

Requirements Engineering

Software Engineering
Andreas Zeller • Saarland University

Requirements Engineering

The Real World

Requirements Engineering

A description of what the system should do (but not how)

“Requirement”

Standard Glossary of Software Engineering Terminology
(ANSI/IEEE Standard 610.12-1990)

1. A condition or capability needed by a user to solve a problem or achieve an objective.
2. A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed documents.
3. A documented representation of a condition or capability as in (1) or (2).

What requirements are?

- Building blocks we use to describe what the system should do.
- Failure to meet a requirement jeopardizes the system success
- Well documented
- Consistent

Types of Requirements

Types of Requirements



Functional Requirements

- An *action* the product must take to be useful

The product shall allow to track individual payments of coffee servings

Nonfunctional Requirements

- A *property or quality* the product must have

The product shall be accessible in
multiple languages
(such as German and English)

Constraints

- *Global* requirements – on the project or the product

The product shall be available before
March 1st.

Analysis vs Design

- Analysis = what the software should do
 - Software functionality
 - Software properties
- Design = how it should do it

Up-front RE

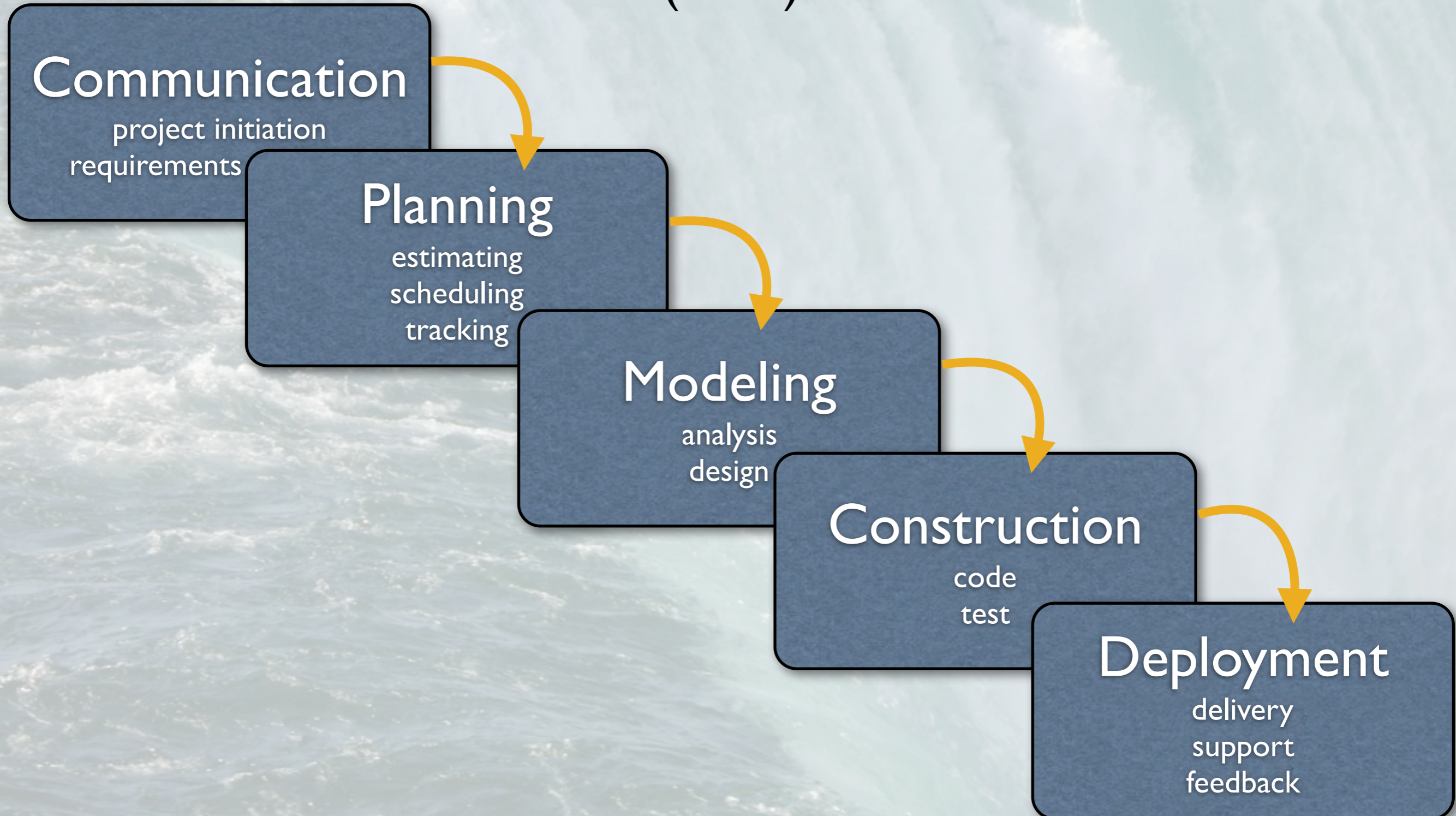
- “We must know [exactly] what to build *before* we can build it”
- classical engineering viewpoint
- leads to...

Waterfall Model

(1968)

Waterfall Model

(1968)



Why requirements are important?

