

**E-Finance Case Study:
Requirements and Analysis -
Version 2.0**

Bert Lagaisse

Bart De Win

Wouter Joosen

Johan Van Oeyen

Report CW 438, March 2006



Katholieke Universiteit Leuven
Department of Computer Science

Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

E-Finance Case Study: Requirements and Analysis - Version 2.0

Bert Lagaisse

Bart De Win

Wouter Joosen

Johan Van Oeyen

Report CW438, March 2006

Department of Computer Science, K.U.Leuven

Abstract

This document defines a case study in the world of e-finance. The scope of this case study is retail banking services, in particular basic banking services (like current and savings accounts) and more advanced banking services for private investment. This document describes the first part in the full development of the case study: the requirements and analysis, with a particular focus on security. Compared to CW report 404, this (second) version of the document contains updates to the background context of the case (including applicable laws and regulations), the application of misuse cases to describe illegal behavior, and a fully detailed security policy.

E-Finance Case Study:
Requirements and Analysis - Version 2.0

Bert Lagaisse, Bart De Win, Wouter Joosen
DistriNet, K.U.Leuven

Johan Van Oeyen
Ubizen

March 9, 2006

Contents

1	Introduction	6
1.1	Financial products	7
1.1.1	Stocks	7
1.1.2	Shares	8
1.1.3	Options	8
1.2	The trading process	8
1.3	Custody accounts and depots	9
1.4	Overview of the rest of this document	9
2	Functional requirements	10
2.1	Actors and stakeholders	12
2.2	Use cases for basic banking services	13
2.2.1	Administration	13
2.2.2	Transactions	17
2.2.3	Account inspection	18
2.2.4	Bank system	20
2.3	Use cases for investment services	21
2.3.1	Custody account administration	22
2.3.2	Custody account inspection	22
2.3.3	Stock market catalogue inspection	23
2.3.4	Stock market transactions	24
2.3.5	Stock market feedback	27
2.4	Use Cases for transaction processing	29
2.4.1	Operational management	29
2.4.2	Transaction processing	29
2.5	Domain model	31
3	Security Requirements	35
3.1	Security and business context	35
3.1.1	Applicable regulations and legislations	35
3.2	Security analysis	43
3.2.1	Important actors and stakeholders	43
3.2.2	Misuse Cases	43
3.2.3	Business model and data sensitivity	56
3.3	Security requirements	60
3.3.1	Identification and Authentication	61
3.3.2	Access control	63
3.3.3	Privacy and confidentiality	65

<i>CONTENTS</i>	2
3.3.4 Data integrity	66
3.3.5 Non-repudiation and accountability	67
3.3.6 Audit	72
3.3.7 Recovery	73
4 Other non-functional requirements	74
4.1 Business Continuity	74
4.2 Quality attributes of the IT infrastructure	75
4.2.1 Availability	75
4.2.2 Performance	77
4.2.3 Usability	77
4.2.4 Security	77
5 Conclusion	78

Overview of Use Cases

	Page	Section
Use cases for basic banking services		
Administration	p13	2.2.1
- Create new customer	p13	2.2.1.1
- Show customer information	p14	2.2.1.2
- Edit a customer	p14	2.2.1.3
- Open a current account	p15	2.2.1.4
- Close a current account	p15	2.2.1.5
- Open a savings account	p15	2.2.1.6
- Close a savings account	p16	2.2.1.7
- Block accounts of a customer	p16	2.2.1.8
Transactions	p17	2.2.2
- Withdraw from a current account	p17	2.2.2.1
- Deposit on an account	p17	2.2.2.2
- Transfer between two accounts	p18	2.2.2.3
Account inspection	p18	2.2.3
- Show account information	p18	2.2.3.1
- Search a transaction	p19	2.2.3.2
Bank system	p20	2.2.4
- Transfer to another financial institution	p20	2.2.4.1
Use cases for investment services		
Custody account administration	p22	2.3.1
- Open custody account	p22	2.3.1.1
- Close custody account	N/A	N/A
Custody account inspection	p22	2.3.2
- Show custody accounts from customer	p22	2.3.2.1
- Show custody account information	p23	2.3.2.2
Stock market catalogue inspection	p23	2.3.3
- Show list of markets	p23	2.3.3.1
- Show list of stocks on market	p23	2.3.3.2
- Show stock information	p24	2.3.3.3
- Search stock information based on company name	N/A	N/A
Stock market transactions	p24	2.3.4
- Place a direct order for a stock	p24	2.3.4.1
- Place a limited order for a stock	p25	2.3.4.2
- Execute put option	p25	2.3.4.3
- Execute call option	p26	2.3.4.4
Stock market feedback	p27	2.3.5
- Confirm a buying order	p27	2.3.5.1
- Confirm a selling order	p27	2.3.5.2
- Deny an order	p27	2.3.5.3
- Send an invoice for a trading transaction	p28	2.3.5.4

Depot inspection	N/A	N/A
- Show pending orders	N/A	N/A
- Show security depot: list of open positions	N/A	N/A
- Show details of an open position	N/A	N/A
- Show realization depot : list of closed positions	N/A	N/A
- Show details of a closed position	N/A	N/A
Depot management	N/A	N/A
- Open a position on a stock	N/A	N/A
- Close a position on a stock	N/A	N/A

Use cases for transaction processing

Operational management	p29	2.4.1
-Start handling transaction queue	p29	2.4.1.1
Transaction processing	p29	2.4.2
- Process a deposit	p29	2.4.2.1
- Process a withdrawal	p30	2.4.2.2
- Process a transfer	p30	2.4.2.3
- Process a charge	p31	2.4.2.4

Overview of MisUse Cases

Misuse case	Page	Section
Customer and account management	p44	3.2.2.1
- Create false customer	p44	3.2.2.1
- Show customer information	p45	3.2.2.1
- Edit customer information	p46	3.2.2.1
- Create a false account	p47	3.2.2.1
- Close an account with a negative balance	p48	3.2.2.1
Transactions	p49	3.2.2.2
- Withdraw from a current account below credit limit	p49	3.2.2.2
- Deposit on a current account without delivering cash	p50	3.2.2.2
- Create an illegal transaction	p51	3.2.2.2
Banking system	p52	3.2.2.3
- Transfer to another financial institution	p52	3.2.2.3
Investment services	p53	3.2.2.4
- Custody account management	p53	3.2.2.4
- Catalogue inspection and false information	p53	3.2.2.4
- Misusing the custody account	p55	3.2.2.4
- Creation of illegal stock trade transactions	p55	3.2.2.4

Chapter 1

Introduction

This document defines a case study in the world of electronic finance. This case study is one of the case studies that is being defined in the context of the SoBeNeT project.¹ The scope of the case study is retail banking services. We particularly target on a subset of retail banking services: basic banking services (like current accounts and savings accounts) and more advanced retail banking services concerning private investments; other services (loans and insurance) are considered out of scope. Furthermore, the case study focuses on the subsystem of the administrative software of the bank for the retail banking services. Subsequently, the internals of other systems involved (e.g., the stock market) are not considered to be part of the case study; only communication between these other systems and the bank's system will be included.

The typical users of the system for retail banking services are bank clerks and ordinary customers. Bank clerks use the system from workstations in different branch offices to handle requests from a customer at the branch office. Customers can use the system directly through a self-banking terminal, through home banking or indirectly as a secondary (supporting) actor at the bank's branch office.

We assume the reader is familiar with basic banking services. In summary, each customer of a bank owns a current account to execute transactions like cash deposits, cash withdrawals and transfers. Next to a current account a customer can possess a savings account, which is associated with the customer's current account. Savings accounts offer a higher interest rate but have some restrictions on the possible transactions with it. For instance, it is not possible to do a cash withdrawal on a savings account and transfers are only possible to the associated current account.

In the rest of this chapter we introduce less well-known concepts including the relevant financial products, the stock market, the trading process, custody accounts and depots. We conclude with an overview of the rest of this document.

¹The SoBeNeT project is an IWT-SBO project on the development of secure application software. Refer to <http://sobenet.cs.kuleuven.ac.be/> for more information about the project.

1.1 Financial products

One can invest in a lot of financial products:

- The basic investment products like savings accounts.
- Products with a higher risk like stocks.

In this section we focus on *stocks*. Two kinds of stocks exist: securities and derivatives. In this case study one kind of security and one kind of derivative will be considered:

- *Shares* (as a kind of security)
- *Options* (as a kind of derivative)

First we will explain the main concepts of stocks, shares and options. For the sake of this case study, irrelevant details will be simplified or even omitted.

1.1.1 Stocks

A stock is a financial instrument in which one can invest. It has a certain trading price that fluctuates based on the economical laws of ask and bid. All stocks have some common properties:

Stock Code (ISIN): A standardized identification number to uniquely identify a stock across all markets.

Last Sale: The last trading price.

Currency: The currency in which the stock is notated.

Today's High: The intra-day highest trading price.

Today's Low: The intra-day lowest trading price.

Historical notations: The historical notations of a stock are a list of all important properties of a stock for each trading day at the market. A notation for one trading day contains: the date, opening trading price, highest trading price, lowest trading price and closing trading price

The market. Every stock is notated at a certain market. At his market stocks can be traded in a controlled way. The laws of ask and bid determine the trading price of the stocks. Normally a stock is notated at just one market and can only be traded on this market. Markets are determined by a unique market code. Often markets specialize in one kind of stock ². All stocks notated at a market have the same currency. This currency is determined by the market. Each market has a stock catalogue in which is publishes the stocks notated at that market. It contains the different values associated with a stock and the historical notations of the stock.

²like shares or options

1.1.2 Shares

A share is a participation in the capital of a corporation. Next to the properties mentioned in the generic description of a stock, shares also have historical notations of the traded volume on a certain day.

A share also gives the right to take part in the profits of the corporation. Every year the corporation can decide to pay a dividend to the shareholders. This dividend can be money, shares, options or other values.

1.1.3 Options

An option is the right to buy (call option) or sell (put option) a specific security at a price which is determined in advance by the financial institution that issues the option. It is valid for a limited period. Typical properties of an option are:

The kind of option: call or put option

The underlying security: the security for which the option gives you the right to sell or to buy. For example, this security can be a share of IBM.

Period: The period during which the option can be executed.

Execution price: The price to buy or sell the underlying security. This price is granted by the financial institution that issues the option.

Trading price: Next to the execution price, an option also has a trading price. As mention in the properties of a stock, this is the price against which the option is traded.

Quantisation: The amount of securities on which the option applies.

1.2 The trading process

When one wants to buy or sell an amount of securities, an order is sent to the market. All orders are notated in the order book at the market. This order book has two sides: the selling orders and the buying orders. Orders are also divided in direct orders and limited orders. A direct order is an order to buy or sell a stock at the market's current best displayed price. A limit order is an order to buy or sell a stock at a customer specified price. Based on these limit orders the best ask price and best bid price for a stock at a certain moment are known. The best bid price is the highest buying price of all limited buying orders. The best ask price is the lowest selling price of all limited selling orders.

The market will try to process orders by searching matches between the buying and selling sides of the order book. The amount of stocks involved in the orders is not necessarily the same in matching orders. For the amount of stocks in an order that could not be matched, a new order is created and entered in the order book.

When a match is found, the trading transaction will be published. The settlement of the transaction (exchange of the values and payment) is handled by a settlement organisation designated by the stock market.

