CS342 Software Engineering

Dr. Ismail Hababeh
German Jordanian University

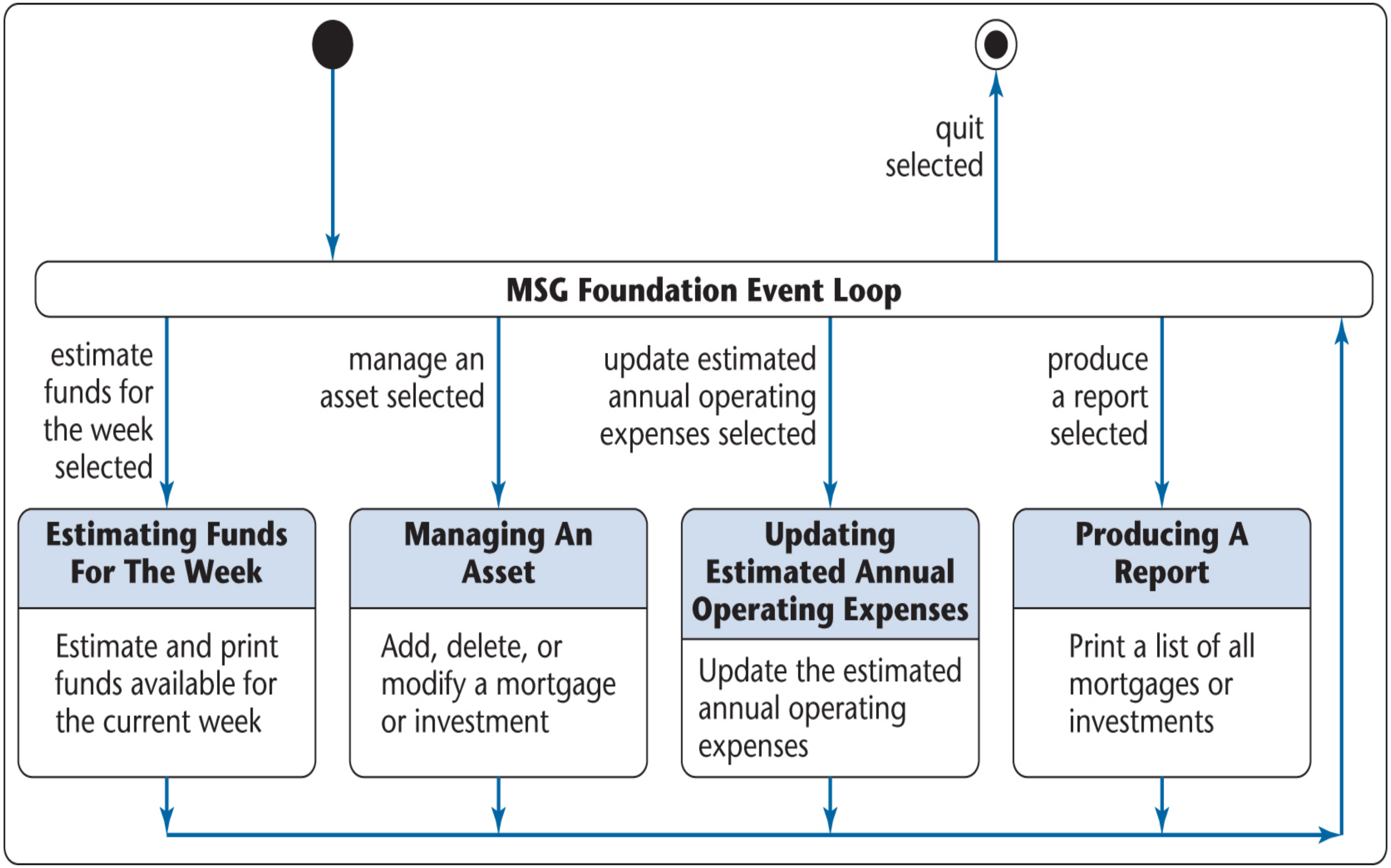
Lecture 14
OBJECT-ORIENTED ANALYSIS

Adapted from Software Engineering, by Dr. Paul E. Young
& slides by Dr. Mohammad Daoud

The Initial Dynamic Model - MSG Case Study

- Dynamic modeling is the third step in extracting the entity classes
- A statechart is constructed to reflect all operations performed by the software product
- The operations are extracted from scenarios