“The hardest single part of building a software system is deciding what to build. No part of the work so cripples the resulting system if done wrong. No other part is more difficult to rectify later”

Fred Brooks

Denver International Airport



DIA: Automated Baggage System



Glass' Law

Requirement deficiencies
are the prime source
of project failures.

Why Requirement Analysis?

- We need to systematically address the problem of getting the right set of requirements
- So, who should I ask first?

Stakeholders

- Persons or organizations who...
 - have a valid interest in the system
 - are affected by the system

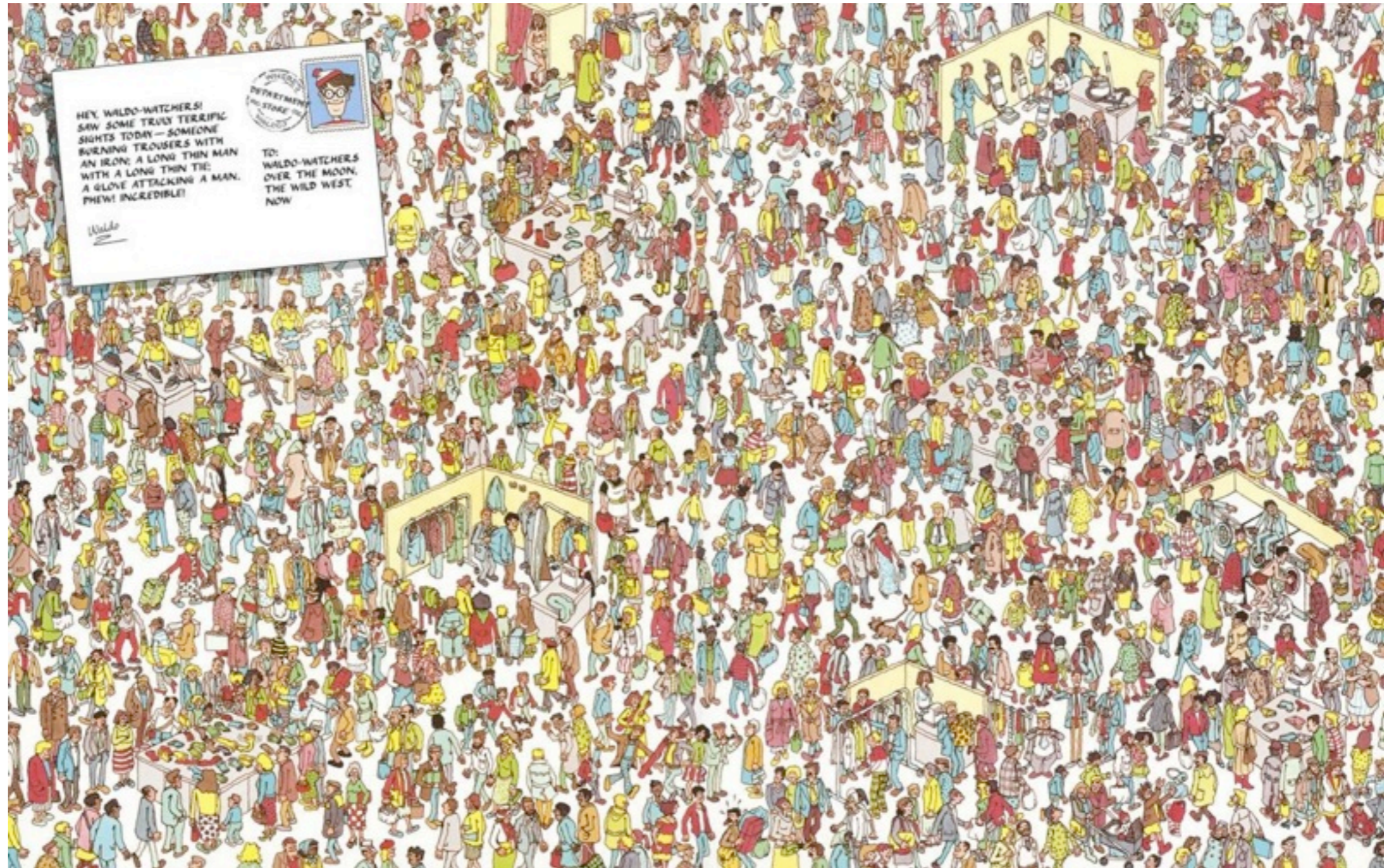
Stakeholders

- anyone who *operates* the system
(normal and maintenance operators)
- anyone who *benefits* from the system
(functional, political, financial and social beneficiaries)
- anyone involved in *purchasing* or *procuring* the system

Stakeholders

- organizations which *regulate* aspects of the system
(financial, safety, and other regulators)
- organizations responsible for systems which *interface* with the system under design
- people or organizations *opposed* to the system
(*negative* stakeholders)

Identify stakeholders



Understand each stakeholder

Bla bla bla, bla bla bla, bla bla
bla Bla bla bla, bla bla bla, bla
bla bla Bla bla bla, bla bla bla,
bla bla bla Bla bla bla, bla bla
bla, bla bla bla Bla bla bla, bla
bla bla, bla bla bla Bla bla bla,
bla bla bla, bla bla bla Bla bla
bla, bla bla bla, bla bla bla Bla
bla bla, bla bla bla, bla bla bla
Bla bla bla, bla bla bla, bla bla
bla Bla bla bla, bla bla bla, bla
bla bla Bla bla bla, bla bla bla,
bla bla bla Bla bla bla, bla bla
bla, bla bla bla Bla bla bla, bla
bla bla, bla bla bla Bla bla bla,
bla bla bla, bla bla bla Bla bla
bla, bla bla bla, bla bla bla

Help stakeholders to understand themselves



Help stakeholders to understand each other



Reach a consensus among stakeholders



“Requirements Analysis”

Standard Glossary of Software Engineering Terminology
(ANSI/IEEE Standard 610.12-1990)

- The process of studying user needs to arrive at a definition of system, hardware, or software requirements.
- The process of studying and refining system, hardware, or software requirements.