1.3 Custody accounts and depots

The investment services of a bank are offered to the customer by means of a *custody account*. This account is used to execute the financial transactions involved in trading stocks. In case of buying stocks, the amount to pay is withdrawn from the account, in case of selling stocks it is deposited on the account.

A customer can choose the currency of each of his custody accounts, depending on the market on which he is frequently trading. This saves the customer currency exchange costs. So normally, a custody account in dollar will be used to trade on the stock markets in dollar.

To open a custody account, a customer already needs a current account, with which the custody account will be associated. The possible money transactions with a custody account are limited: only money transfers involved in the settlement of a trading transaction on the stock market, and transfers with the associated current account are allowed.

When a customer owns a custody account, he can request the bank to place an order on a stock market. The custody account contains an overview of all orders that are pending on the market. When the stock market gives feedback about an order, the pending order will be marked as confirmed (if the order could be processed by the market) or rejected. In case the order is confirmed, the market will also provide the details about the associated trading transaction.

A custody account contains a *security depot*. Within a security depot a customer can open a position on a certain stock. Opening a position on a stock means the customer wants to follow up on this stock and wants to invest in it. All trading transactions of the customer on this stock are stored in this security depot, in the open position on the stock. Each open position shows the amount of stocks in the position, profits or losses already made and important properties of the stock like its current trading price and the historical notations.

When a customer is no longer interested in investing in a certain stock, he can close the position (at this time the amount of stocks in the position has to be nihil). Closed positions are stored in the *realization depot*. The realization depot contains for each closed position the definitive profit or loss and the trading transactions on the stock.

1.4 Overview of the rest of this document

In the second chapter we will elaborate on the functional requirements of the bank's private investment software. The third chapter will further elaborate on the security requirements. The fourth chapter will discuss other non-functional requirements (or software qualities).

Chapter 2

Functional requirements

This chapter presents the functional requirements of the e-finance system, which includes basic banking services, investment services and transaction processing. The functional requirements are expressed in the form of actors, stakeholders and use cases. The actors will be described in more detail in Section 2.1. The main part of the chapter contains use case descriptions of the system: use cases for basic banking services in Section 2.2, use cases for investment services in 2.3 and use cases for transaction processing in Section 2.4. At the end, a domain model is included to present a detailed diagrammatic overview of the problem domain.

A graphical overview of the functional requirements is depicted in Figure 2. It contains the different actors (represented as person icons), systems (rectangles) and use cases (ovals, some of which represent use case categories rather than single use cases).

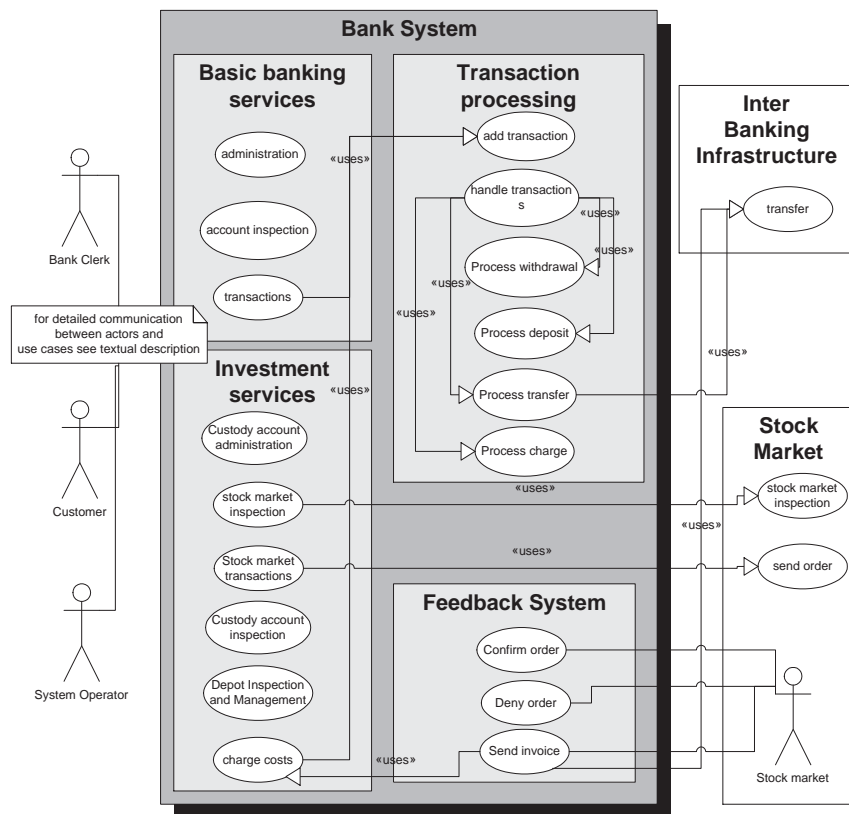


Figure 2.1: Use case model

2.1 Actors and stakeholders

The actors of the banking system are the entities that use the functionality of the system. This includes people, organizations and software systems. We distinguish the following primary actors in the system: Bank clerks and customers.

Bank clerks use the system from workstations in different branch offices, to handle requests from a customer at the branch office.

Customers use the system directly through a self-banking terminal, through home banking or indirectly as a secondary (supporting) actor at the bank's branch office.

Other important actors of the system are the following members of the bank's personnel: the office manager and the operational manager.

Office manager He is the manager of a branch office and is the supervisor of the bank office personnel. He is allowed to block accounts of customers.

Operational manager The operational manager is responsible for the operational management of the banking system.

The following systems are also actors within the banking system: the transaction handler, the stock market system and the settlement organization.

Transaction handler The transaction handler is a system that processes the transaction queue. All transactions (withdrawals, deposits, transfers and cost charges) created by the different users are stored in a transaction queue. Afterwards (mostly at night) this queue is processed in batch by the transaction handler.

Stock market The stock markets are external systems for stock trading known by the bank.

Settlement organization The settlement organization handles the payment of the traded financial products and delivers the financial products to the buyer.

Finally, because money must be transferred between different banks and between the stock market and the banks, we will assume that every financial institution has an account at an **inter banking infrastructure**. These organizations are trusted to enable transactions between financial institutions.

2.2 Use cases for basic banking services

This section describes the uses case of the subsystem for the basic banking services. We describe them using the standard usecases.org format¹. The following cases will be discussed:

Administration

- Create new customer
- Show customer information
- Edit a customer
- Open a current account
- Close a current account
- Open a savings account
- Close a savings account
- Block a customer account

Transactions

- Withdraw from a current account
- Deposit on an account
- Transfer between two accounts

Account inspection

- Show account information.
- Search a transaction.

Bank system

- Transfer between the bank's own account and another financial institution.

2.2.1 Administration

2.2.1.1 Use Case: Create new customer

Primary actor: Bank clerk

Supporting actor: Customer

Stakeholders: Customer, Bank, Bank clerk

¹For more information, we refer to <http://usecases.org/>.

Basic flow:

1. The actor selects to create a new customer.
 2. The system asks for the required information about the customer (first name, last name, birth date, address, social security number).
 3. The actor gives the required information.
 4. The system verifies the given information.
 5. The system creates a new customer when verification succeeded.
 6. The system notices the actor (success or failure).
-

2.2.1.2 Use Case: Show customer information**Primary actor:** Bank clerk**Basic flow:**

1. The actor selects to show a customer's information.
2. The system asks for the social security number.
3. The actor gives the social security number.
4. The system shows the customer information

Alternative scenarios:**1-4a. The customer has to be looked up by name**

1. The actor chooses to search for the customer.
 2. The system asks for the name of the customer.
 3. The actor gives the name of the customer.
 4. The system shows a list of matching customers.
 5. The actor selects a customer from the list.
 6. The system shows the customer information.
-

2.2.1.3 Use Case: Edit customer information**Primary actor:** Bank clerk**Supporting actor:** Customer**Stakeholders:** Customer, Bank, Bank clerk**Preconditions:** The actor has selected to show a customer's information.**Basic flow:**

1. The actor edits the customer's information.
 2. The system verifies the given information.
 3. The system creates a new customer when verification succeeded.
 4. The system notices the actor (success or failure).
-

2.2.1.4 Use Case: Open a current account**Primary actor:** Bank clerk**Supporting actor:** Customer**Stakeholders:** Customer, Bank, Bank clerk**Basic flow:**

1. The actor selects to open a new current account.
 2. The system asks for the owners of the current account.
 3. The actor adds customers as owners.
 4. The system verifies the given information.
 5. The system generates an account number.
 6. The system asks to sign the request for a new current account
 7. Each owner and the bank sign the request
 8. The system creates a new current account.
 9. The system notices the actor (success or failure).
-

2.2.1.5 Use Case: Close a current account**Primary actor:** Bank clerk**Supporting actor:** Customer**Stakeholders:** Customer, Bank, Bank clerk**Preconditions:** The actor has selected a current account.**Basic flow:**

1. The actor selects to close a current account.
 2. The system verifies on the existence associated savings accounts.
 3. The system verifies if the balance of the current account is zero.
 4. The system asks to sign the request for closure.
 5. Each owner and the bank sign the request.
 6. The system closes the current account.
 7. The system notices the actor (success or failure).
-

2.2.1.6 Use Case: Open a savings account**Primary actor:** Bank clerk**Supporting actor:** Customer

Stakeholders: Customer, Bank, Bank clerk

Basic flow:

1. The actor selects to open a new savings account.
 2. The system asks for the associated account of the savings account.
 3. The actor selects the associated account.
 4. The system verifies the given information.
 5. The system generates an account number.
 6. The system asks to sign the request for a new account
 7. Each owner and the bank sign the request
 8. The system creates a new savings account.
 9. The system notices the actor (success or failure).
-

2.2.1.7 Use Case: Close a savings account

Primary actor: Bank clerk

Supporting actor: Customer

Stakeholders: Customer, Bank, Bank clerk

Preconditions: The actor has selected a savings account.

Basic flow:

1. The actor selects to close a savings account.
 2. The system verifies if the balance of the account is zero.
 3. The system asks to sign the request for closure.
 4. Each owner and the bank sign the request.
 5. The system closes the savings account.
 6. The system notices the actor (success or failure).
-

2.2.1.8 Use Case: Block a customer account

Primary actor: Bank clerk

Supporting actor: Office manager

Stakeholders: Customer, Bank, Bank clerk, Office manager

Preconditions: The actor has selected a customer.

Basic flow:

1. The actor selects to block the accounts of the selected customer.
 2. The system looks up the accounts of the customer.
 3. The system asks to sign the request to block the accounts.
 4. The office manager signs the request.
 5. The system blocks the accounts.
-

2.2.2 Transactions**2.2.2.1 Use Case: Withdraw from a current account**

Primary actor: Bank clerk or customer

Supporting actor: Customer (in case Bank clerk is primary actor)

Stakeholders: Customer, Bank, Bank clerk (in case Bank clerk is primary actor)

Preconditions: The actor has selected a current account.

Basic flow:

1. The actor starts a withdrawal.
2. The system creates a new withdrawal.
3. The system asks for the amount to withdraw.
4. The actor enters the amount to withdraw.
5. The system verifies the withdrawal.
6. The system asks to sign the withdrawal.
7. The customer and the bank sign the withdrawal.
8. The system sends the withdrawal to the transaction handler.
9. The cash money is provided to the customer.

