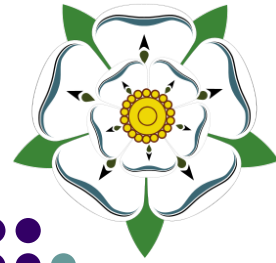
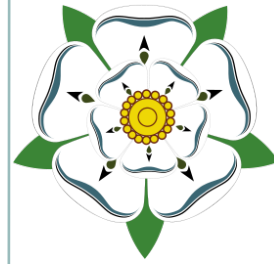


Use Cases - A Review



“There is no order in the world around us, we must adapt ourselves to the requirements of chaos instead.”

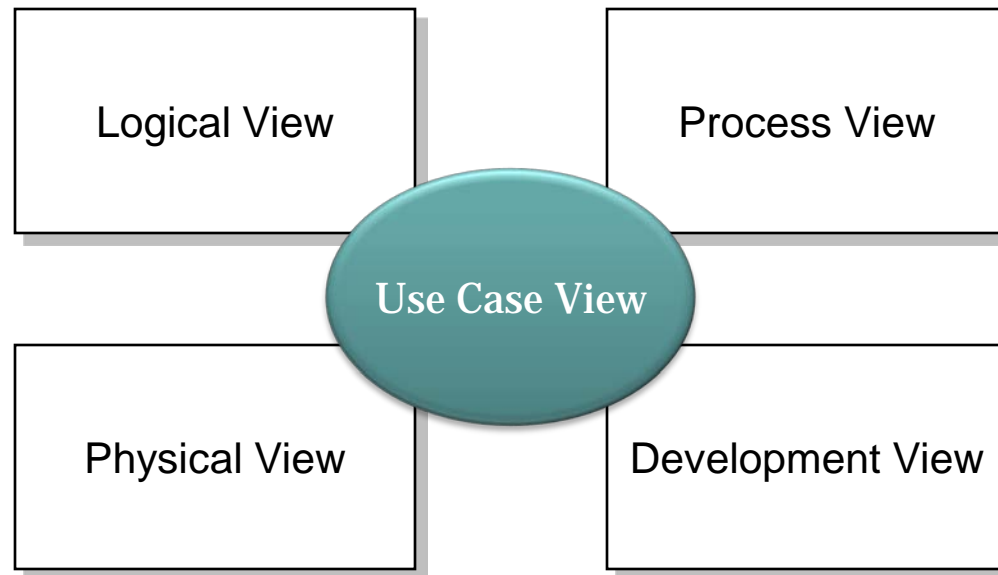
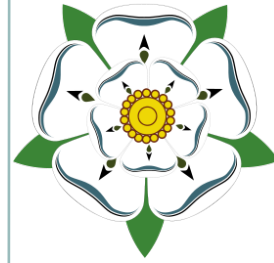
*Kurt Vonnegut
Breakfast of Champions*