I) Inception

- Identify stakeholders
- Recognize multiple viewpoints
- Ask first Q&A

2) Elicitation

- Collaborate with stakeholders
- Address problems (scope/understanding/volatility)
- Method: Collaborative Requirements Gatherings

Problems of *scope*

- What is the boundary of the system?
- What details are actually required?

Problems of understanding

- Users don't know what they want
- ...don't know what is needed
- ...have a poor understanding of their computing environment
- ...don't have full understanding of their domain

Problems of volatility

- Requirements change over time

3) Elaboration

- Expand & refine information
- Document requirements taken during elicitation
 - Use cases
 - Contract style
 - UML diagrams

4) Negotiation

- Reconcile conflicts
- Ranking of requirements
- Risk estimation (very rough)

5) Specification

- Written document
 - Natural Language +
 - Graphical models

6) Validation

- Search for
 - Inconsistencies
 - Omissions
 - Errors
- Technical Review
 - Review Team

7) Requirements Management

- Control Requirements change
- Trace requirements to:
 - Features
 - Code
 - Subsystems
- How Requirements are related to each other

Collaborative Requirement Gathering



Collaborative Requirement Gathering

- Meetings attended by both customers and software engineers (+other stakeholders)
- Rules for preparation and participation are established
- Agenda is suggested
- A facilitator controls the meeting
- A “definition mechanism” is used

Collaborative Requirement Gathering

- Goal:
 - Identify problem
 - Propose elements of solution
 - Negotiate approaches
 - Specify preliminary set of requirements

Documenting Requirements

- Contract-style requirements
- Use cases (user stories)
- UML-diagrams
- Paper prototyping

Contract Style

| Requirement | Comment |
|---|--|
| <hr/> <p>The system will support client inquiries from four access points: in person, paper-based mail, voice communication, and electronic communication (Internet, dial-up, and LAN/WAN).</p> | <p>Four access points are how; we should focus instead on who needs access from where.</p> |
| <hr/> <p>The telephone system must be able to support an 800 number system.</p> | <p>An 800 number? Can't use 888 or 877? Again, what's missing is who needs what kind of access from where.</p> |
| <hr/> <p>The telephone system must be able to handle 97,000 calls per year and must allow for a growth rate of 15 percent annually. Of these calls it is estimated that 19 percent will be responded to in an automated manner and 81 percent will be routed to call center staff for response. Fifty percent of the calls can be processed without reference to the electronic copy of the paper file, and approximately 50 percent will require access to the system files.</p> <hr/> | <p>Valuable statistics; this one is actually pretty good.</p> |

Contract Style

Classify product features as

- *Must-have* features
“The product must conform to accessibility guidelines”
- *May-have* features
“The product may eventually be voice-controlled”
- *Must-not-have* features
“The product supports only one language”

Be explicit about *must-not-have* features!

Contract Style

- Provide a *contract* between sponsors and developers
- Can run to *hundreds of pages*
- Abstract all requirements, with little context

Contract Style



love it



hate it

Use Case

- An *actor* is something that can act – a person, a system, or an organization
- A *scenario* is a specific sequence of *actions* and *interactions* between actors (where at least one actor is a system)
- A *use case* is a collection of related scenarios – successful and failing ones
- Useful for *clients* as well as for *developers*

Actors and Goals

- What are the *boundaries* of the system? Is it the software, hardware and software, also the user, or a whole organization?
- Who are the *primary actors* – i.e., the stakeholders?
- What are the *goals* of these actors?
- Describe how the system fulfills these goals (including all exceptions)

Example: SafeHome



Initial Scenario

Use case: display camera views

Actor: homeowner

If I'm at a remote location, I can use any PC with appropriate browser software to log on to the SafeHome Web site. I enter my user ID and two levels of passwords and, once I'm validated, I have access to all the functionality. To access a specific camera view, I select "surveillance" and then "select a camera". Alternatively, I can look at thumbnail snapshots from all cameras by selecting "all cameras". Once I choose a camera, I select "view"...

Refined Scenario

Use case: display camera views

Actor: homeowner

- 1. The homeowner logs on to the Web Site*
- 2. The homeowner enters his/her user ID*
- 3. The homeowner enters two passwords*
- 4. The system displays all major function buttons*
- 5. The homeowner selects “surveillance” button*
- 6. The homeowner selects “Pick a camera”...*

Alternative Interactions

- Can the actor take some other action at this point?
- Is it possible that the actor encounters some error condition? If so, which one?
- Is it possible that some other behavior is encountered? If so, which one?