Extensions:

- 9a The bank clerk provides the money to the customer.
 - 9b The self banking terminal provides the money to the customer.
-

2.2.2.2 Use Case: Deposit on an account

Primary actor: Bank clerk.

Supporting actor: Customer.

Stakeholders: Customer, Bank, Bank clerk.

Preconditions: The actor has selected an account.

Basic flow:

1. The actor starts a deposit.
 2. The system creates a new deposit.
 3. The system asks for the amount to deposit.
 4. The actor enters the amount to deposit.
 5. The system asks to sign the deposit.
 6. The customer and the bank sign the deposit.
 7. The system sends the deposit to the transaction handler.
-

2.2.2.3 Use Case: Transfer between two accounts

Primary actor: Bank clerk or customer

Supporting actor: Customer (in case bank clerk is primary actor)

Stakeholders: Customer, Bank, Bank clerk (in case bank clerk is primary actor)

Basic flow:

1. The actor selects an account.
 2. The actor starts a transfer.
 3. The system creates a new transfer.
 4. The system asks for the destination account
 5. The actor enters the destination account
 6. The system asks for the amount to transfer.
 7. The actor enters the amount to deposit.
 8. The system verifies the transfer.
 9. The system asks to sign the transfer.
 10. The customer and the bank sign the transfer.
 11. The system sends the transfer to the transaction handler.
-

2.2.3 Account inspection**2.2.3.1 Use Case: Show account information**

Primary actor: Bank clerk or customer

Supporting actor: Customer (in case bank clerk is primary actor)

Stakeholders: Customer, Bank, Bank clerk (in case bank clerk is primary actor)

Basic flow:

1. The actor selects to show an account's information.
2. The system asks for the account number.
3. The actor gives the account number.
4. The system shows the account information containing the account number, owners of the account, balance of the account and an overview of pending transactions and completed transactions.

Alternative scenarios:**1a. The account has to be looked up by the customer's id**

1. The actor chooses to search for the account.
2. The system asks to select a customer.
3. The actor selects a customer.
4. The system shows a list of accounts of the customer.
5. The actor selects an account from the list.
6. The system shows the account information.

Alternative scenarios:**1a. The customer is operating on the system using home banking**

1. The actor selects to show an account's information.
2. The system shows a list of accounts of the customer.
3. The actor selects an account from the list.
4. The system shows the account information.

2.2.3.2 Use Case: Search a transaction**Primary actor:** Bank clerk or customer**Supporting actor:** Customer (in case bank clerk is primary actor)**Stakeholders:** Customer, Bank, Bank clerk (in case bank clerk is primary actor)**Preconditions:** The actor has selected an account.**Basic flow:**

1. The actor selects to search a transaction
 2. The system asks for searching variables like the type of transaction, the range of the amount of the transaction and in case of a transfer the destination account.
 3. The actor gives some of the searching variables
 4. The system shows an overview of pending transactions and completed transactions matching the search variables.
-

2.2.4 Bank system

2.2.4.1 Use Case: Transfer between the bank's own account and another financial institution

Primary actor: The bank system.

Stakeholder: Inter banking infrastructure providing this service.

Basic flow:

1. The bank system requests a transfer at the inter banking infrastructure providing the destination account, the amount to transfer and a message specifying other information
 2. The bank and the inter banking infrastructure sign the transfer.
-

2.3 Use cases for investment services

The use cases for investment services relate to custody accounts, the stock market and depots. The following use cases are included:

Custody account administration

- Open custody account.
- Close custody account.

Custody account inspection

- Show custody accounts from customer
- Show custody account information

Custody account transactions These transactions are identical to the transfer between normal accounts and will not be worked out in detail.

- Transfer from account to custody account
- Transfer from custody account to account

Stock market catalogue inspection

- Show list of markets
- Show list of stocks on market
- Show stock information
- Search stock information based on company name

Stock market transactions

- Place a direct order for a stock
- Place a limited order for a stock
- Execute put option
- Execute call option

Stock market feedback

- Confirm a buying order.
- Confirm a selling order.
- Deny an order.
- Send an invoice for a trading transaction.

Depot management

- Open a position on a stock
- Close a position on a stock

Depot inspection

- Show pending orders
- Show security depot: list of open positions
- Show details of an open position
- Show realization depot : list of closed positions
- Show details of a closed position

2.3.1 Custody account administration**2.3.1.1 Use Case: Open a custody account**

Primary actor: Bank clerk

Supporting actor: Customer

Stakeholders: Customer, Bank, Bank clerk

Basic flow:

1. The actor selects to open a new custody account.
2. The system asks for the associated account of the custody account.
3. The actor selects the associated account.
4. The system asks for the currency of the custody account
5. the actor selects the currency
6. The system verifies the given information: all owners should be adults.
7. The system asks to sign the request for a new custody account.
8. Each owner of the associated account and the bank sign the request.
9. The system generates a custody account number.
10. The system creates a new custody account.
11. The system notices the actor (success or failure).

The use case for closing a custody account is not further specified.

2.3.2 Custody account inspection**2.3.2.1 Use Case: Show custody accounts from customer**

Primary actor: Bank clerk or customer

Preconditions: A customer is selected.

Basic flow:

1. The actor selects to get an overview of the custody accounts of the selected customer.
2. The system shows a list of custody accounts of which the customer is owner.

Alternate scenarios**1a The customer is the actor**

1. The actor chooses to get an overview of his custody accounts
-

2.3.2.2 Use Case: Show custody account information

Primary actor: Bank clerk or customer

Preconditions: The system is showing a list of custody accounts of the involved customer.

Basic flow:

1. The actor selects to get an overview of a custody account.
 2. The system shows an overview of the custody account: balance, currency, a list of pending orders, a financial balance of the security depot and a financial balance of the realization depot.
-

2.3.3 Stock market catalogue inspection**2.3.3.1 Use Case: Show list of markets**

Primary actor: Bank clerk or customer

Basic flow:

1. The actor selects to get an overview of the markets
 2. The bank system asks the stock market catalogue system for a list of the markets
 3. The catalogue system gives the bank system the requested list.
 4. The bank system shows an overview of the markets containing market code, market name and currency.
-

2.3.3.2 Use Case: Show list of stocks on market

Primary actor: Bank clerk or customer

Basic flow:

1. The actor selects to get an overview of the stocks on a certain market.
 2. The bank system asks for the market code.
 3. The actor gives the market code.
 4. The bank system asks the stock market catalogue system for a list of stocks on the given market.
 5. The catalogue system gives the bank system the requested list.
 6. The bank system shows an overview of the stocks containing stock code (ISIN), company name, last sale, currency, absolute change, relative change and share volume.
-

2.3.3.3 Use Case: Show stock information

Primary actor: Bank clerk or customer

Basic flow:

1. The actor selects to get an overview of a stock.
 2. The bank system asks for the ISIN code of the stock.
 3. The actor gives the ISIN code.
 4. The bank system asks the stock market catalogue system for detailed data of the stock.
 5. The catalogue system gives the bank system the requested list.
 6. The bank system shows an overview of the stock containing stock code (ISIN), company name, last sale, currency, absolute change, relative change, share volume, best bid, best ask , today's high / low and the historical notations containing: the date, opening rate, highest rate, lowest rate, closing rate and share volume.
-

2.3.4 Stock market transactions**2.3.4.1 Use Case: Place a direct order for a stock**

Primary actor: Bank clerk or customer

Supporting actors: Involved stock market.

1. The involved custody account has been selected.

Basic flow:

1. The actor selects to place a direct order on a stock.
 2. The bank system asks to select between buy or sell.
 3. The actor selects between buy or sell.
 4. The bank system asks for the ISIN code of the stock.
 5. The actor gives the ISIN code.
 6. The bank system asks for the amount of stocks involved.
 7. The actor gives the amount.
 8. The bank system verifies the order.
 9. The customer and bank sign the order.
 10. The bank system creates a new pending order.
 11. The bank system sends the order to the stock market.
 12. The stock market gives the order a unique number.
 13. The stock market associates the bank as agent for the order.
 14. The bank system and the stock market sign the order.
 15. The stock market confirms the receipt of the order.
 16. The bank system adds the order to the pending orders of the custody account.
 17. The bank system notifies the actor.
-

2.3.4.2 Use Case: Place a limited order for a stock**Primary actor:** Bank clerk or customer**Supporting actors:** Involved stock market.

1. The requesting customer has been selected.
2. The involved custody account has been selected.

Basic flow:

1. The actor selects to place a limited order on a stock.
 2. The bank system asks to select between buy or sell.
 3. The actor selects between buy or sell.
 4. The bank system asks for the ISIN code of the stock.
 5. The actor gives the ISIN code.
 6. The bank system asks for the amount of stocks involved.
 7. The actor gives the amount.
 8. The bank system asks for the limited price involved.
 9. The actor gives the limited price.
 10. The bank system verifies the order.
 11. The customer and bank sign the order.
 12. The bank system creates a new pending order.
 13. The bank system sends the order to the stock market.
 14. The stock market gives the order a unique number.
 15. The stock market associates the bank as agent for the order.
 16. The bank system and the stock market sign the order.
 17. The stock market confirms the receipt of the order.
 18. The bank system adds the order to the pending orders of the custody account.
 19. The bank system notifies the actor.
-

2.3.4.3 Use Case: Execute put option**Primary actor:** Bank clerk or customer**Supporting actors:** Involved issuer of the option.

1. The requesting customer has been selected.
2. The involved custody account has been selected.
3. The involved option has been selected.

Basic flow:

1. The actor selects to execute a put option
 2. The bank system asks for the amount of options involved.
 3. The actor gives the amount.
 4. The bank system verifies the execution.
 5. The customer and bank sign the execution.
 6. The bank system sends the execution to the issuing financial institution.
 7. The issuer verifies the execution.
 8. The bank system and the issuer sign the execution.
 9. The bank system sends an invoice to the issuer with the total amount to pay.
 10. The bank system updates the custody account by marking the involved options as executed, updating the balance of the account with the value of the sold stocks and updating the amount of stocks in the associated position.
-

2.3.4.4 Use Case: Execute call option**Primary actor:** Bank clerk or customer**Supporting actors:** Involved issuer of the option.

1. The requesting customer has been selected.
2. The involved custody account has been selected.
3. The involved option has been selected.

Basic flow:

1. The actor selects to execute a put option
 2. The bank system asks for the amount of options involved.
 3. The actor gives the amount.
 4. The bank system verifies the execution.
 5. The customer and bank sign the execution.
 6. The bank system sends the execution to the issuing financial institution.
 7. The issuer verifies the execution.
 8. The issuer gives a unique number to the execution.
 9. The bank system and the issuer sign the execution.
 10. The bank system updates the custody account by marking the involved options as executed. In case no position has been opened for the stock related with the option, a new position is opened. The amount of stocks in the new or already existing position is raised with the amount of bought stocks.
-

2.3.5 Stock market feedback

2.3.5.1 Use Case: Confirm a selling order

Primary actor: Stock market system

Basic flow:

1. The stock market system notifies the bank system that a certain selling order has been processed. The stock market gives the unique id of the order, the actual selling price and the unique id of the trading transaction. If the amount of stocks in the order could not be matched completely the stock market also provides the information of the new order that has been entered in the order book with the rest of the amount.
 2. The bank system and the stock market sign the confirmation.
 3. The bank system sends an invoice to the stock market with the total amount to pay.
 4. The bank system updates the custody account by confirming the pending order, specifying the actual selling price, updating the balance of the account and updating the amount of stocks in the associated position.
-

2.3.5.2 Use Case: Confirm a buying order

Primary actor: Stock market system

Basic flow:

1. The stock market system notifies the bank system that a certain buying order has been processed. The stock market gives the unique id of the initial order, the actual selling price and the unique id of the trading transaction. If the amount of stocks in the order could not be matched completely the stock market also provides the information of the new order that has been entered in the order book with the rest of the amount.
 2. The bank system and the stock market sign the confirmation.
 3. The bank system updates the custody account by confirming the pending order, specifying the actual buying price and updating the amount of stocks in the associated position.
-

2.3.5.3 Use Case: Deny an order

Primary actor: Stock market system

Basic flow:

1. The stock market system notifies the bank system that a certain order has been rejected.
2. The bank system asks for the unique id of the order and the reason for rejection.
3. The stock market gives the id and the reason for rejection.

