergency" function from her FRIEND laptop. 2. Alice enters the address of the building, a brief description of its location (i.e., northwest corner), and an emergency level. In addition to a fire unit, she requests several paramedic units on the scene, given that the area appears to be relatively busy. She confirms her input and waits for an acknowledgment. 3. John, the Dispatcher, is alerted to the emergency by a beep of his workstation. He reviews the information submitted by Alice and acknowledges the report. He allocates a fire unit and two paramedic units to the Incident site and sends their estimated arrival time (ETA) to Alice. 4. Alice receives the acknowledgment and the ETA.

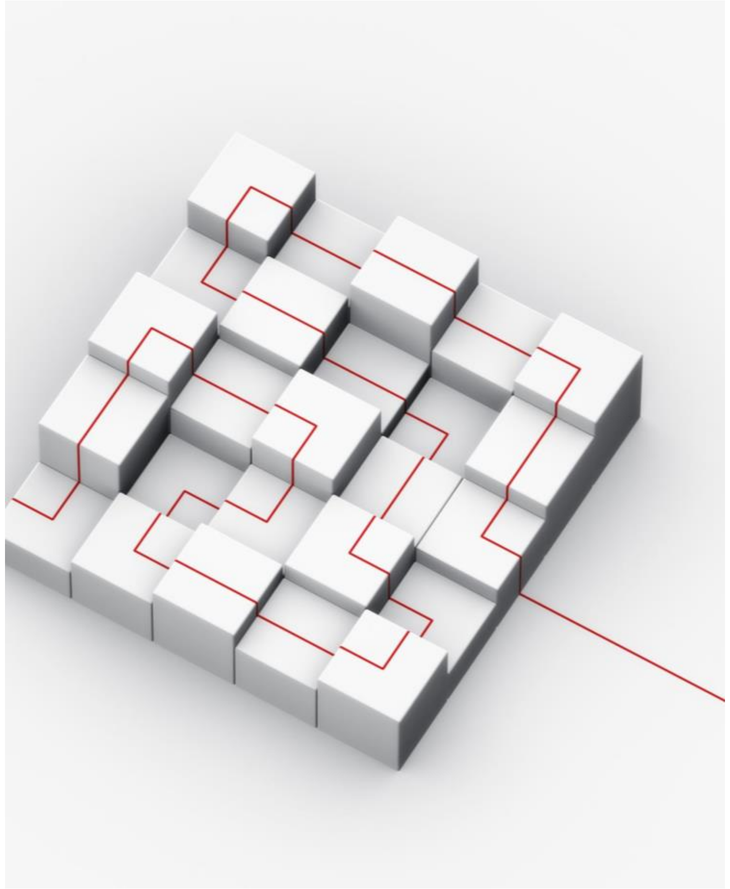
Figure 4-6 warehouseOnFire scenario for the ReportEmergency use case.

Summary

Most Important Thing:

1. What do you want to show?
   - A complex decision process.
   - There are three different stages including initial requirement drafts, refining requirements and architecture design.
   - 5 diagrams from 13 UML diagrams.
   - The structure of a system etc.
2. Which diagram or model is best for this?
   - Which step are you located ?
   - Which diagram is selected?
3. Avoid ”trying to show everything in one diagram”.

- Pick out essential stuff and some details are ignored.



Optional Readings

- UML Diagrams
- Scenarios Method

THANK
YOU