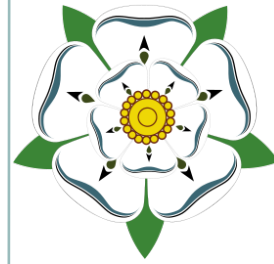


Module Objectives

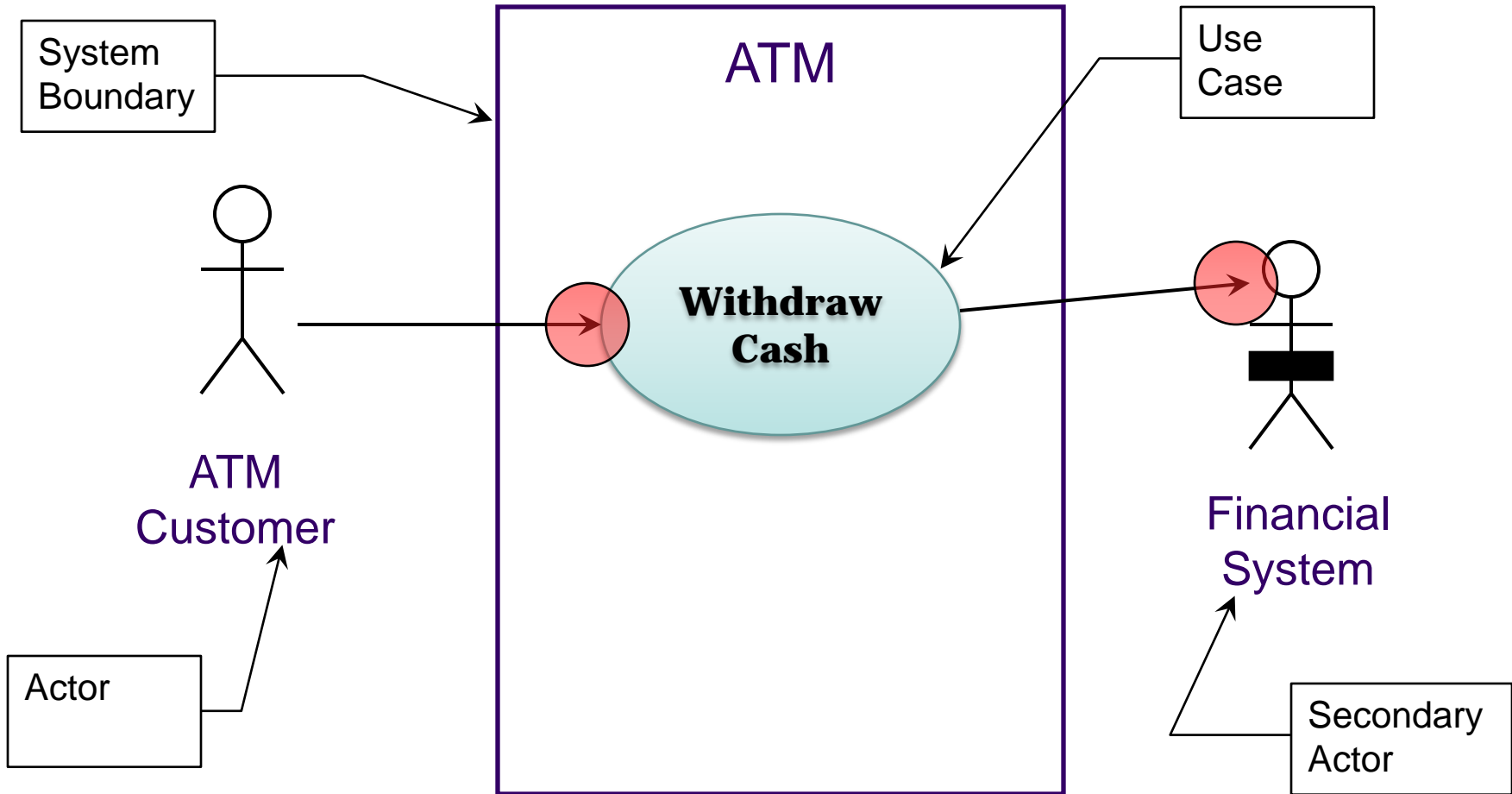
- To review the requirements process
- To recap the artifacts relating to Use Cases
- Introduce the new concept of personas
- To recall the supplemental requirements

The Use Case View

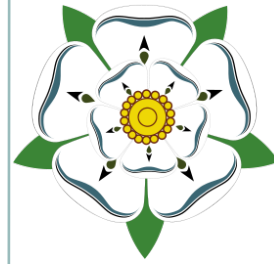




The Use Case Model



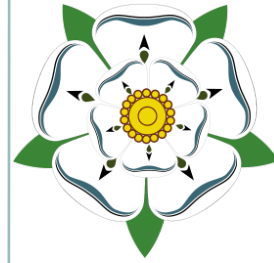
Actors Equate to Users



“An Actor is a person, system, event or anything *outside of the system* that interacts with the system.”