*Exploring alternatives is the key
to successful requirements analysis!*



Use-Case Template for Surveillance

Use-case: Access camera surveillance—display camera views (ACS-DCV).

Primary actor: Homeowner.

Goal in context: To view output of camera placed throughout the house from any remote location via the Internet.

Preconditions: System must be fully configured; appropriate user ID and passwords must be obtained.

Trigger: The homeowner decides to take a look inside the house while away.

Scenario:

1. The homeowner logs onto the *SafeHome Products* Web site.
2. The homeowner enters his or her user ID.
3. The homeowner enters two passwords (each at least eight characters in length).
4. The system displays all major function buttons.
5. The homeowner selects "surveillance" from the major function buttons.
6. The homeowner selects "pick a camera."
7. The system displays the floor plan of the house.
8. The homeowner selects a camera icon from the floor plan.

9. The homeowner selects the "view" button.
10. The system displays a viewing window that is identified by the camera ID.
11. The system displays video output within the viewing window at one frame per second.

Exceptions:

1. ID or passwords are incorrect or not recognized—see use-case: "validate ID and passwords."
2. Surveillance function not configured for this system—system displays appropriate error message; see use-case: "configure surveillance function."
3. Homeowner selects "view thumbnail snapshots for all cameras"—see use-case: "view thumbnail snapshots for all cameras."
4. A floor plan is not available or has not been configured—display appropriate error message and see use-case: "configure floor plan."
5. An alarm condition is encountered—see use-case: "alarm condition encountered."

Priority: Moderate priority, to be implemented after basic functions.

When available: Third increment.

Frequency of use: Infrequent.



Use-Case Template for Surveillance

Use-case: Access camera surveillance—display camera views (ACS-DCV).

Primary actor: Homeowner.
Goal in context: To view output of camera placed throughout the house from any remote location via the Internet.
Preconditions: System must be fully configured; appropriate user ID and passwords must be obtained.
Trigger: The homeowner decides to take a look inside the house while away.

Scenario:

1. The homeowner logs onto the *SafeHome Products* Web site.
2. The homeowner enters his or her user ID.
3. The homeowner enters two passwords (each at least eight characters in length).
4. The system displays all major function buttons.
5. The homeowner selects "surveillance" from the major function buttons.
6. The homeowner selects "pick a camera."
7. The system displays the floor plan of the house.
8. The homeowner selects a camera icon from the floor plan.

9. The homeowner selects the "view" button.
10. The system displays a viewing window that is identified by the camera ID.
11. The system displays video output within the viewing window at one frame per second.

Exceptions:

1. ID or passwords are incorrect or not recognized—see use-case: "validate ID and passwords."
2. Surveillance function not configured for this system—system displays appropriate error message; see use-case: "configure surveillance function."
3. Homeowner selects "view thumbnail snapshots for all cameras"—see use-case: "view thumbnail snapshots for all cameras."
4. A floor plan is not available or has not been configured—display appropriate error message and see use-case: "configure floor plan."
5. An alarm condition is encountered—see use-case: "alarm condition encountered."

Priority: Moderate priority, to be implemented after basic functions.

When available: Third increment.

Frequency of use: Infrequent.



Use-Case Template for Surveillance

Use-case: Access camera surveillance—display camera views (ACS-DCV).

Primary actor: Homeowner.
Goal in context: To view output of camera placed throughout the house from any remote location via the Internet.
Preconditions: System must be fully configured; appropriate user ID and passwords must be obtained.
Trigger: The homeowner decides to take a look inside the house while away.

Scenario:

1. The homeowner logs onto the *SafeHome Products* Web site.
2. The homeowner enters his or her user ID.
3. The homeowner enters two passwords (each at least eight characters in length).
4. The system displays all major function buttons.
5. The homeowner selects "surveillance" from the major function buttons.
6. The homeowner selects "pick a camera."
7. The system displays the floor plan of the house.
8. The homeowner selects a camera icon from the floor plan.

9. The homeowner selects the "view" button.
10. The system displays a viewing window that is identified by the camera ID.
11. The system displays video output within the viewing window at one frame per second.

Exceptions:

1. ID or passwords are incorrect or not recognized—see use-case: "validate ID and passwords."
2. Surveillance function not configured for this system—system displays appropriate error message; see use-case: "configure surveillance function."
3. Homeowner selects "view thumbnail snapshots for all cameras"—see use-case: "view thumbnail snapshots for all cameras."
4. A floor plan is not available or has not been configured—display appropriate error message and see use-case: "configure floor plan."
5. An alarm condition is encountered—see use-case: "alarm condition encountered."

Priority: Moderate priority, to be implemented after basic functions.

When available: Third increment.

Frequency of use: Infrequent.

SAFEHOME



Use-Case Template for Surveillance

Use-case: Access camera

surveillance—display camera views (ACS-DCV).

Primary actor:

Homeowner.

Goal in context:

To view output of camera placed throughout the house from any remote location via the Internet.

Preconditions:

System must be fully configured; appropriate user ID and passwords must be obtained.

Trigger:

The homeowner decides to take a look inside the house while away.