4. The bank system and the stock market sign the rejection.
 5. The bank system updates the custody account by updating the pending order.
-

2.3.5.4 Use Case: Send an invoice for a trading transaction

Primary actor: Stock market system (or settlement organization) or issuer of an option.

Stakeholders: Customer, bank system, inter banking infrastructure.

Preconditions: The stock market has confirmed a buying order or a call option has been executed.

Basic flow:

1. The actor sends an invoice to the bank for a trading transaction in which the bank was a buying agent. This invoice contains the total amount to pay (buying price of the traded stocks and handling costs). The invoice also contains an invoice number, the unique number of the trading transaction and the account number of the actor.
 2. The bank system and actor sign the invoice.
 3. The bank system creates a charging transaction on the custody account of the customer with the total amount to pay and its own handling costs. This charging transaction is sent to the transaction handler.
 4. The bank system requests a transfer from the bank's own account to the account of the actor (the stock market or the settlement organization). The amount on this transfer is what the bank has to pay to the stock market for the trading transaction. This transfer contains a message that refers to the invoice number.
-

The use cases for depot inspection and depot management are not further specified in this document.

2.4 Use Cases for transaction processing

The transaction handler is a system that processes the transaction queue. All transactions (withdrawals, deposits, transfers and cost charges) created by the different users are stored in a transaction queue. Afterwards (mostly at night) this queue is processed in batch by the transaction handler. This system includes the following uses cases.

Operational management

- Start handling transaction queue

Transaction processing

- Process a deposit
- Process a withdrawal
- Process a transfer
- Process a charge

2.4.1 Operational management

2.4.1.1 Use Case: Start handling transaction queue

Primary actor: Operational manager.

Stakeholders: Bank system.

Basic flow:

1. The operational manager starts the transaction handler
 2. The transaction handler iterates the transaction queue and executes each transaction (see following use cases for details).
-

2.4.2 Transaction processing

2.4.2.1 Use Case: Process a deposit

Primary actor: Transaction handler.

Stakeholders: Bank system.

Preconditions: The transaction handler is processing the transaction queue

Basic flow:

1. The transaction handler starts the processing of a deposit transaction.
 2. The bank system adds the amount to the account.
 3. *The bank system updates the bank's internal accountancy.* (To be investigated)
 4. An entry with the transaction information is added to the overview of the completed transactions of the customers account.
 5. The transaction is removed from the transaction queue.
-

2.4.2.2 Use Case: Process a withdrawal

Primary actor: Transaction handler.

Stakeholders: Bank system.

Preconditions: The transaction handler is processing the transaction queue

Basic flow:

1. The transaction handler starts the processing of a withdrawal transaction.
 2. The bank system withdraws the amount from the account.
 3. *The bank system updates the bank's internal accountancy.* (To be investigated)
 4. An entry with the transaction information is added to the overview of the completed transactions of the customers account.
 5. The transaction is removed from the transaction queue.
-

2.4.2.3 Use Case: Process a transfer

Primary actor: Transaction handler.

Stakeholders: Bank system, inter banking infrastructure.

Preconditions: The transaction handler is processing the transaction queue

Basic flow:

1. The transaction handler starts the processing of a transfer transaction.
2. The bank system withdraws the transfer amount from the originating account.
3. The bank system requests a transfer from the bank's own account to the account of the bank of the destination account. This transfer contains a message that contains the details of the transaction: the originating account from the customer, the destination account from the other customer and a transfer message.

4. *The bank system updates the bank's internal accountancy.* (To be investigated)
 5. An entry with the transaction information is added to the overview of the completed transactions of the customers account.
 6. The transaction is removed from the transaction queue.
-

2.4.2.4 Use Case: Process a charge

Primary actor: Transaction handler.

Stakeholders: Bank system.

Preconditions: The transaction handler is processing the transaction queue

Basic flow:

1. The transaction handler starts the processing of a charge transaction.
 2. The bank system withdraws the amount from the account.
 3. *The bank system updates the bank's internal accountancy.* (To be investigated)
 4. An entry with the transaction information is added to the overview of the completed transactions of the customers account.
 5. The transaction is removed from the transaction queue.
-

2.5 Domain model

Complementary to the previous use cases, the following three figures describe the problem domain using a domain model. Figure 2.5 represents an overview of basic banking services. Figure 2.5 describes the investment service more in detail. Finally, Figure 2.5 zooms in on the multiple inheritance structure that describes order requests and orders.

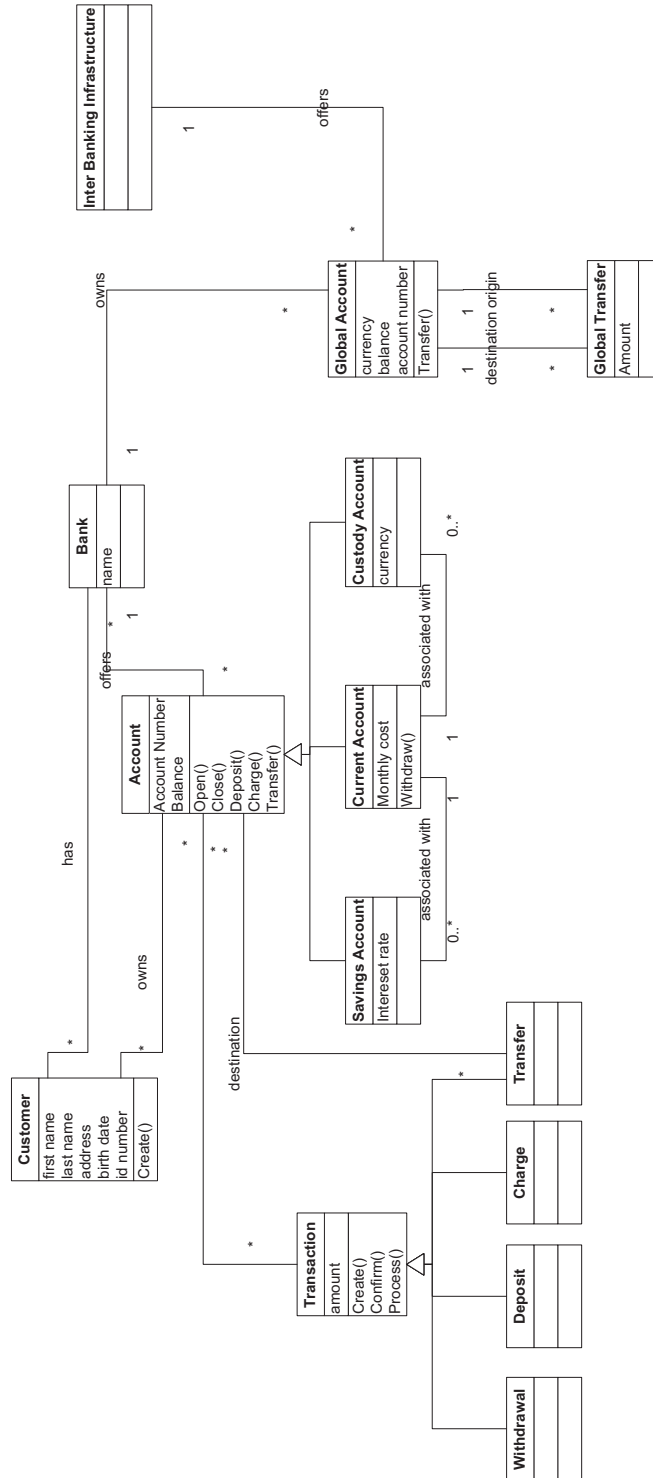


Figure 2.2: Domain model for basic banking services

Chapter 3

Security Requirements

This chapter elaborates on the security requirements for the e-finance case study. The requirements are mostly based on the use cases that were described in the previous chapter, and because of the focus of the SoBeNeT project we restrict ourselves deliberately to software security requirements purely, as opposed to network or hardware security requirements.

The structure of the chapter is as follows. In the first section we introduce the broader context of security for financial services. The main drivers in this context are the regulations and legislations. In a second section, we approach the case study from a risk analysis point by describing business and security risks of the e-finance case as misuse cases. Finally, we focus on the specification of security requirements, in the form of concrete policies, that address the needs that have been identified in the broader context.

3.1 Security and business context

In this section we give a short overview of the applicable regulations and legislations concerning e-finance applications. This overview is intended to be an introduction rather than an exhaustive overview. It will give insight into why certain policy rules are included. However, it is only background information and, hence, it can be skipped accordingly.

3.1.1 Applicable regulations and legislations

Applications in the financial world are subject to different statutory regulations and legislations. These determine to a great extent the security requirements of the system. For example, the following regulations may be applicable for organizations using the e-finance software:

- Privacy legislation on the storage and usage of personal data [3],[4]. This legislation requires:
 - the definitions of personal data, sensitive data, and data processing;
 - the definitions of all entities involved in data processing, their roles and responsibilities (controller, processor, operator, subject);

- the obligations relating to public and private data controllers with specific reference to the legitimate purpose of data processing and the adoption of minimal precautionary security measures to minimize the risks on data.
- E-commerce and the usage of electronic signatures [7], [10]
- The Banking and Finance Commission (BFC) defined a law [1], agreed upon by the federal government and currently effective, stating the regulation to prevent fraud, launder money and finance terrorism. The most important security requirements following from this law are the user identification process and the detection of atypical transactions.
- BASEL II[2]: risk management for financial organizations focuses amongst others on operational risk of which a substantial part is the technological risk of the IT infrastructure.

It is clear that there are a lot of other regulations applicable in the financial world, but starting with the above mentioned we have a meaningful contribution to the security requirements of the e-finance case study. In the rest of this section we will zoom into each of these topics and we address the most relevant topics.

3.1.1.1 Customer identification

One of the key elements in the process to prevent fraud, laundering money and financing terrorism is to avoid anonymous transactions. This means that the financial institutions must be able to identify the natural person who is holder of an account and doing the transaction. The financial institute must prevent the usage of accounts under a pseudonym or false identity. In the context of this case study we won't focus on the identification process itself with a face to face interaction, copy of the identity card, validation of the data with the national register, etc. but on the procedures within the bank on the usage and manipulation of this information. It is important how the link between the natural person and the electronic identity is maintained, what information is essential in defining this link and how is that information accessed and manipulated by the bank personnel. For example if the physical address is used to send a new pin code to a cardholder it is important to validate who can access, and modify the address. Next to this it is important to keep a record of changes of customer information in the system for future forensics research.