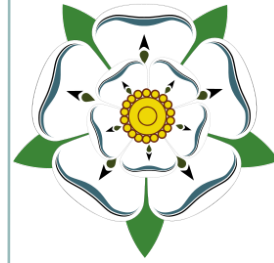


Actor Example



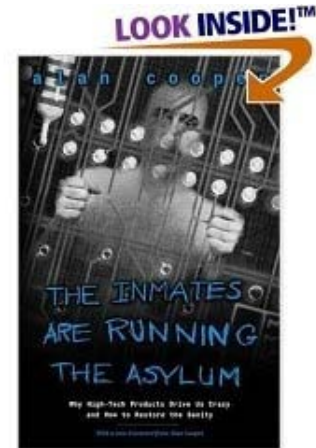
Name	ATM Customer
Actor Identifier	A1
Role	The ATM Customer is the primary user of the ATM. They will perform the routine tasks of withdrawing cash, depositing cash and checks, transferring money and making general account inquiries.
Business Knowledge	Medium
Technical Experience	Low to medium
Other Characteristics	The ATM customer will come from a wide background of technical and business knowledge. This means that the system interfaces should be simple to understand.
Population	1000 per ATM

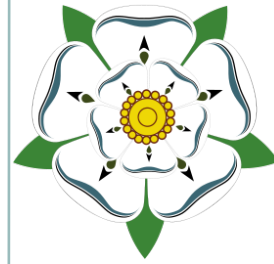
Personas



The Inmates Are Running the Asylum
1998, *Alan Cooper*

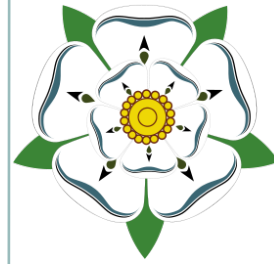
- Poor UI design is often the result of not fully understanding your user
- Personas are an easy way to develop a picture (model) of a set of typical users
- With these models in mind, the UI can be better tailored to their need





Developing a Persona

- In most cases, personas are synthesized from a series of ethnographic interviews with real people
- They are captured in 1-2 page descriptions that include:
 - Behavior patterns
 - Goals
 - Skills
 - Attitudes
 - Environment
 - Some fictional personal details to bring the persona to life



Example - Mary Jane

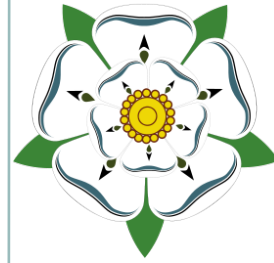


Mary Jane is in her late forties, married with three children, Bob, Kate and young William. She home-schools all of the children, and hence does not get much time during the day to attend to anything but the children and keeping the house clean. Her degree in English is a great help in developing lesson plans, but she does struggle sometimes with math and science subjects.

The only chance to run errands is in the evening when her husband, Ken, gets home, and even then only after she has served the evening meal. Mary and Ken are strong believers in family time, and insist that the entire family sit down to their evening meal together.

This means that the only time that Mary can get to the bank is in the evening, and must therefore use the ATM to get cash for the week. They live on a pretty tight budget, so she prefers to use cash for most shopping, rather than run up large credit card bills.

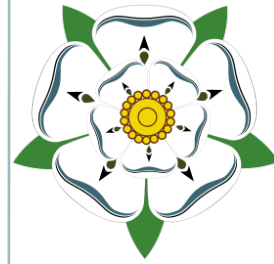
A Use Case Definition



“A Use Case is a description of a set of *sequences of actions, including variants*, that a system performs that *yield an observable result* of value to an Actor.”

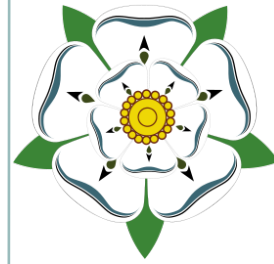
Grady Booch





The *Essential* Use Case

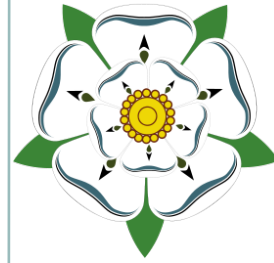
- Identify the use cases
 - Use a short, fully descriptive name
 - Note the primary actor
 - Write a narrative describing the task
- This level of detail is termed the “Essential Use Case”
- Sufficient to determine the project scope and Order of Magnitude estimate
- Breadth first approach
 - Wide and shallow



Essential Use Case Example

Name	Withdraw Cash
Identifier	UC 1
Primary Actor	ATM Customer
Secondary Actor	Financial System
Description	<p>This use case begins when an ATM customer chooses a type of account from which the cash is to be withdrawn (e.g. checking) from a list of possible accounts, and to choose a dollar amount from a list of possible amounts. The system sends the transaction to the financial system for verification. If the financial system approves the transaction, the machine dispenses the appropriate amount of cash and issues a receipt. The dispensing of cash is also recorded in the ATM's log.</p>
Source	Mary Ellen – Main Branch Teller Supervisor