Scenario:

1. The homeowner logs onto the *SafeHome Products* Web site.
2. The homeowner enters his or her user ID.
3. The homeowner enters two passwords (each at least eight characters in length).
4. The system displays all major function buttons.
5. The homeowner selects "surveillance" from the major function buttons.
6. The homeowner selects "pick a camera."
7. The system displays the floor plan of the house.
8. The homeowner selects a camera icon from the floor plan.
9. The homeowner selects the "view" button.
10. The system displays a viewing window that is identified by the camera ID.
11. The system displays video output within the viewing window at one frame per second.

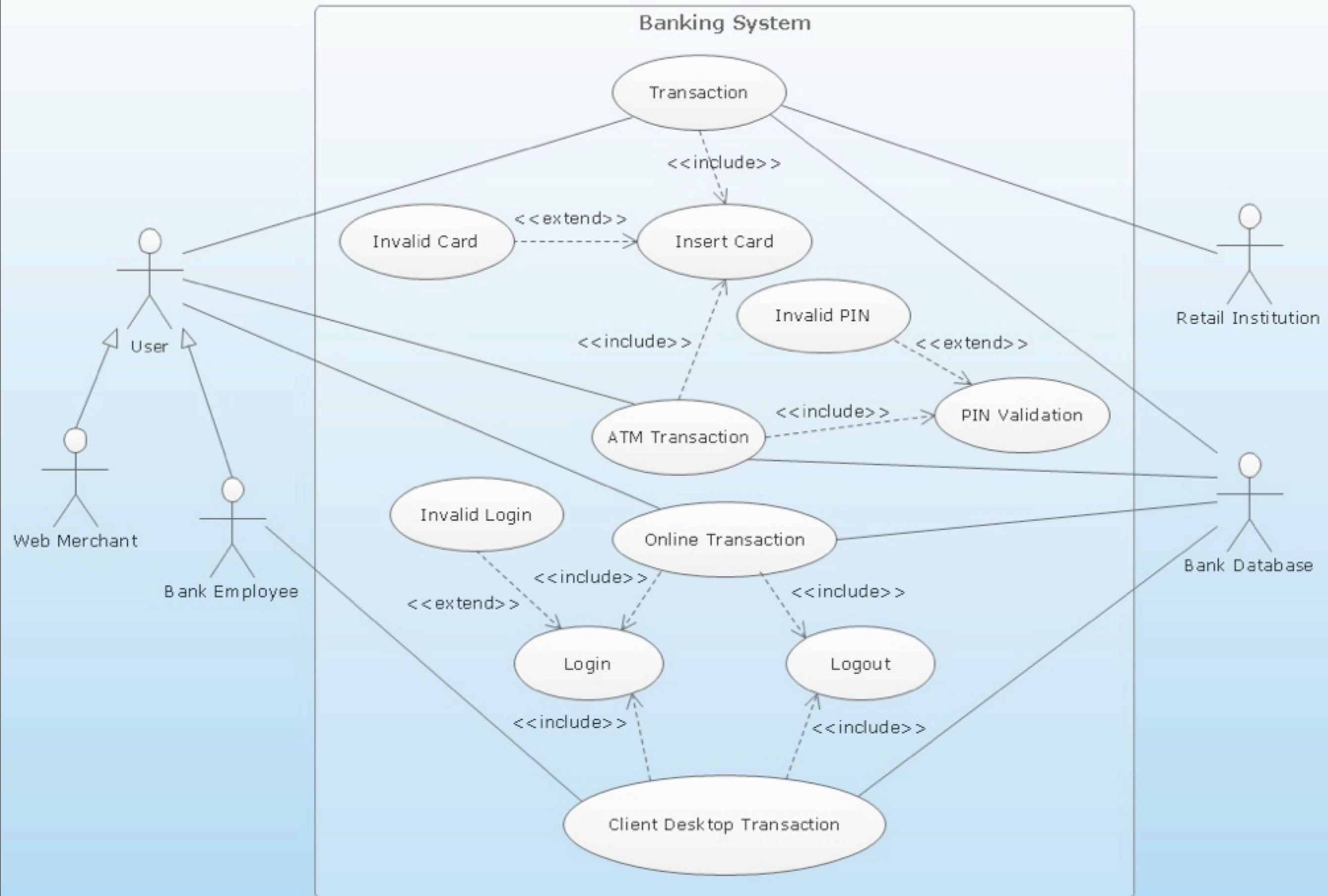
Exceptions:

1. ID or passwords are incorrect or not recognized—see use-case: “validate ID and passwords.”
2. Surveillance function not configured for this system—system displays appropriate error message; see use-case: “configure surveillance function.”
3. Homeowner selects “view thumbnail snapshots for all cameras”—see use-case: “view thumbnail snapshots for all cameras.”
4. A floor plan is not available or has not been configured—display appropriate error message and see use-case: “configure floor plan.”
5. An alarm condition is encountered—see use-case: “alarm condition encountered.”