The Initial Dynamic Model: MSG Statechart


quit selected
MSG Foundation Event Loop
estimate funds for the week selected
manage an asset selected
update estimated annual operating expenses selected
produce a report selected
Estimating Funds For The Week
Estimate and print funds available for the current week
Managing An Asset
Add, delete, or modify a mortgage or investment
Updating Estimated Annual Operating Expenses
Update the estimated annual operating expenses
Producing A Report
Print a list of all mortgages or investments

The Initial Dynamic Model - MSG Statechart

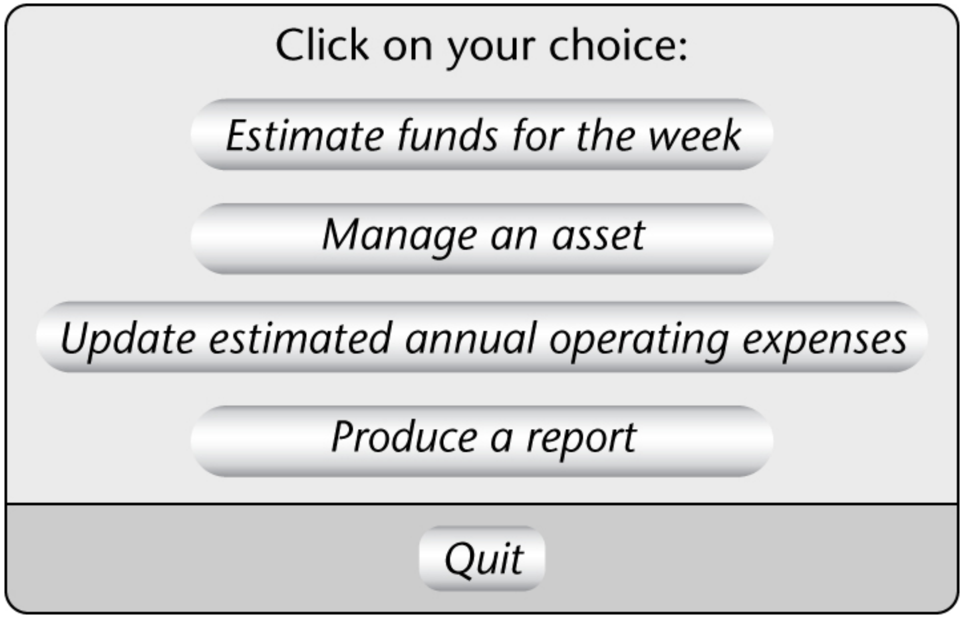
- The statechart reflects the operations of the complete MSG information system:
  - The solid circle on the top left represents the initial state, the starting point of the statechart
  - The white circle containing the small black circle on the top right represents the final state
  - Operation states are represented by rounded rectangles.
  - The arrows represent possible transitions from state to state

The Initial Dynamic Model - MSG Statechart

- In the statechart of the MSG (Event Loop), one of five events (options) can occur:
  1. Estimate funds for the week
  2. Manage an asset
  3. Update estimated annual operating expenses
  4. Produce a report
  5. Quit
- An event causes a transition between states

The Initial Dynamic Model - MSG Statechart

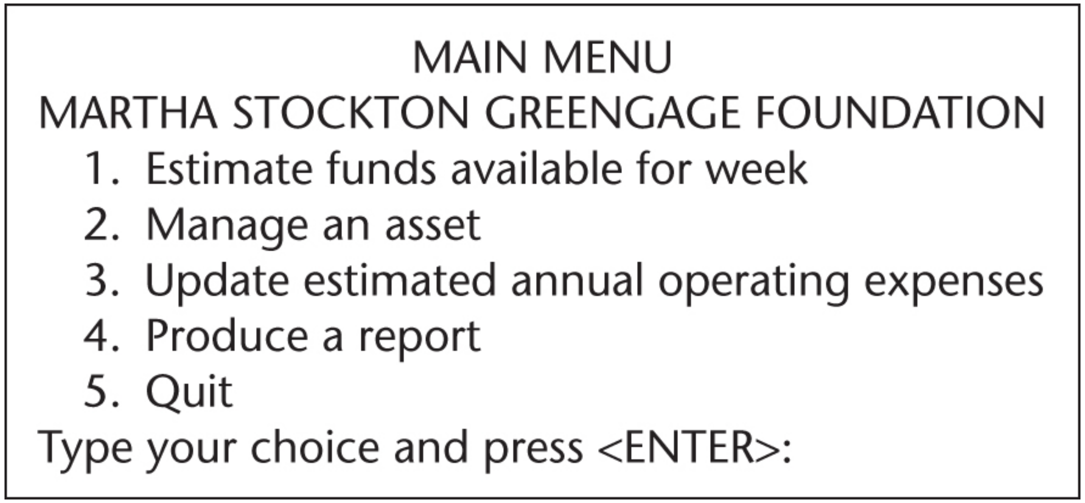
- An MSG staff member selects an option by clicking on the buttons of the GUI