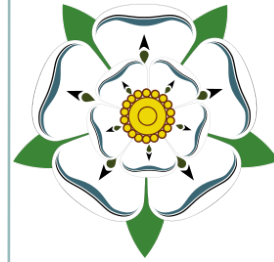
Exercise Introduction



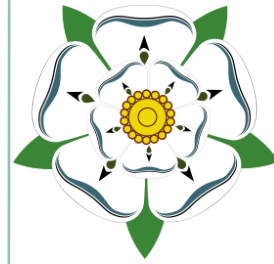
The exercises in this class will all be based on the game of Monopoly. We have been asked to develop a software simulation of the traditional board game.

You have each been given an exercise book that gives you an overview of the game plus an actual board game for each team.

Exercise 2

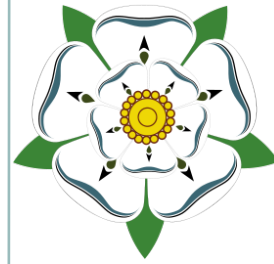


From the information provided, identify the actors in the game, and develop three essential use cases in which those actors will participate.



Essential Use Case Example

Name	Withdraw Cash
Identifier	UC 1
Primary Actor	ATM Customer
Secondary Actor	Financial System
Description	<p>This use case begins when an ATM customer chooses a type of account from which the cash is to be withdrawn (e.g. checking) from a list of possible accounts, and to choose a dollar amount from a list of possible amounts. The system sends the transaction to the financial system for verification. If the financial system approves the transaction, the machine dispenses the appropriate amount of cash and issues a receipt. The dispensing of cash is also recorded in the ATM's log.</p>
Source	Mary Ellen – Main Branch Teller Supervisor

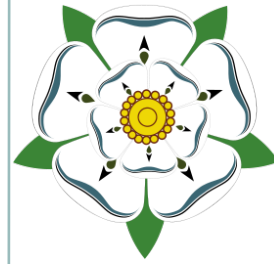


Exercise Review

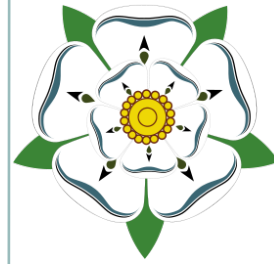
- Who are the Actors?
 - Player
 - Banker
- What are the use cases?
 - Move token
 - Buy property
 - Improve property
 - Get out of Jail
 - Pay rent
 - Pass Go



Scenarios

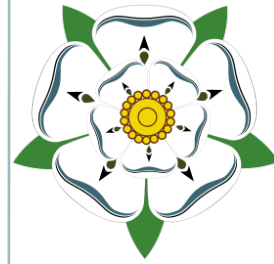


- Use cases comprise one or more scenarios
 - A scenario is a series of steps that describe a course of action through a use case
 - Steps vary between the actors and the system
- All use cases have a primary scenario
 - Commonly called the “Sunny Day Scenario” or “Happy Path”
 - The main flow of events
- A use case typically has secondary scenarios
 - Almost always one or more alternate flows
 - May describe exception conditions or options



Example - Withdraw Cash

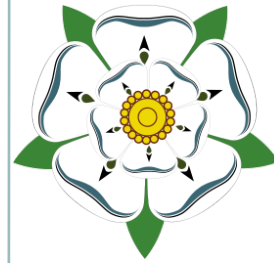
Name	Withdraw Cash
Identifier	UC4
Primary Actor	ATM Customer
Secondary Actor(s)	Customer Account System
Description	A withdrawal transaction asks the Customer to choose a type of account to withdraw from (e.g. checking) from a menu of possible accounts, and to choose a dollar amount. The system verifies that it has sufficient money on hand to satisfy the request before sending the transaction to the Customer Account and dispensing the cash.
Pre-Condition	The ATM Customer is validated and a new session started.
Main Flow	<ol style="list-style-type: none"> 1. This use case begins when the ATM Customer selects the Withdraw option. 2. The system displays the available account numbers and account names for the ATM Customer to select. 3. The ATM Customer selects the required account number. 4. The system requests the amount of cash requested. 5. The ATM Customer enters an amount, which must be in multiples of \$20. 6. The system communicates the account number and amount requested to the Customer Account System and requests authorization. 7. The Customer Account System verifies that the ATM Customer has sufficient funds in the account, debits the account and authorizes the system to dispense the cash. 8. This use case ends when the system dispenses the cash to the ATM Customer.
Post Condition	The ATM Customer account is debited by the amount of withdrawal.