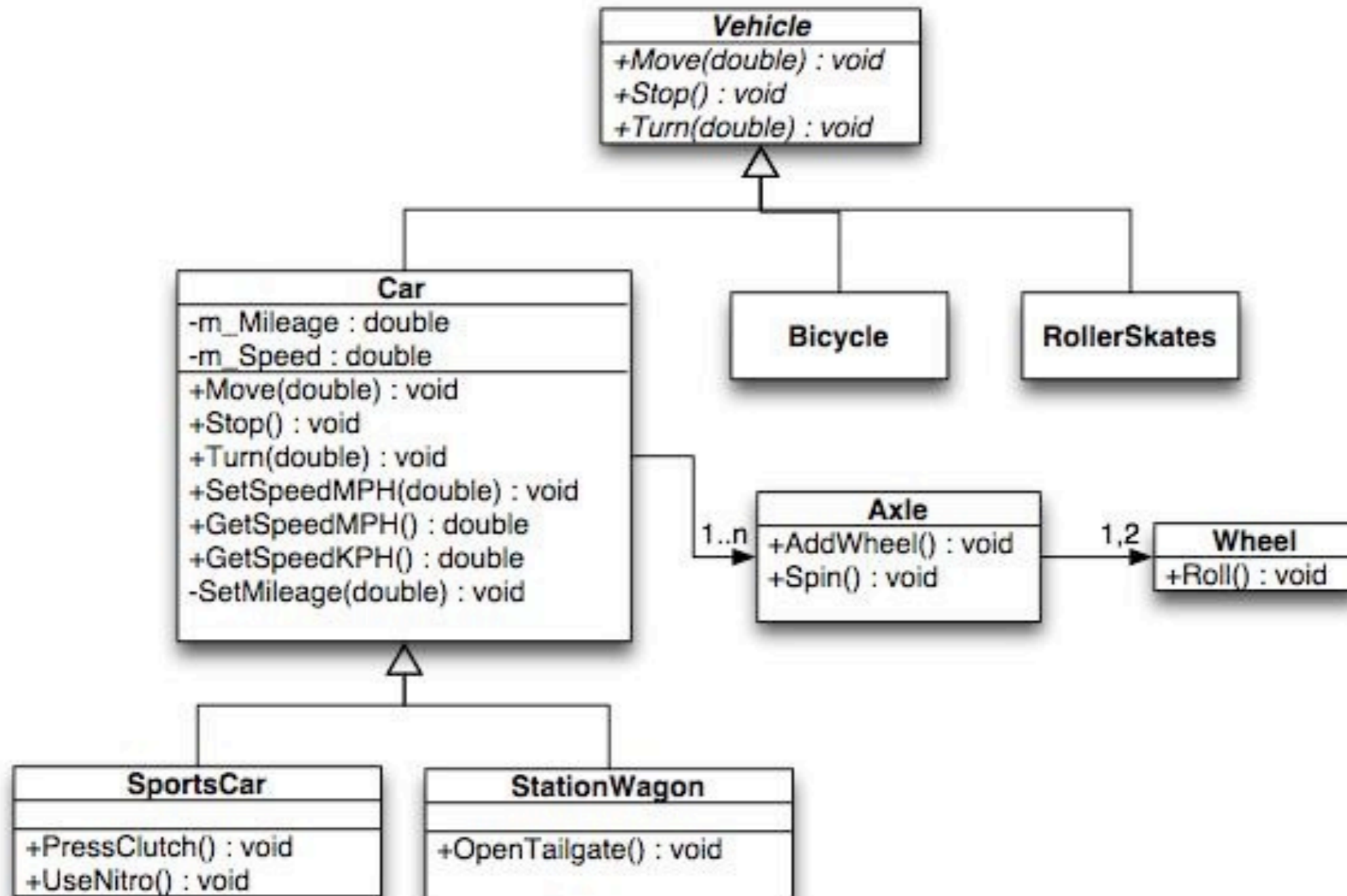
UML Diagrams

- UML: Unified Modeling Language
- Graphical models useful for communicating ideas (both stakeholders and engineers)
- UML class diagram: static snapshot of system's relationships (classes, attributes, operations)
- UML sequence diagram: how processes interact