Click on your choice:
Estimate funds for the week
Manage an asset
Update estimated annual operating expenses
Produce a report
Quit

- This graphical user interface (GUI) requires special software

The Initial Dynamic Model - MSG Statechart

- Equivalent textual interface that can run on any computer

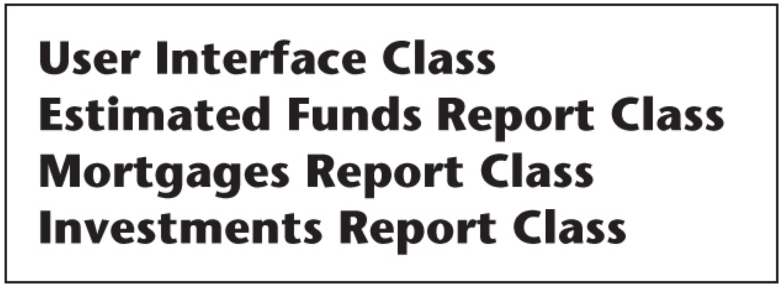


```
MAIN MENU
MARTHA STOCKTON GREENGAGE FOUNDATION
   1.  Estimate funds available for week
   2.  Manage an asset
   3.  Update estimated annual operating expenses
   4.  Produce a report
   5.  Quit
Type your choice and press <ENTER>:
```

Extracting the Boundary Classes

- Each one of the
  - Input screen
  - Output screen
  - Report

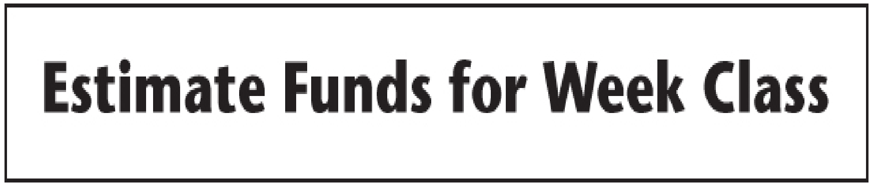
  is modeled by its own boundary class. Here are the four initial boundary classes



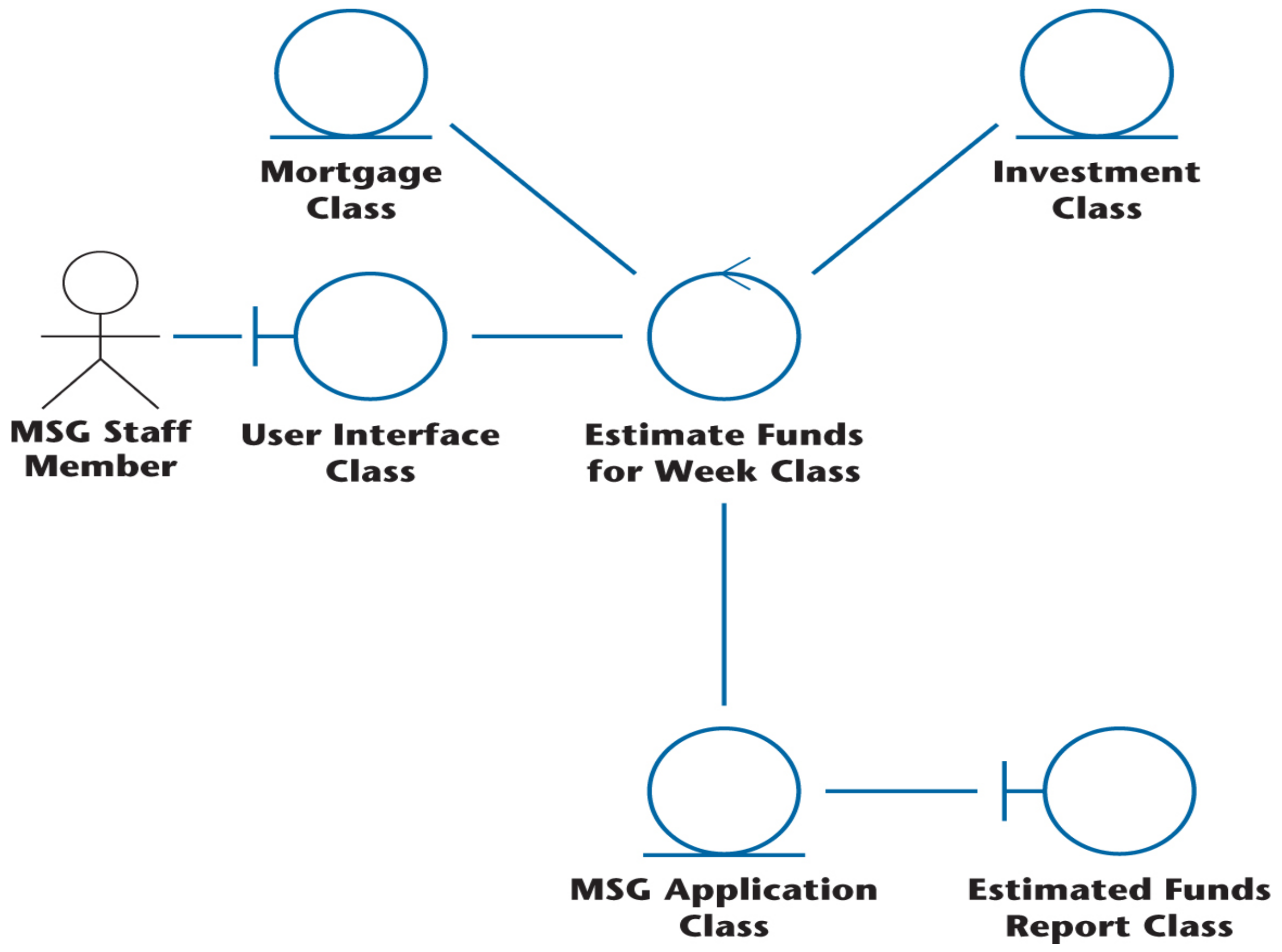
User Interface Class
Estimated Funds Report Class
Mortgages Report Class
Investments Report Class