Alternate Flows

- Describe what happens when business rules are violated or errors or exception conditions occur
- Only consider “normal business” exceptions
 - Wrong password typed
 - Invalid credit card
 - Part not available
- Alternate flows are part of the use case

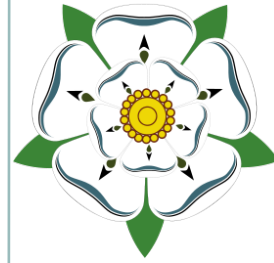
Prior Example - Withdraw Cash



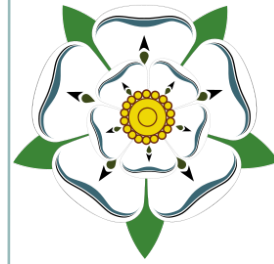
Name	Withdraw Cash
Identifier	UC4
Primary Actor	ATM Customer
Secondary Actor(s)	Customer Account System
Description	A withdrawal transaction asks the Customer to choose a type of account to withdraw from (e.g. checking) from a menu of possible accounts, and to choose a dollar amount. The system verifies that it has sufficient money on hand to satisfy the request before sending the transaction to the Customer Account and dispensing the cash.
Pre-Condition	The ATM Customer is validated and a new session started.
Main Flow	<ol style="list-style-type: none"> 1. This use case begins when the ATM Customer selects the Withdraw option. 2. The system displays the available account numbers and account names for the ATM Customer to select. The ATM Customer selects the required account number. The system requests the amount of cash requested. The ATM Customer enters an amount, which must be in multiples of \$20. The system communicates the account number and amount requested to the Customer Account System and requests authorization. The Customer Account System verifies that the ATM Customer has sufficient funds in the account, debits the account and authorizes the system to dispense the cash. 8. This use case ends when the system dispenses the cash to the ATM Customer.
Post Condition	The ATM Customer account is debited by the amount of withdrawal.

What if there isn't enough money?