3.1.1.2 Fraud detection

Based on a dictionary definition, fraud is defined as a criminal deception; the use of false representations to gain unjust advantage. Bank fraud cost the financial organizations multiple billion dollars on a yearly basis. This is ten to fifteen times the taken in bank robberies annually. While they catch most of the suspects of bank robberies within 48 hour, fraudsters are hardly got. There are different types of bank and financial fraud:

- Check fraud
- New account fraud
- Identity fraud

- Credit/Debit card fraud
- ATM transaction fraud
- Wire fraud
- Loan fraud

The legislation and regulations for the prevention of fraud require a first-line and a second-line monitoring system to detect suspicious customer behavior and atypical transactions. The personnel of the front office, responsible for the customer facing, do the first-line monitoring. The second-line monitoring system must be an automatic system that implements the different rules mentioned in section 2.1.2. There are guidelines describing the handling of this kind of events including the information flows and reporting hierarchies.

Another issue concerns the actual measures that support fraud detection. What special cases must trigger an alerting system? What information is required to support the incident handling process? What information is required to support the root cause analysis? The research to identify the various ways in which bank fraud occurs is beyond the scope of this work, but we can take some examples from the community [5] as real-world cases for our e-finance case study. The fraudulent transactions based on the usage of checks, ATM's, credit cards and loans are omitted because they are not explicitly modeled in our case study.

Applied to the case study:

The following actions are defined as suspicious within the scope of our case study:

- A transaction type occurs above a specified number in 48 hours.
- More than one session is active at the same time.
- If any set of transaction causes more than 80% credit limit in 48 hours (one transaction or sum of transactions in the 48 hour period) or trying to withdraw more money than the limit in credit.
- Deposit activity out of the normal range for any account. For example an excessive numbers of deposited items, total deposit amounts greater than average, large deposited items masked by smaller deposit transactions, the amount exceeds the historical average deposit amount by more than a specified percentage, the number of deposits exceeds the normal activity by the customer, a customer make several bank deposits below a specified threshold, a rapid increase in the size and frequency of cash deposits with no corresponding increase in non-cash deposits

We do not decide yet whether these operations will be denied or just marked as suspicious by the system.

3.1.1.3 Data Protection

The Belgian laws [3] and the EC directives [4] on processing of personal data and the protection of privacy define the measures to be taken when personal information is used in an application. It is beyond the scope of the e-finance case study to implement every detail of these legislations and regulations. We select

some key rules that can contribute to the security requirements and illustrate its importance in the security requirements definition process.

Definition of the personal data Based on the definitions of Article 2 *Definitions* of directive 95/46/EC personal data means any information relating to an identified or identifiable natural person. The commission for safeguarding the personal privacy classifies the personal information in categories. An organization is only allowed to process personal information declared at the commission. An exhaustive list of categories and types of information can be found in the directive 95/46/EC we mentioned earlier.

Applied to the case study:

We summarize the categories of personal data that are considered relevant for the case study. A detailed view on personal data can be found in the security analysis (see Section 3.2.3).

1. Identification data
 - Personal identification data: name, title, address (home, work), phone number (home, work);
 - Identification numbers other than the national registration number, identity card number, passport number, license plate, pension number.
2. Financial data
 - Bank account numbers, secret code.
 - Savings.
 - Investments.
3. Personal characteristics
 - Age, sex.

In the context of the e-finance case study we can map the following categories to these two actors (see next section for a complete overview of all actors):

- For customers: 1, 2 and 3.
- For employees: 1 and 3.

The information of the different subcategories needs to be treated differently depending on the intended data processing processes. In the next paragraphs we explain this data processing process and the important entities and their roles in it.

Definition of the data processing process Based on the directive 95/46/EC (definitions of Article 2 *Definitions*), processing of personal data means any operation or set of operations which is performed upon personal data whether or not by automatic means such as collection, recording, organization, storage, adaptation or alteration, retrieval, consultation, use, disclosure by transmission, dissemination or otherwise making available, alignment or combination, blocking, erasure or destruction. The main purpose of collecting the personal data of the customer is for enabling the banking business:

- To support the customer identification process (e.g. national registration number)

- To determine the profile of the customer (e.g. investor profile)
- To determine the credit limit (e.g. salary, outstanding debts, mortgage, ...)
- To interact with customer and exchange information (e.g. pin, to mail statement of an account, etc)
- ...

Important note: if personal information is used in a critical decision it is important to validate its correctness and audit all modification before and after the decision.

Identification of all entities involved in the data processing After the identification of the personal data and the required processing tasks, it is important to identify the entities with their roles and responsibilities that are involved in the data processing. This should allow a clear and unambiguous identification of the responsible person in case malicious transactions are detected or private information is leaked. It is not the intention and definitely beyond the scope of this document to model a complete organization structure of a financial institution. Nevertheless it is important when developing an e-finance system that *certain roles and responsibilities* exist for the correct and secure working of the system. We are trying to select a few specific roles and responsibilities that shed a new light on the definition of the security requirements of the e-finance case study. We would like to use the following definitions as described in Article 2 *Definitions* of the directive 95/46/EC:

Data Subject is an identified or identifiable natural person; an identifiable person is one who can be identified, directly or indirectly, in particular by reference to an identification number or to one or more factors specific to his physical, physiological, mental, economic, cultural or social identity.

Applied to the case study:

In the e-finance case the data subjects are primarily the customers. The customers have always the right to retrieve and to correct its personal data stored in the bank. The bank personnel can also be taken into account if we want to detect complex fraud within the banking organization. But this may go beyond the scope of our case study.

Data Controller is the natural or legal person, public authority, agency or any other body which alone or jointly with others determines the purposes and means of the processing of the personal data.

Applied to the case study:

In the e-finance case the data controllers are the heads of three groups namely:

- Front Office Financial Transactions: this group is responsible for all customer-facing issues. These are the people in the different branches and the people of the call center supporting the helpdesk.

- **Back Office Operations:** this group provides technical support to the front office. They are responsible for the correct working of the IT systems including the daily operational and system management tasks. They are also responsible for the correct working of the home-banking facilities and the systems interacting with 3rd parties (e.g. the stock market).
- **Internal Audit:** this group of people is independent of the other two groups and their responsibilities are twofold. They have to validate if (1) the internal defined procedures are in line with the applicable legislations and (2) the internal defined procedures are also executed in practice. This validation needs to be done for the practices in front office as well as in the back office.

The heads of these groups determine the policies and procedures that their team members need to follow. We make abstraction of the other divisions (e.g. Human Resources, Legal, etc) that are typically in a financial institution.

Data Processor is a natural or legal person, public authority, agency or any other body, which processes personal data on behalf of the controller.

Applied to the case study:

In the e-finance case the data processors are the team members of the groups defined in the paragraph above.

- **Front Office Financial Transactions group:**
 - **Office Manager:** this person is the head of a branch office and as such responsible for the definition of the roles of the branch personnel and all the transactions done via the desk. He will take final decisions in the opening/closing of a bank account and the definitions of the credit limits.
 - **Bank Clerk:** this person is serving the customers visiting the desk. He can handle the administrative part of the transactions. Some transactions can only be completed if the Office Manager authorizes them.
- **Back Office Operations group:**
 - **System Administrators:** these persons are responsible for the installation and maintenance of the systems running the back office applications. They deploy the new software releases of operating systems and business applications.
 - **Security Manager:** this person is responsible for the design, implementation and operational management of the IT security infrastructure such as firewalls, intrusion detectors, authentication and authorization servers, etc.
 - **User Account Manager:** this person is responsible for the overall creation of user accounts in the application. These are the accounts known at the application level, not at the OS or DBA level.

- Internal Audit group: they are responsible for (1) the validation of procedures and logs for prevention against fraud, money laundry and financing of terrorism, but also for (2) the validation of procedures and logs for the operational management of the IT infrastructure.

Minimal precautionary security requirements The privacy legislation and the protection of personal information dictates that the data kept in the system must be protected according to the current applicable standards. But there is no specific technology choice defined to implement the security measures for authentication, authorization, antivirus, data backup and restore, . . .

Applied to the case study:

An example of this requirement is that at least 2-factor authentication is required to login to the banking system via the internet (e.g. internet banking), but bankcard and pin code authentication is sufficient at the cash desk. We will elaborate on this in the security requirements on authentication and customer identification.

3.1.1.4 Electronic signatures

The Belgian laws [7], [8], [9] and the EC directive 1999/93/EC [10] on the community framework for electronic signatures define some guidance and high-level requirements for the application developers. For example if electronic signatures are based on public/private key technology with certificates, the requirements for these qualified certificates are described in Annex I from [10]: Qualified certificates must contain:

- an indication that the certificate is issued as a qualified certificate;
- the identification of the certification-service-provider and the state in which it is established;
- the name of the signatory or a pseudonym, which shall be identified as such;
- provision for a specific attribute of the signatory to be included if relevant, depending on the purpose for which the certificate is intended;
- signature-verification data which corresponds to signature-creation data under the control of the signatory;
- an indication of the beginning and end of the period of validity of the certificate;
- the identity code of the certificate;
- the advanced electronic signature of the certification-service-provider issuing it;
- limitations on the scope of use of the certificate, if applicable; and
- limits on the value of transactions for which the certificate can be used, if applicable.

Other annexes contain the requirements for the certification-service providers issuing qualified certificates, secure signature-creation devices and recommendations for secure signature verification. However, initially the legal validity

of digital signatures was limited to the usage of public-private key pair encryption and certification authorities. Later a technological-natural terminology was introduced with a subtle difference between digital signatures and electronic signatures. The former is related to a technology and the latter is related to the legal concept. The intersection of both is obviously not empty. See [6] for a detailed discussion on this subject. In our case this means that we can define an *advanced electronic signature* as an electronic signature meeting the following four requirements:

1. uniquely linked to the signatory;
2. capable of identifying the signatory;
3. created using means that the signatory can maintain under his sole control;
4. linked to the data to which it relates in such a manner that any subsequent change of the data is detectable.

3.1.1.5 Audit requirements

Audit requirements are defined by an *external* and *independent* organization that defines a set of controls the system should be compliant with. An auditor verifies if a system is working properly and if the right operational procedures are in place conform the selected controls.

Validation of the checklists for validating the controls may result in requirements like for example logging of all employee transactions. This may provide an answer on questions like:

- What bank employees have logged on the system in period X? What are the roles and access rights of an employee at timestamp Y?
- What are the accounts accessed by a specific bank clerk today?
- What are the customers created by a specific bank clerk?
- What accounts are created, and closed in a period of 6 months?
- ...

Next to the information in the audit trail, it is also important to define the security requirements of the audit subsystems itself.

- What mechanisms are required to avoid tampering with the audit information?
- Who can enable/disable auditing?
- How to protect sensitive data (for example user id's) against discovery and misuse?
- How to ensure that information in audit trails does not compromise system security?
- ...

From [5] : Financial institutions should maintain, for at least five years, all necessary records on transactions, both domestic or international, to enable them to comply swiftly with information requests from the competent authorities. Such records must be sufficient to permit reconstruction of individual

transactions (including the amounts and types of currency involved if any) so as to provide, if necessary, evidence for prosecution of criminal activity. Financial institutions should keep records on the identification data obtained through the customer due diligence process (e.g. copies or records of official identification documents like passports, identity cards, driving licenses or similar documents), account files and business correspondence for at least five years after the business relationship is ended. The identification data and transaction records should be available to domestic competent authorities upon appropriate authority.