Extracting the Control Classes - MSG Case Study

- Each nontrivial computation is modeled by a control class
- Each computation is usually modeled by a control class
- The MSG case study has just one non-trivial computation
  - *Estimate the funds available for the week*
- There is one initial control class



Class Diagram for MSG Use Cases



Class Diagram Description - MSG

- **User Interface Class (Boundary)**
  - This class models the user interface
- **Estimate Funds for Week Class (Control)**
  - This control class models the computation of the estimate of the funds that are available to fund mortgages during that week
- **Mortgage Class (Entity)**
  - This class models the estimated grants and payments for the week

Class Diagram Description – MSG

- **Investment Class (Entity)**
  - This class models the estimated return on investments for the week
- **MSG Application Class (Entity)**
  - This class models the estimated operating expresses for the week
- **Estimated Funds Report Class (Boundary)**
  - This class models the printing of the report

Class Diagram Disadvantage

- A class diagram shows the classes of the use cases and their relationships.

But:

- A working information system uses objects, not classes
  - Example: A specific mortgage is represented by **Mortgage object**.
- A class diagram doesn't show the objects nor the sequence of messages as they are sent between objects.

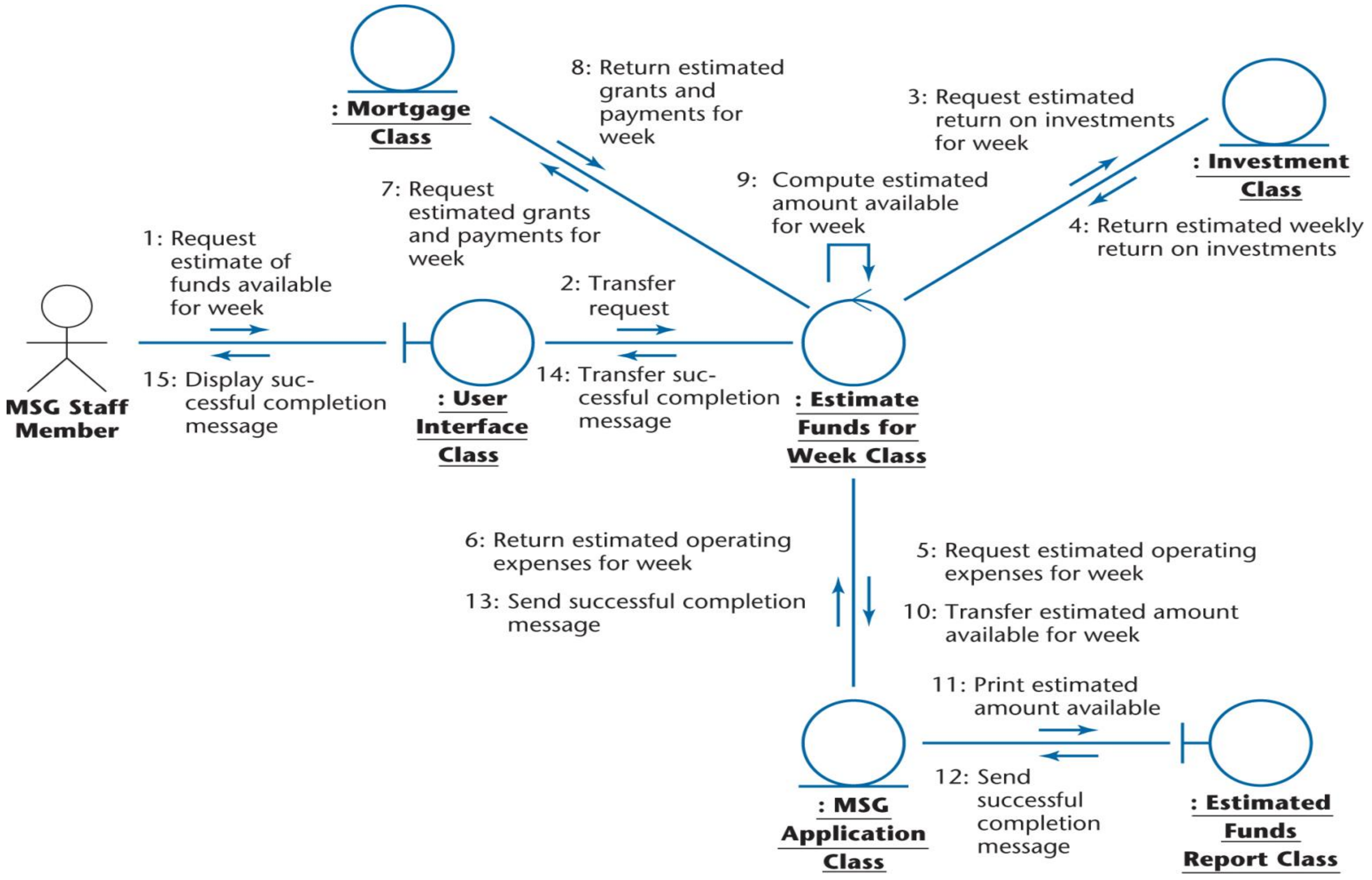
Solution…Interaction Diagrams

- Two types of Interaction Diagrams:
  - Collaboration Diagram
  - Sequence Diagram
- The strength of the Interaction Diagrams
  - it shows the classes and the flow of messages and their order in clearer way.

Collaboration Diagram

- Collaboration Diagram is a realization of the scenarios' use cases.
- The Collaboration Diagram shows the objects as well as the messages, numbered in the order in which they are sent in the specific scenario.