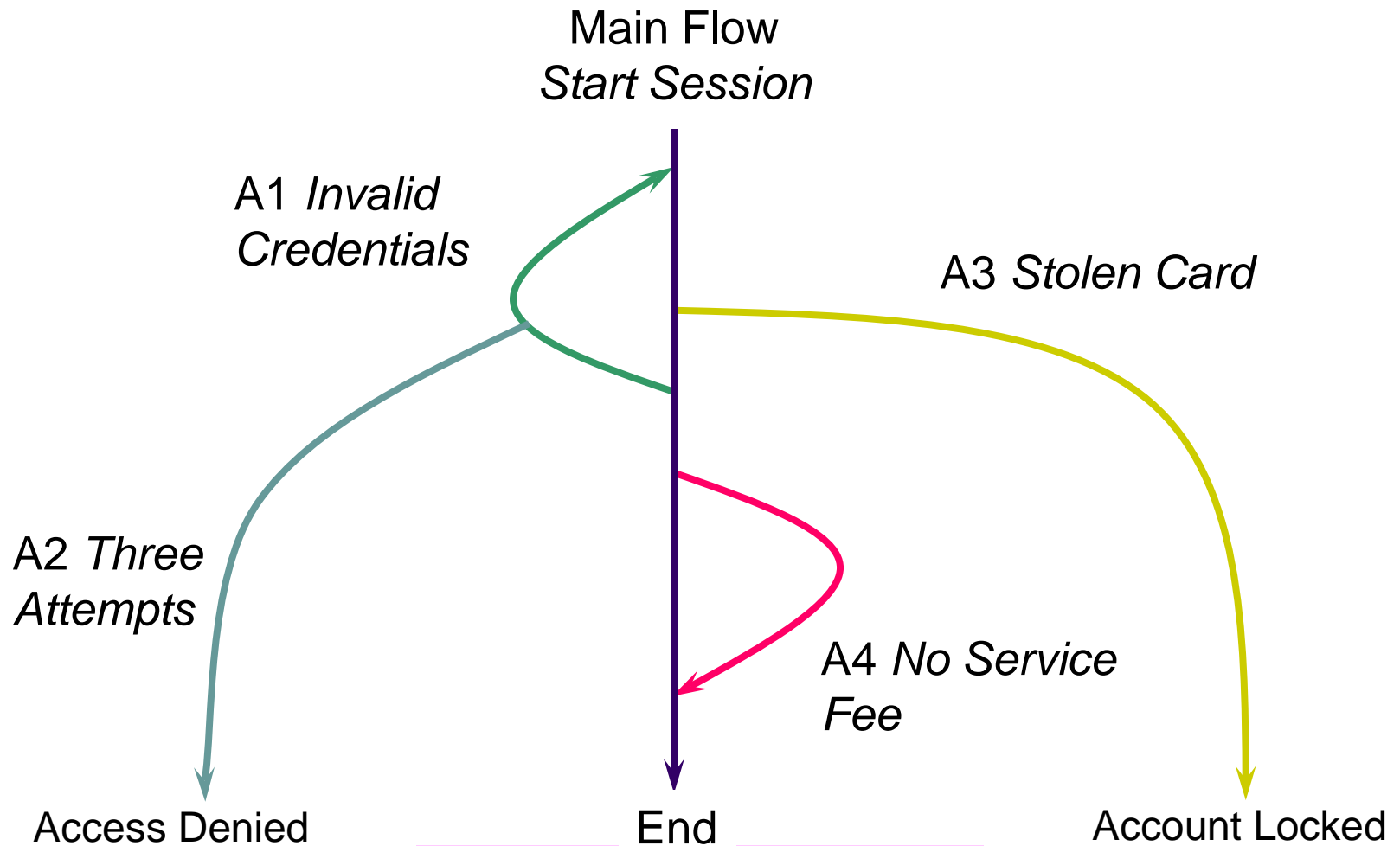
Example - Insufficient Funds

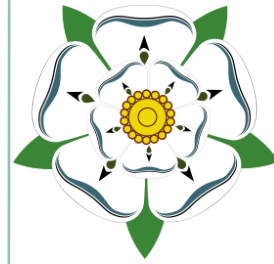


Alternate Flow	A1	Insufficient Funds
Originating Step		Main flow Step 7
Business Rule	BR1	Positive account balance must be maintained at all times.
Alternate Condition		A Customers account does not have sufficient funds to cover the amount requested.
Alternate Flow of Events		A1.1 The Customer Account System determines that there are insufficient funds in the customers' account to cover the withdrawal and denies the request from the system. A1.2 The system displays an error message to the ATM Customer and redisplay the options menu.
Alternate Post Condition		Options menu is redisplayed. Account balance is not changed.



Possible Scenarios

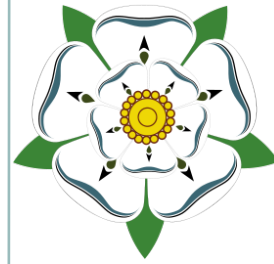




Use Case Tips

- Don't believe that use cases are static
 - Refactoring will occur as more knowledge is obtained
- Remember that the team goal is to produce working software
 - Not extensive documentation
 - Keep it lean – just good enough
- Take the breadth first approach
 - Understand a little about a lot

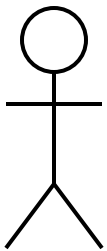
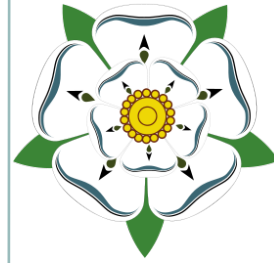




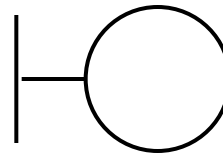
Robustness Analysis

- Described by Rosenberg & Scott in *Use Case Driven Object Modeling with UML*, 1999
- Also termed Use Case Analysis (IBM, 2003)
- Used to analyze the steps of a use case to validate the business process and business logic
- Confirms the usage requirements of the system being proposed