3.2 Security analysis

This section first describes the different actors from a security point of view. Afterwards, we describe the possible misuse cases based on the set of use cases described in the previous chapter. At the end of this section, we reiterate on the business model with respect to data sensitivity.

3.2.1 Important actors and stakeholders

The following extra roles are introduced based on the security requirements:

User account manager He manages the users of the banking system. He can create and block user accounts.

Security manager The security manager assigns roles and privileges to the user accounts of the banking system.

External audit organization Organization that defines a set of controls the system should be compliant with.

Internal auditor An auditor verifies whether a system is working properly and whether the right operational procedures are in place conform the selected controls. They are responsible for (1) the validation of procedures and logs for prevention against fraud, money laundry and financing of terrorism, but also for (2) the validation of procedures and logs for the operational management of the IT infrastructure.

Attacker Despite of not being a designed actor of the system, an external attacker (active or passive) will also come into play when analyzing security requirements.

3.2.2 Misuse Cases

The functional requirements in this document have been defined by means of use cases (including administration, transactions, for account inspection, and so forth). These use cases describe the desirable operation of the application and are a good basis for the functional requirements. When it comes to the definition of security requirements it can help to document the *misuse* or *abuse* cases and to analyze the risks related to them. This section describes misuse

cases at the business level based on the functional use cases, and not at the technical/design level. The misuse cases are defined using a textual description based on templates described in [11] and [12].

Remark that, by definition, the list of misuse cases is never complete. Some of the cases will be refined, and new cases will be defined during the further development of the case study. Some cases are obvious but nevertheless they illustrate the importance of the countermeasures that are defined in the more technical security requirements.

3.2.2.1 Misuse cases for customer and account management

MisUse Case: Create false customer

- **Summary:** A customer is created with a false identity.
- **Primary mis-actor:** Bank employee (bank clerk, system administrator, database administrator, etc.)
- **Basic flow:**
 - bf1. The mis-actor selects to create a new customer
 - bf2. The system asks for the required information about the customer
 - bf3. The mis-actor gives the false information
 - bf4. The mis-actor circumvents the verification process via the use of a stolen identity
 - bf5. The system creates a new *false* customer when verification succeeded
- **Alternative flow:**
 - af1. The mis-actor circumvents the verification process via non-existent personal data e.g. bogus address, name, etc. (for bf4)
- **Capture points:**
 - cp1. The verification process needs involvement of more than 1 person
 - cp2. The personal information of a new customer is automatically verified against multiple sources (for example validation of address at the national register)
 - cp3. There is an audit trail of the bank employees' actions that is reviewed regularly
- **Triggers:**
 - tr1. Always true, i.e. this can happen at any time
- **Preconditions:**
 - pc1. The system allows entrance of unverified data by 1 person
 - pc2. There is a weak checking/validating/matching procedure between electronic identity and real world identity
- **Assumptions:**
 - as1. The mis-actor can enter data at a location where no validation/verification is done

- **Worst-case threat:** A malicious bank employee can create dummy customers and accounts to support fraudulent transactions
 - **Prevention guarantee:** No false customers can be created in the system
 - **Detection guarantee:** Potentials false customers are detected and removed, and the malicious bank employee can be identified.
 - **Stakeholders and threats:**
 - sh1. The financial institution:
 - * Loss of money that is expended via the false account
 - * Non-compliance with the regulations
 - sh2. The citizen:
 - * The innocent citizen whose identity is stolen must pay back the money or proves his innocence.
 - **Scope:** Customer registration process
-

MisUse Case: Show customer information

- **Summary:** Customer information is browsed or looked up for the interest of the bank employee without a concrete customer request or authorization.
- **Primary mis-actor:** Bank employee (bank clerk, system administrator, database administrator, etc.)
- **Basic flow:**
 - bf1. The mis-actor chooses to search for the customer
 - bf2. The system requests the search criteria
 - bf3. The mis-actor can use any personal information field to query the repository
 - bf4. The system shows a list of matching customers
 - bf5. The mis-actor selects the target customer
 - bf6. The system show all the customer information
- **Alternative flow:**
- **Capture points:**
 - cp1. The bank employee can only request customer information if the customer is at the desk and authorize the system to do so (e.g. by inserting the credit card and pin)
 - cp2. There is a hierarchical authorization to lookup and browse customer information targeted for example for direct marketing, etc.
 - cp3. There is an audit trail of the bank employees' actions that is regularly reviewed.
- **Triggers:**
 - tr1. Always true, i.e. this can happen at any time.
- **Preconditions:**

pc1. The system allows querying all customer personal information in any branch

- **Assumptions:**

- **Worst-case threat:** A malicious bank employee can see, misuse, and resell the financial situation of any bank customer

- **Prevention guarantee:** An individual bank employee cannot browse customer information without proper authorization

- **Detection guarantee:** The malicious bank employee can be identified if anomalous information querying is detected. Screening can be done manually by internal audit or automatically using data mining techniques. There is a difference between the type of actions done by a bank clerk and the type of actions done by a system or database administrator.

- **Stakeholders and threats:**

sh1. The customer

* Personal information of an innocent customer is misused

- **Scope:** Customer management system

MisUse Case: Edit customer information

- **Summary:** A bank employee modifies customer information without a concrete customer request or authorization.

- **Primary mis-actor:** Bank employee (bank clerk, system administrator, database administrator, etc.)

- **Basic flow:**

bf1. The mis-actor edits the customer's information

bf2. The mis-actor gives the false information

bf3. The mis-actor circumvents the verification process (e.g. use parts of stolen identity)

bf4. The system updates the customer's record with incorrect data.

- **Alternative flow:**

- **Capture points:**

cp1. The bank employee can only request and modify customer information if the customer is at the desk and authorizes the system to do so (e.g. by inserting the credit card and pin)

cp2. The modification process of critical data needs involvement of more than 1 person

cp3. The personal information of a customer is automatically verified against multiple sources (for example validation of address at the national register, ...) after every modification.

cp4. The bank system keeps a detailed history of all modifications on customers' information.

- **Triggers:**

tr1. Always true, i.e. this can happen at any time.

- **Preconditions:**
 - **Assumptions:**
 - **Worst-case threat:** A malicious bank employee can update personal data (e.g. address information) to intercept secret information (pin code) and steal customer's money.
 - **Prevention guarantee:** Undesirable modifications of personal information are avoided.
 - **Detection guarantee:** The malicious bank employee can be identified if anomalous information modification is detected.
 - **Stakeholders and threats:**
 - sh1. The customer
 - * Personal information and money of an innocent customer are misused
 - **Scope:** Customer management system
-

MisUse Case: Create a false account

- **Summary:** A bank employee adds a false account without a concrete customer request or authorization.
- **Primary mis-actor:** Bank employee (bank clerk, system administrator, database administrator, etc.)
- **Basic flow:**
 - bf1. The mis-actor adds a false account to the customer's information record
 - bf2. The mis-actor circumvents the verification process (e.g. via access to the data after the authorization signature is validated)
 - bf3. The system adds the false account to the customer's information record and is ready to accept transactions from that account.
- **Alternative flow:**
- **Capture points:**
 - cp1. The authorization system is implemented at all levels in the banking system.
 - cp2. The bank system keeps a detailed history of all operations on account creation and deletion.
- **Triggers:**
 - tr1. Always true, i.e. this can happen at any time.
- **Preconditions:**
- **Assumptions:**

- **Worst-case threat:** A malicious bank employee can create false accounts and transfer money from it.
 - **Prevention guarantee:** No false accounts can be created in the system
 - **Detection guarantee:** The malicious bank employee can be identified if the false account is detected, e.g. after complaints of the customer.
 - **Stakeholders and threats:**
 - sh1. The customer
 - * Personal information and money of an innocent customer is mis-used.
 - **Scope:** Customer management system
-

MisUse Case: Close an account with a negative balance

- **Summary:** A bank employee closes an account with a negative balance and embezzles the money.
- **Primary mis-actor:** Bank employee (bank clerk, system administrator, database administrator, etc.)
- **Basic flow:**
 - bf1. The mis-actor transfers money from a *non-existing* account of another bank to virtually put the balance of the account on zero.
 - bf2. The mis-actor closes the account before the transaction of (1) is verified
 - bf3. The system removes the account.
- **Alternative flow:**
- **Capture points:**
 - cp1. The closing of an account can only be done after verification/execution of all pending transactions
 - cp2. The bank system keeps a detailed history of all operations on account creation, deletion or disabling
- **Triggers:**
 - tr1. Always true, i.e. this can happen at any time.
- **Preconditions:**
 - pc1. The banking system works with temporal balance in the front-end.
- **Assumptions:**
- **Worst-case threat:** A malicious bank employee can embezzle money from the bank.
- **Prevention guarantee:** The account remains open until all pending transactions are successfully executed.
- **Detection guarantee:** The malicious bank employee can be identified if an account with negative balance is closed.

- **Related business rules:**

- br1. The balance of an account must be zero before it is closed

- **Stakeholders and threats:**

- sh1. The financial institution:

- * Loss of money that is expended via the closed account

- **Scope:** Transactional system

3.2.2.2 Misuse cases for transactions

MisUse Case: Withdraw from a current account below credit limit

- **Summary:** A customer tries to withdraw more money from the current account than allowed by the credit limit

- **Primary mis-actor:** Bank customer

- **Basic flow:**

- bf1. The mis-actor visits a branch office to withdraw money until the credit limit

- bf2. The mis-actor visits immediately afterwards an ATM to withdraw extra money

- **Alternative flow:**

- af1. The mis-actor visits multiple ATM's or branch offices.

- af2. The mis-actor visits multiple shops where card payment is possible.

- af3. This misuse case scenario is executed on a false account/customer record.

- **Capture points:**

- cp1. Increasing the on-line transactional capabilities of the system reduces the window of vulnerability.

- cp2. Limit the maximum cash withdrawal in ATM's.

- cp3. Log the details of the accounts with a deficit below credit limit

- **Triggers:**

- tr1. Always true, i.e. this can happen at any time.

- **Preconditions:**

- pc1. The front office and ATM's are connected in near real-time for the transaction processing, resulting in a window of vulnerability

- **Assumptions:**

- as1. The mis-actor has a valid bank card to withdraw cash or pay in a shop.