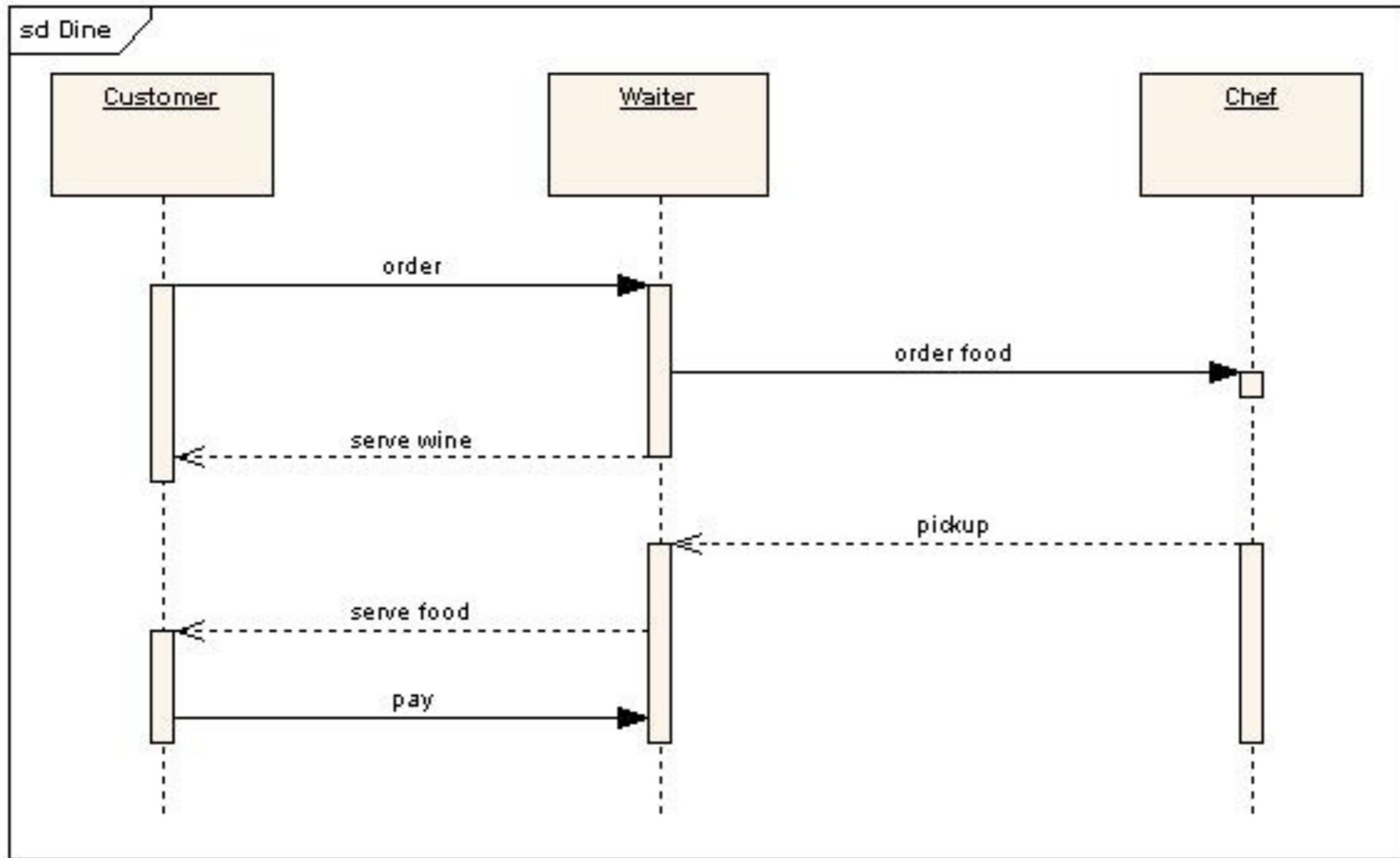
UML Use Case Diagram



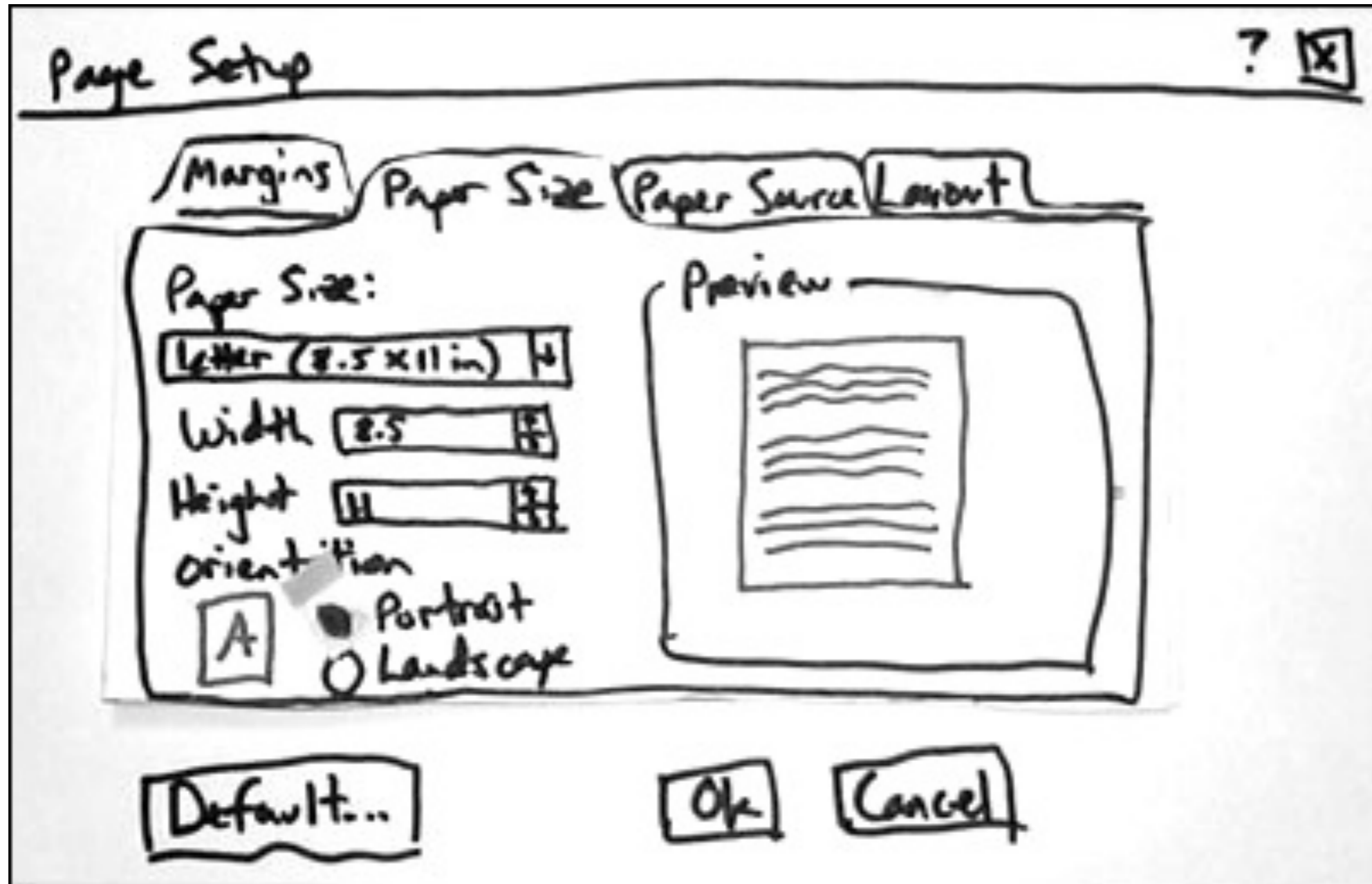
UML class diagram



UML sequence diagram



Paper prototyping



Paper prototyping



Summary

“Requirement”

Standard Glossary of Software Engineering Terminology
(ANSI/IEEE Standard 610.12-1990)

1. A condition or capability needed by a user to solve a problem or achieve an objective.
2. A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed documents.
3. A documented representation of a condition or capability as in (1) or (2).

Summary

“Requirement”

Standard Glossary of Software Engineering Terminology
(ANSI/IEEE Standard 610.12-1990)

1. A condition or capability needed by a user to solve a problem or achieve an objective.
2. A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed documents.
3. A documented representation of a condition or capability as in (1) or (2).

Contract Style

| Requirement | Comment |
|--|---|
| The system will support client inquiries from four access points: in person, paper-based mail, voice communication, and electronic communication (Internet, dial-up, and LAN/WAN). | Four access points are how; we should focus instead on who needs access from where. |
| The telephone system must be able to support an 800 number system. | An 800 number? Can't use 888 or 877? Again, what's missing is who needs what kind of access from where. |
| The telephone system must be able to handle 97,000 calls per year and must allow for a growth rate of 15 percent annually. Of these calls it is estimated that 19 percent will be responded to in an automated manner and 81 percent will be routed to call center staff for response. Fifty percent of the calls can be processed without reference to the electronic copy of the paper file, and approximately 50 percent will require access to the system files. | Valuable statistics; this one is actually pretty good. |

Summary

“Requirement”

Standard Glossary of Software Engineering Terminology
(ANSI/IEEE Standard 610.12-1990)

1. A condition or capability needed by a user to solve a problem or achieve an objective.
2. A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed documents.
3. A documented representation of a condition or capability as in (1) or (2).

Contract Style

| Requirement | Comment |
|--|---|
| The system will support client inquiries from four access points: in person, paper-based mail, voice communication, and electronic communication (Internet, dial-up, and LAN/WAN). | Four access points are how; we should focus instead on who needs access from where. |