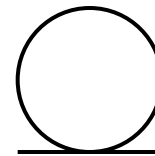
Visual Stereotypes



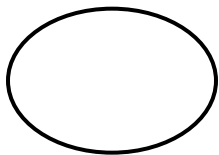
Actor



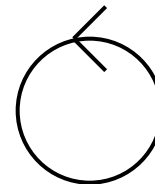
Boundary
(Interface)



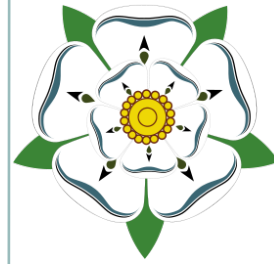
Entity
(Domain)



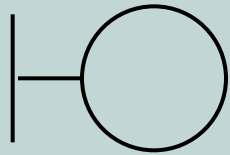
Use Case



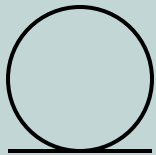
Control
(Process)



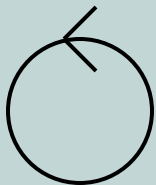
New Symbols



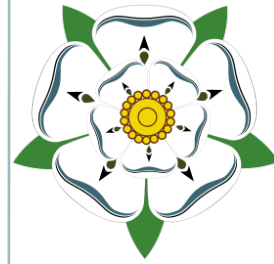
Boundary elements represent things such as screens, printers, system interfaces, card readers that allow actors to interface with the system.



Domain Entities map to things found in the conceptual model. We will discuss these in the next section.



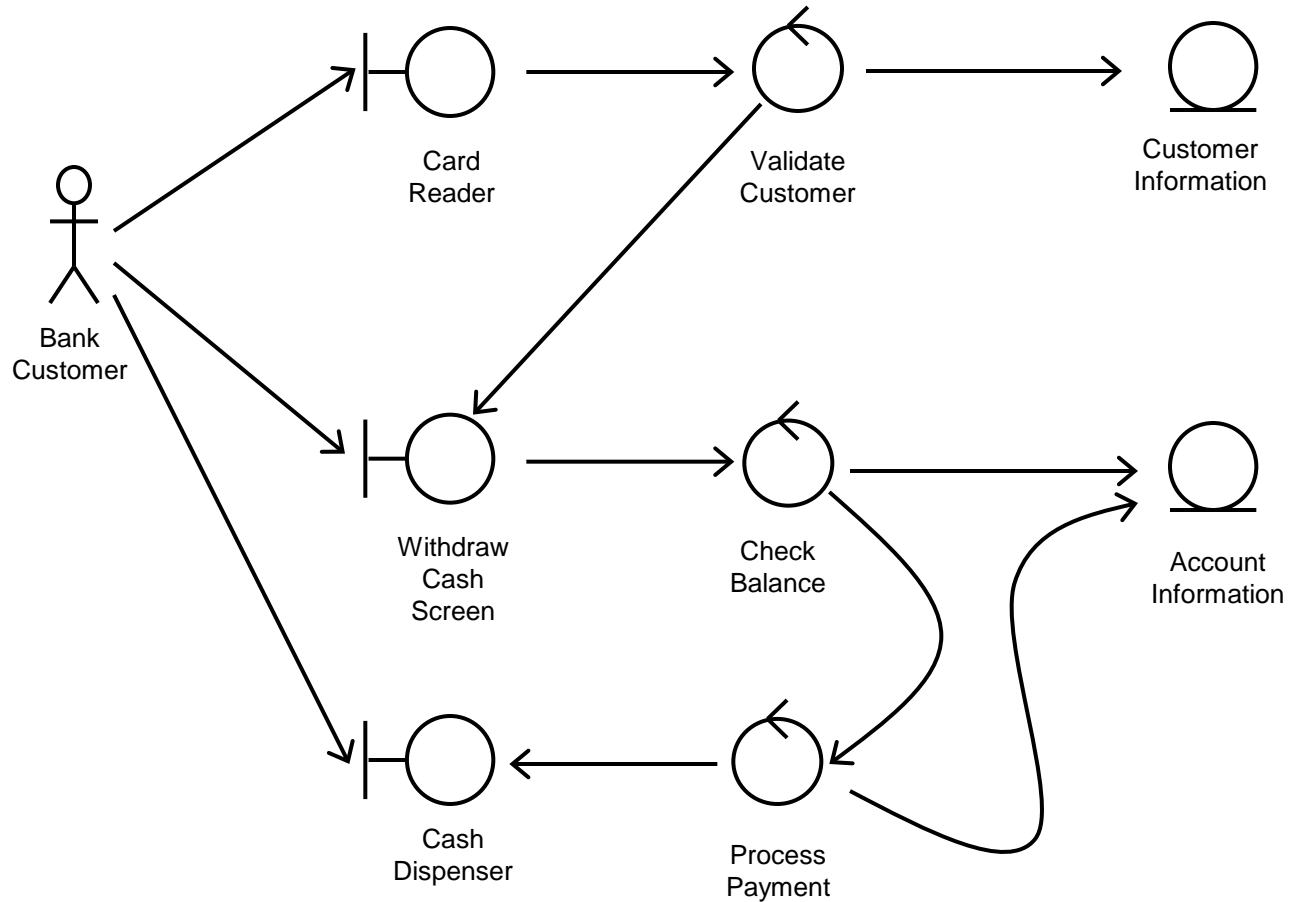
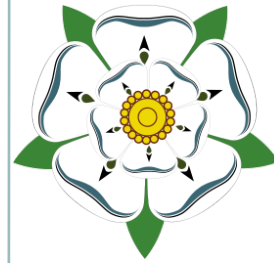
Control or Process elements implement the business logic and impose the business rules for the application.