- **Worst-case threat:** A mis-actor uses a higher credit limit than allowed. The system enters a state where the invariant of an account balance is false (i.e. balance below credit limit)
 - **Prevention guarantee:** Due to the scale of the banking system, a near real-time process is required resulting in the window of vulnerability. As a consequence this kind of transactions cannot be prevented.
 - **Detection guarantee:** The customer and account is signaled at the moment the transaction occurs.
 - **Related business rules:**
 - br1. The debit balance of an account is less than the credit limit.
 - **Stakeholders and threats:**
 - sh1. The financial institution
 - * A customer can use more money than the credit limit. The risk related to this may be seen as low if the customer coupled to the account is known, but becomes higher in case the account or the customer record is false.
 - **Scope:** Transactional system
-

MisUse Case: Deposit on a current account without delivering cash

- **Summary:** A bank employee tries to deposit money on a current account without delivering cash.
- **Primary mis-actor:** Bank employee (bank clerk, system administrator, database administrator, etc.)
- **Basic flow:**
 - bf1. The mis-actor starts a deposit
 - bf2. The mis-actor creates a new deposit
 - bf3. The system asks the amount to deposit
 - bf4. The mis-actor enters the amount to deposit
 - bf5. The system asks to sign the deposit
 - bf6. The mis-actor signs the deposit
 - bf7. The system sends the deposit to the transaction handler.
 - bf8. The mis-actor avoids the registration of the deposit used for the cash up
- **Alternative flow:**
 - af1. The mis-actor circumvent the verification in the transaction handler
- **Capture points:**
 - cp1. Final deposit transaction contains additional information such as branch office, bank clerk, terminal id, time stamp etc.
 - cp2. Daily matching between net cash income in a branch office and the transactions registered at that branch office.

- **Triggers:**
 - tr1. Always true, i.e. this can happen at any time.
 - **Preconditions:**
 - **Assumptions:**
 - **Worst-case threat:** A mis-actor can *create* electronic money out of the blue (i.e. without delivering cash)
 - **Prevention guarantee:** Increase the frequency for matching the content of the cash register with the amount of transferred money. This allows early detection of error prone calculations.
 - **Detection guarantee:** The customer and account is signaled at the moment the transaction occurs but the success of failure of this notification depends on the channel used.
 - **Related business rules:**
 - **Stakeholders and threats:**
 - sh1. The financial institution
 - * Loss of electronically created money without real cash deposits.
 - **Scope:** Transactional system, Front office application
-

MisUse Case: Create an illegal transaction

- **Summary:** A customer or a bank employee tries to do an illegal transaction
- **Primary mis-actor:** Bank customer or bank employee (bank clerk, system administrator, database administrator, etc.)
- **Basic flow:**
 - bf1. The mis-actor selects an account number he does not own and transfers money from
- **Alternative flow:**
 - af1. The mis-actor uses a memo-date in the past
 - af2. The mis-actor transfers an amount above the credit limit
- **Capture points:**
 - cp1. The transaction parameters and constraints need to be validated (and signed?) in every stage of the execution flow
- **Triggers:**
 - tr1. Always true, i.e. this can happen at any time.
- **Preconditions:**
- **Assumptions:**
- **Worst-case threat:** An illegal transaction is executed and has bypassed all the validations

- **Prevention guarantee:** All illegal transaction attempts are detected
 - **Detection guarantee:** The illegal transaction is executed, but the mis-actor can be identified
 - **Stakeholders and threats:**
 - sh1. The financial institution
 - * Loss of money that is transferred via the illegal transaction
 - * Non-compliance with the regulations
 - sh2. The customer
 - * Loss of money that is transferred via the illegal transaction
 - **Scope:** Transactional system
-

3.2.2.3 Misuse cases for the banking system

MisUse Case: Transfer between the bank's own account and another financial institution

- **Summary:** An inter banking transfer is modified before the transaction is sent to the inter banking service
- **Primary mis-actor:** Bank employee (system administrator, database administrator, etc.)
- **Basic flow:**
 - bf1. The mis-actor, employee of bank A, transfers money from his account in bank A to an account in bank B.
 - bf2. The system withdraws the amount from the mis-actors account in bank A
 - bf3. The system adds the transaction to the list of transferred items for bank B
 - bf4. The mis-actor modifies the details of the list of transferred items (e.g. double the amount of money of the transaction)
 - bf5. The system continues processing the pending transactions of the day
 - bf6. At the end of the day the global transfer transaction is signed by bank A and sent to the inter banking service with bank B as destination.
 - bf7. Bank B transfers double the amount of money to the mis-actors account on bank B
- **Alternative flow:**
- **Capture points:**
 - cp1. There is matching between the final signed inter banking transfer and the sum of the individual transactions
 - cp2. The withdrawal of an account is done in the same transaction as the transfer of the money to the other financial institution via the inter banking service
- **Triggers:**

tr1. Always true, i.e. this can happen at any time.

- **Preconditions:**

pc1. The mis-actor can intercept and modify his own transactions in the back office

- **Assumptions:**

- **Worst-case threat:** A mis-actor can transfer money owned by his bank A to an account of bank B.

- **Prevention guarantee:** No modified transaction can be inserted in the whole processing chain.

- **Detection guarantee:** Details of the transactions reveal the target bank and account number to trace the mis-actor, but since no customer is harmed and no one will complain, the misuse is only detected via detailed audit or automatic analysis tools.

- **Stakeholders and threats:**

sh1. The financial institution

* Loss of money that is transferred via the modified transaction

- **Scope:** Transactional system

3.2.2.4 Misuse cases for investment services

MisUse Case: Custody account management The misuse cases for administration defined for current and savings accounts are also applicable for custody accounts:

- Creation of a false custody account
 - Edit customer information required to obtain a custody account (e.g. investment profile, credit limit, etc.)
 - Browse information of custody accounts
-

MisUse Case: Catalogue inspection and false information

- **Summary:** Bank employee wrongly informs the customer to stimulate specific stock trading

- **Primary mis-actor:** Bank employee (bank clerk, system administrator, database administrator, etc.)

- **Basic flow:**

bf1. The customer requests an overview of a stock

bf2. The bank system asks for the ISIN code of the stock

bf3. The customer gives the ISIN code

bf4. The bank system asks the stock market catalogue system for detailed data of the stock.

- bf5. The catalogue system gives the bank system the requested data.
- bf6. The mis-actor manipulates the data to mislead the customer
- bf7. The bank system shows an overview of the manipulated stock data
- bf8. The customer buys/sells stocks via signed transactions based on incorrect information.
- bf9. The customer can afterwards detect his mistakes but can't prove that the information distributed was false since it is volatile data.

- **Alternative flow:**

- **Capture points:**

- cp1. The customer can verify the integrity of the data catalogue system
- cp2. The integrity of the information flow between the different subsystems in the solution is preserved.
- cp3. The bank system logs all data received from the catalogue system and all data that has been send in a customer session.

- **Triggers:**

- tr1. The customer is interested in a stock at the moment the mis-actor is active and the market conditions of a particular stock are suitable.

- **Preconditions:**

- **Assumptions:**

- as1. The customer does not double-check the stock market information via other channels before he decides to buy/sell stocks.

- **Worst-case threat:** The mis-actor is rewarded because a customer trades stocks based on wrong information.

- **Prevention guarantee:** The attempt to modify information sent to the customer is detected and blocked.

- **Detection guarantee:** The falsification of information is not avoided but the facts can be proven based on internal analysis on the log files

- **Stakeholders and threats:**

- sh1. The financial institution
 - * Legal sanctions for misinforming the customers
 - * Bad reputation towards customers and investors
- sh2. The customer
 - * Loss of money due to transaction with the wrong share price

- **Scope:** Can be multiple systems depending on the architecture

MisUse Case: Misusing the custody account

- **Summary:** A customer tries to circumvent the *business rules* on the custody account via stock trading
 - **Primary mis-actor:** Bank customer
 - **Basic flow:**
 - bf1. The mis-actor selects the stock to trade
 - bf2. The mis-actor tries to buy more stocks than the credit limit on his custody account A: (i) via one order with a higher amount of stocks B, (ii) via multiple orders in one or multiple stocks C, (iii) via direct orders where the final transaction price can be higher than expected D or (iv) via multiple delayed orders
 - **Alternative flow:**
 - af1. The mis-actor tries to sell stocks that he does not own
 - **Capture points:**
 - cp1. The bank system takes all pending transactions on all markets into account.
 - cp2. The integrity of the information flow between the different subsystems in the solution is preserved.
 - **Triggers:**
 - tr1. Always true, i.e. this can happen at any time.
 - **Preconditions:**
 - **Assumptions:**
 - **Worst-case threat:** A mis-actor uses a higher credit limit than allowed.
 - **Prevention guarantee:** Enabling direct (i.e. unlimited price) orders restricts the implementation of preventive measures.
 - **Detection guarantee:** The customer and account is signaled at the moment the transaction occurs.
 - **Stakeholders and threats:**
 - sh1. The financial institution
 - * A customer can use more money than the credit limit allows.
 - **Scope:** Transactional system
-

MisUse Case: Creation of illegal stock trade transactions

- **Summary:** A bank employee tries to do illegal stock trading transactions
- **Primary mis-actor:** Bank employee (bank clerk, system administrator, database administrator, etc.)
- **Basic flow:**

- bf1. The mis-actor selects the stock to trade
- bf2. The mis-actor modifies the date of the transaction to a date in the past

- **Alternative flow:**

- af1. The mis-actor modifies the number of shares in the transaction
- af2. The mis-actor modifies the price limit in the transaction

- **Capture points:**

- cp1. The integrity of the information flow between the different subsystems in the solution is preserved.

- **Triggers:**

- tr1. Always true, i.e. this can happen at any time

- **Preconditions:**

- **Assumptions:**

- **Worst-case threat:** The mis-actor is able to execute illegal transactions.

- **Prevention guarantee:** The attempt to execute illegal transactions is detected and blocked.

- **Detection guarantee:** The falsification of information is not avoided but the facts can be proven based on internal analysis on the log files (chain of trust and evidence)

- **Stakeholders and threats:**

- sh1. The financial institution
 - * Loss of money by illegal transactions
 - * Bad reputation towards customers and investors

- **Scope:** Can be multiple systems depending of the architecture
-

3.2.3 Business model and data sensitivity

The following three figures describe the sensitivity of the business objects described in the domain model. Figure 3.2.3 represents an overview of data sensitivity with respect to basic banking services. Figure 3.2.3 describes data sensitivity for the investment service more in detail. Finally, Figure 3.2.3 describes data sensitivity of order requests and orders.

All business objects in the domain model that are considered personal information of the customers as defined by the privacy legislation are colored red. All public information is colored green. White objects are considered private information of the financial institution it belongs to.