Collaboration Diagram - MSG

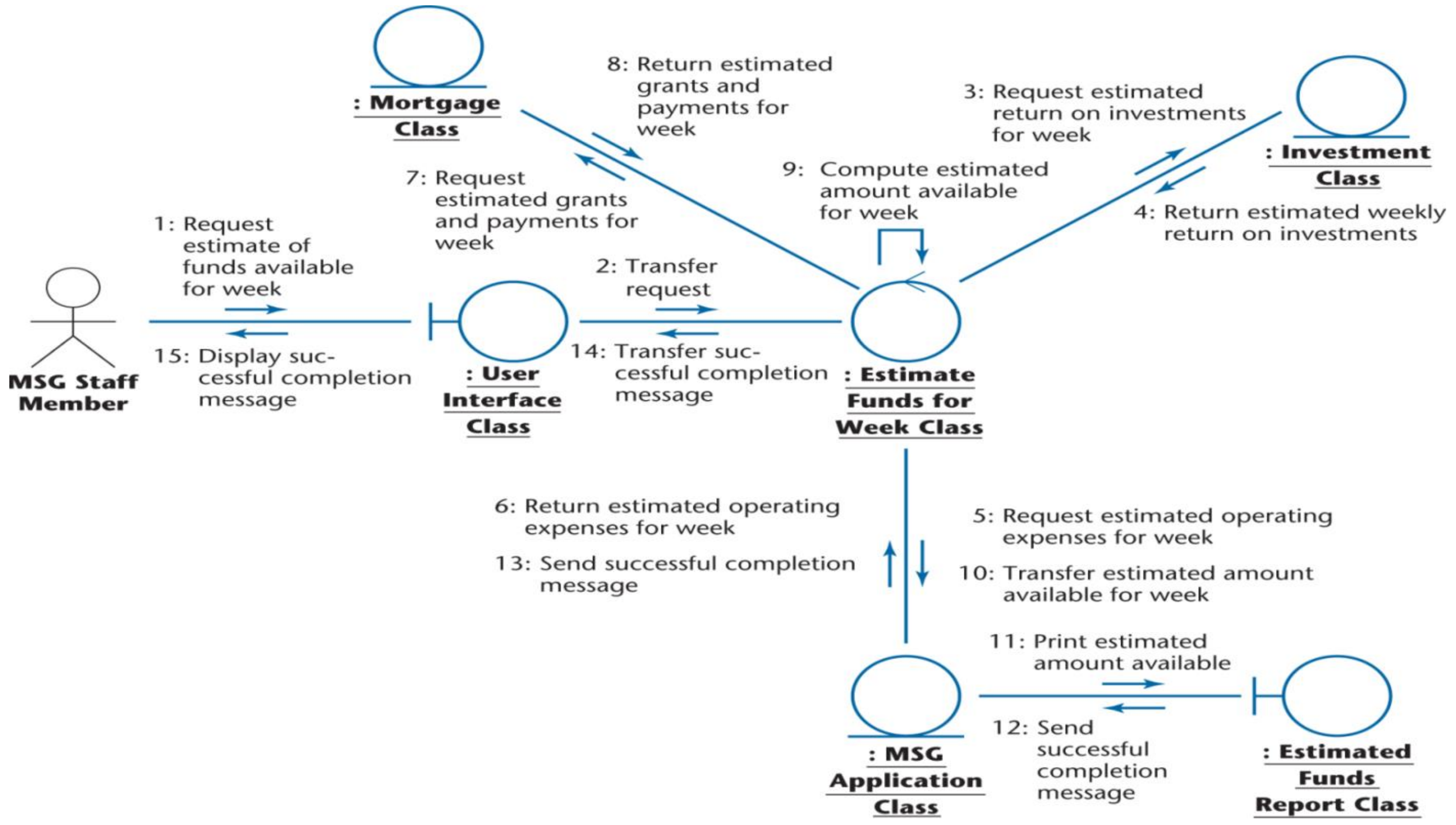

: Mortgage Class
8: Return estimated grants and payments for week
3: Request estimated return on investments for week
: Investment Class
7: Request estimated grants and payments for week
9: Compute estimated amount available for week
4: Return estimated weekly return on investments
1: Request estimate of funds available for week
2: Transfer request
MSG Staff Member
15: Display successful completion message
: User Interface Class
14: Transfer successful completion message
: Estimate Funds for Week Class
6: Return estimated operating expenses for week
13: Send successful completion message
5: Request estimated operating expenses for week
10: Transfer estimated amount available for week
11: Print estimated amount available
12: Send successful completion message
: MSG Application Class
: Estimated Funds Report Class

Collaboration Diagram Disadvantage

- No client will approve the specification document without understanding it
- A written description of the collaboration diagram is needed for the flow of events of the collaboration diagram.

Collaboration Diagram - Written Description

An MSG staff member requests an estimate of the funds available for mortgages for the week (1, 2). The information system estimates the return on investments for the week (3, 4), the operating expenses for the week (5, 6), and the grants and payments for the week (7, 8). Then it estimates (9), stores (10), and prints out (11–15) the funds available for the week.


: Mortgage Class
8: Return estimated grants and payments for week
3: Request estimated return on investments for week
: Investment Class
7: Request estimated grants and payments for week
9: Compute estimated amount available for week
4: Return estimated weekly return on investments
1: Request estimate of funds available for week
2: Transfer request
MSG Staff Member
15: Display successful completion message
: User Interface Class
14: Transfer successful completion message
: Estimate Funds for Week Class
6: Return estimated operating expenses for week
13: Send successful completion message
5: Request estimated operating expenses for week
10: Transfer estimated amount available for week
11: Print estimated amount available
: MSG Application Class
12: Send successful completion message
: Estimated Funds Report Class