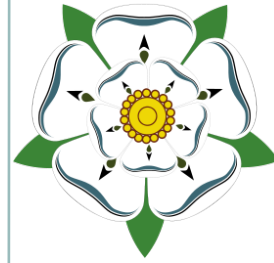
Example - Withdraw Cash

Name	Withdraw Cash
Identifier	UC4
Primary Actor	ATM Customer
Secondary Actor(s)	Customer Account System
Description	A withdrawal transaction asks the Customer to choose a type of account to withdraw from (e.g. checking) from a menu of possible accounts, and to choose a dollar amount. The system verifies that it has sufficient money on hand to satisfy the request before sending the transaction to the Customer Account and dispensing the cash.
Pre-Condition	The ATM Customer is validated and a new session started.
Main Flow	<ol style="list-style-type: none">1. This use case begins when the ATM Customer selects the Withdraw option.2. The system displays the available account numbers and account names for the ATM Customer to select.3. The ATM Customer selects the required account number.4. The system requests the amount of cash requested.5. The ATM Customer enters an amount, which must be in multiples of \$20.6. The system communicates the account number and amount requested to the Customer Account System and requests authorization.7. The Customer Account System verifies that the ATM Customer has sufficient funds in the account, debits the account and authorizes the system to dispense the cash.8. This use case ends when the system dispenses the cash to the ATM Customer.
Post Condition	The ATM Customer account is debited by the amount of withdrawal.

Withdraw Cash Robustness Model

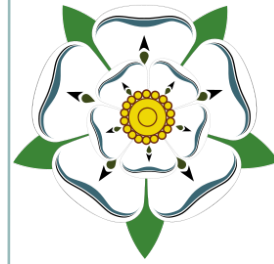


Just Good Enough



- Requirements need to be good enough
 - But not perfect
- Perfection is the enemy
- If requirements are unclear or insufficient, get more information
 - Ask The Stakeholder!
 - Keep talking
 - Use the whiteboard

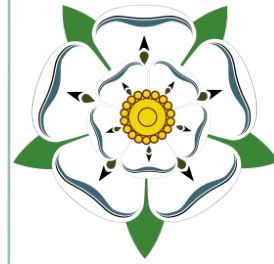




Requirement Categories

- Functional - Use Cases
 - The behavior of a system specified by the input/output conditions that are expected to result
- Business Rules
 - Policies established by the Business Unit that modify system functionality
- Non-functional
 - Attributes that describe how the system must perform
 - 'ilities': scalability, maintainability, usability

Summary



- Use Cases can be used to model the user functional requirements
- Personas can be useful in developing a better view of the users
- Robustness analysis helps in understanding the processing behind the use case
- Remember to KISS