| The telephone system must be able to support an 800 number system. | An 800 number? Can't use 888 or 877? Again, what's missing is who needs what kind of access from where. |
| The telephone system must be able to handle 97,000 calls per year and must allow for a growth rate of 15 percent annually. Of these calls it is estimated that 19 percent will be responded to in an automated manner and 81 percent will be routed to call center staff for response. Fifty percent of the calls can be processed without reference to the electronic copy of the paper file, and approximately 50 percent will require access to the system files. | Valuable statistics; this one is actually pretty good. |

Summary

SAFEHOME

Use-Case Template for Surveillance

Use Case

Use-case: Access to surveillance—display (ACS-DCV).

Primary actor: Homeowner.

Goal in context: To view output of camera placed throughout the house from any remote location via the Internet.

Preconditions: System must be fully configured; appropriate user ID and passwords must be obtained.

Trigger: The homeowner decides to take a look inside the house while away.

Scenario:

1. The homeowner logs onto the SafeHome Products Web site.
2. The homeowner enters his or her user ID.
3. The homeowner enters two passwords (each at least eight characters in length).
4. The system displays all major function buttons.
5. The homeowner selects “surveillance” from the major function buttons.
6. The homeowner selects “pick a camera.”
7. The system displays the floor plan of the house.
8. The homeowner selects a camera icon from the floor plan.
9. The homeowner selects the “view” button.

i. The system displays a viewing window that is identified by the camera ID.

ii. The system displays video output within the viewing window at one frame per second.

Exceptions:

1. ID or passwords are incorrect or not recognized—see use-case: “validate ID and passwords.”
2. Surveillance function not configured for this system—system displays appropriate error message; see use-case: “configure surveillance function.”
3. Homeowner selects “view thumbnail snapshots for all cameras”—see use-case: “view thumbnail snapshots for all cameras.”
4. A floor plan is not available or has not been configured—display appropriate error message and see use-case: “configure floor plan.”
5. An alarm condition is encountered—see use-case: “alarm condition encountered.”

Priority: Moderate priority, to be implemented after basic functions.

When available: Third increment.

Frequency of use: Infrequent.