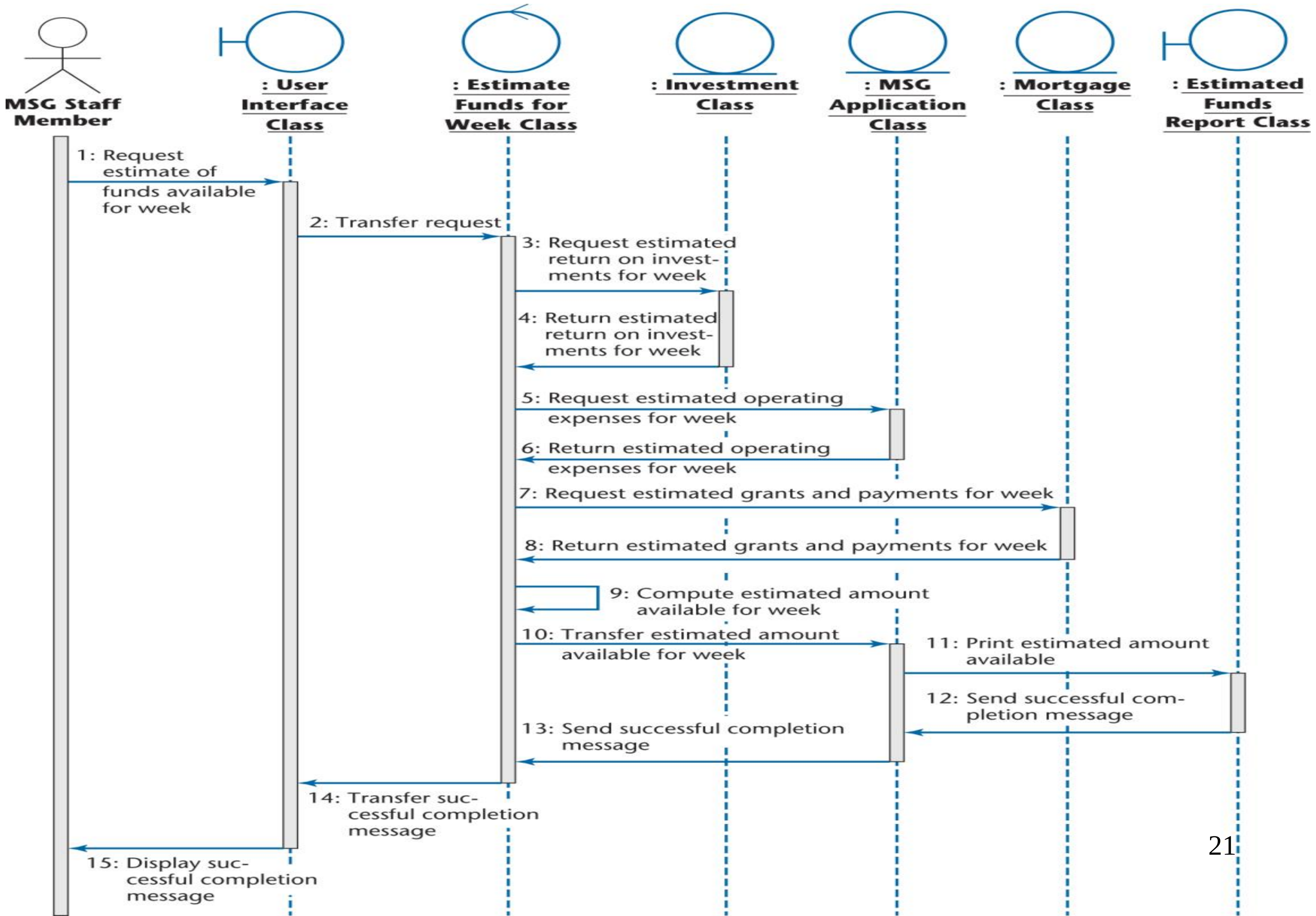
Sequence Diagrams

- Sequence Diagram is equivalent to the Collaboration Diagram
- It shows the flow of messages and their order in clearer way.

Collaboration or Sequence?

- Which is better to describe use cases, Sequence Diagram or Collaboration Diagram?
  - When transfer of information is our main goal, then a sequence diagram is better than collaboration diagram.
  - When the developers are focusing on the classes, then a collaboration diagram is more useful than sequence diagram.

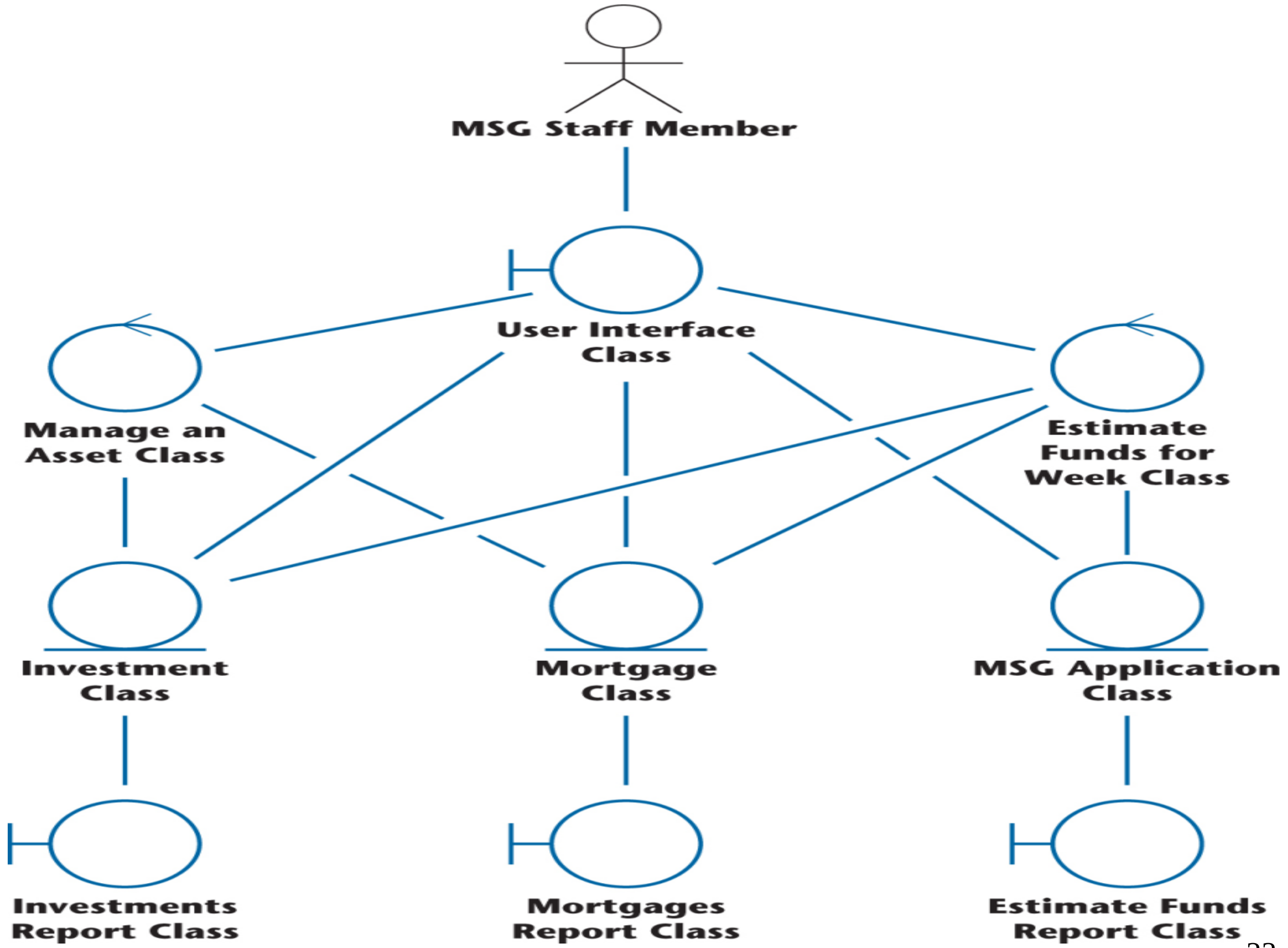
Sequence Diagram - MSG


MSG Staff Member
: User Interface Class
: Estimate Funds for Week Class
: Investment Class
: MSG Application Class
: Mortgage Class
: Estimated Funds Report Class
1: Request estimate of funds available for week
2: Transfer request
3: Request estimated return on investments for week
4: Return estimated return on investments for week
5: Request estimated operating expenses for week
6: Return estimated operating expenses for week
7: Request estimated grants and payments for week
8: Return estimated grants and payments for week
9: Compute estimated amount available for week
10: Transfer estimated amount available for week
11: Print estimated amount available
12: Send successful completion message
13: Send successful completion message
14: Transfer successful completion message
15: Display successful completion message

Class Diagram Realization

- In realizing various use cases:
  - Interrelationships between classes become clearer and reflected in class diagram (called *realization class diagram*)

Class Diagram Realization – MSG Example



CASE Tools for the Object-Oriented Analysis Workflow

- All modern tools support UML
  - Commercial tools
    - IBM Rational Rose
    - Together
  - Open-source tools
    - ArgoUML

Challenges of Object-Oriented Analysis Workflow

- Don’t cross the boundary into object-oriented design
- Don’t allocate methods to classes (reallocating methods to classes during design phase)

Metrics for Object-Oriented Analysis Workflow

- It is essential to measure the five fundamental metrics: **size, cost, duration, effort, and quality**
- Example: a measure of size of the object-oriented analysis
  - Number of pages of UML diagrams