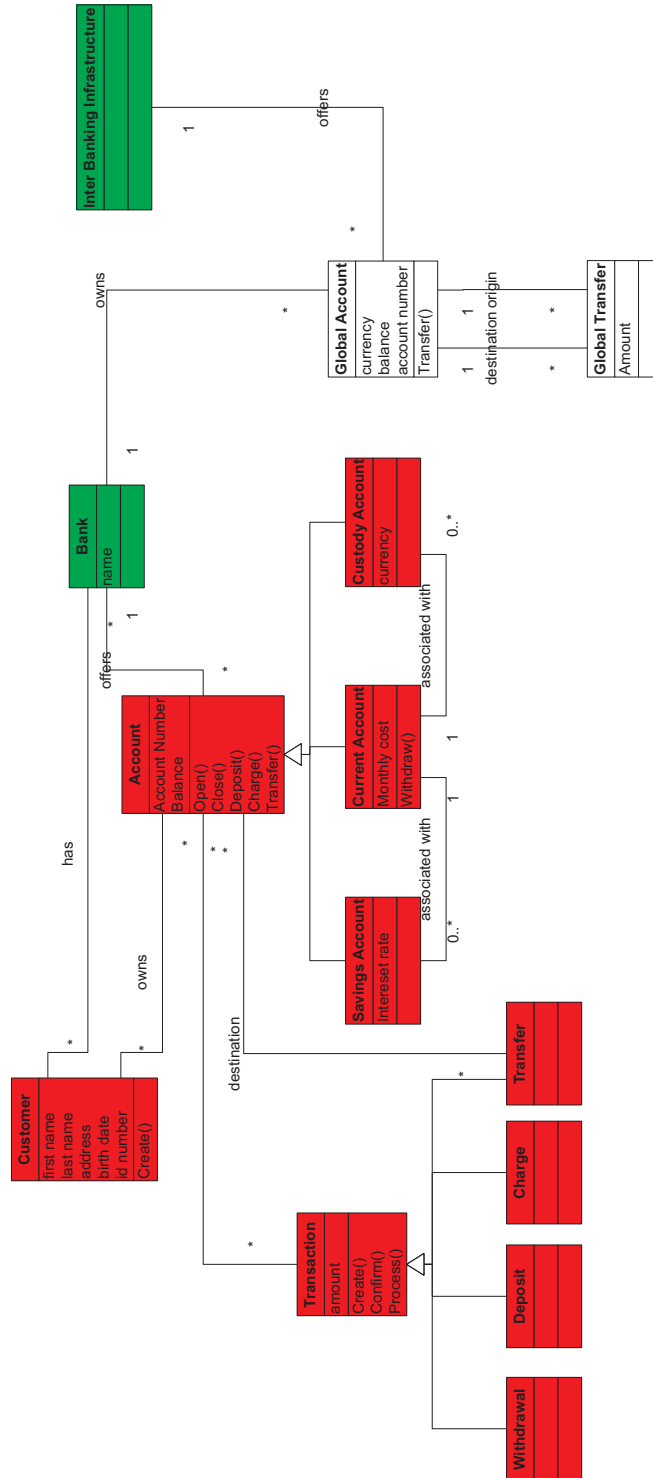


Figure 3.1: Data sensitivity model for basic banking services

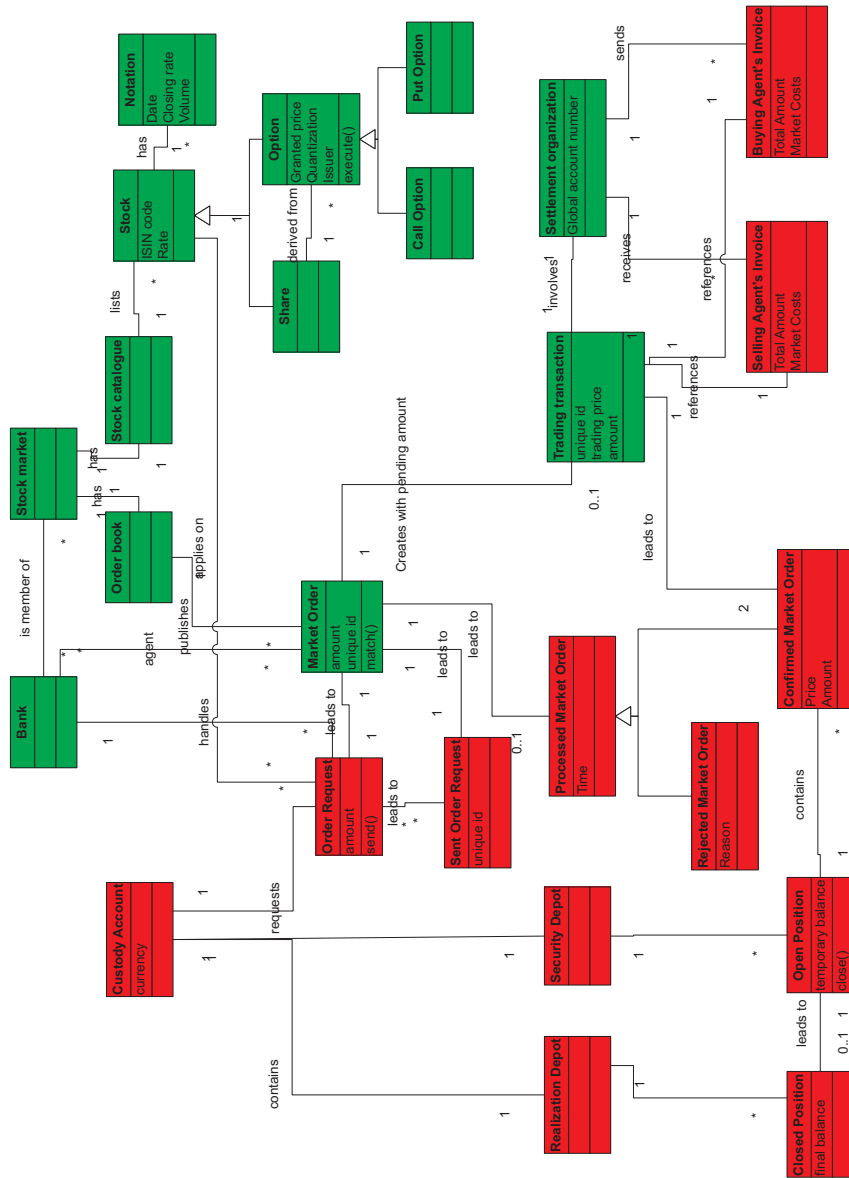


Figure 3.2: Data sensitivity model for investment services

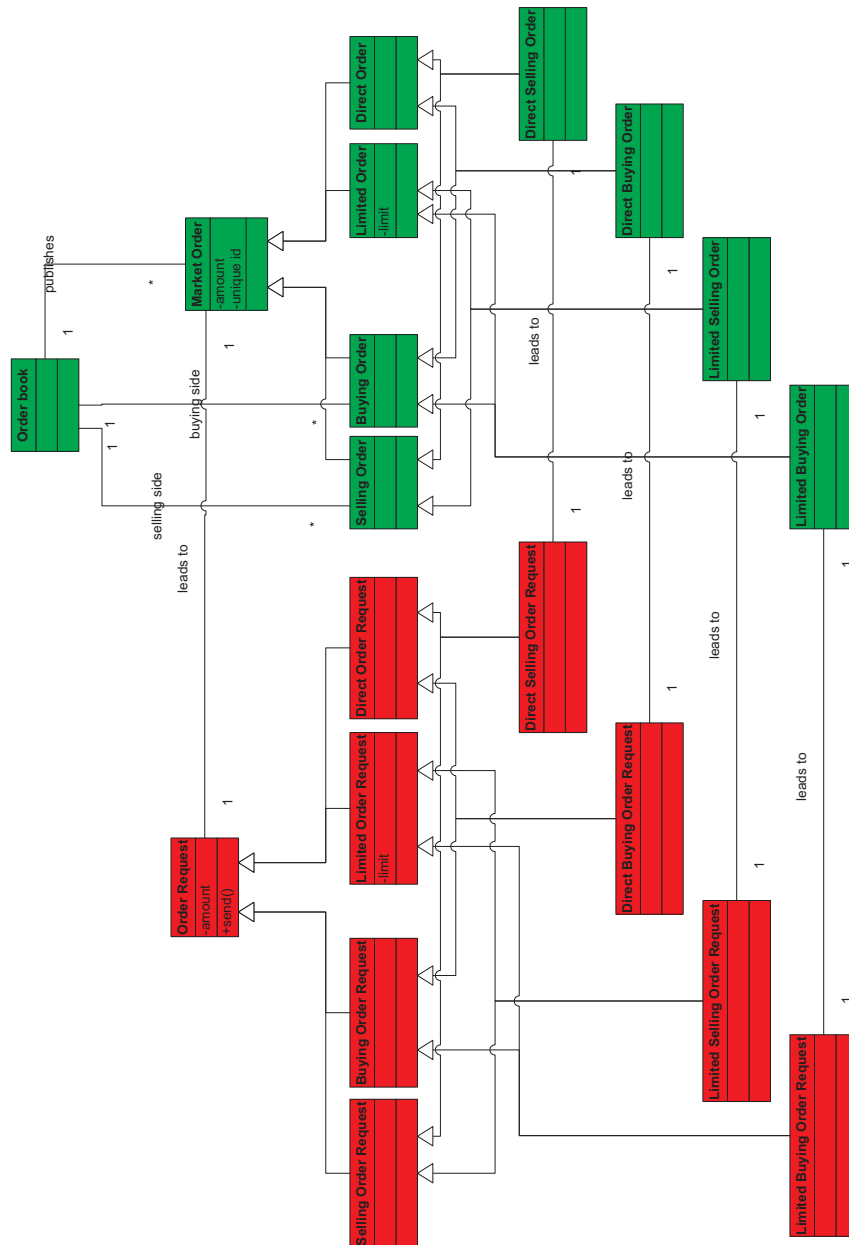


Figure 3.3: Detailed data sensitivity model for orders and their relationship

3.3 Security requirements

The security requirements secure that the system needs to resist unauthorized usage while still providing its services to legitimate users. These requirements need to be supported by a set security services. The following categories of security requirements and services can be distinguished:

- **Authentication:** protection against masquerading.
- **Access control:** protection against non-authorized access to functionality provided by the system.
- **Confidentiality:** protection against non-authorized access to information.
- **Integrity:** protection against non-authorized creation, altering or removal of information.
- **Non-repudiation and accountability:** protection against false denial of participation in communication or certain actions.
- **Audit:** verification of systems and operational procedures to assess whether they are conform to a selected set of rules.

To support the definition of security requirements, we include a very basic architecture of the e-finance system (see Figure 3.3). The main purpose of this architecture is to illustrate the different subsystems and their relationships.

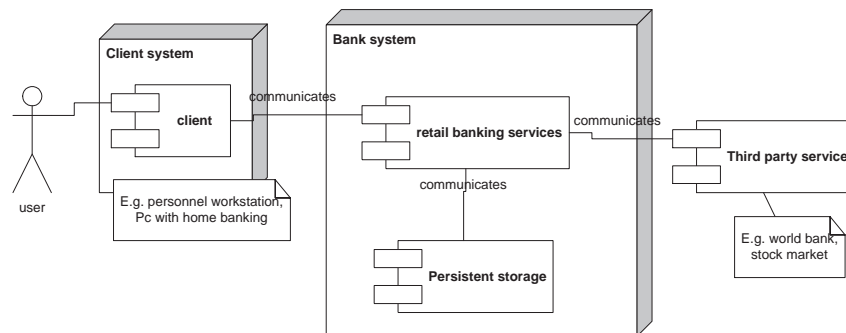


Figure 3.4: Basic architecture of the e-finance case

3.3.1 Identification and Authentication

Authentication relates to the identification of entities or data involved in the execution of the system. In this section, we will first explain the identification of a person before he/she can use the system as personnel or customer. The authentication requirements for the system can be subdivided into two major classes: entity authentication and data authentication, which will be explained subsequently.

3.3.1.1 Identification of the users

Based on the business security requirement *not allowing anonymous transactions*, be it the creation of a customer, opening of an account or a bank transaction, the person willing to execute a transaction, needs to be identified before he can use the system. This involves a user registration process, but also involves the management of the registered users later on.

User Registration The main purpose of the user registration process is to link a real world person with an electronic identity. Since identity theft is a growing crime and yearly responsible for multi billions of dollars fraud, it is important that the user registration process is achieved in a secure way. This process should consists of two phases preferably executed by two different persons:

1. Identity registration: the personal data (name, address, SSN, number of the identity card, birthday, etc.) of the person is entered in the system
2. Identity certification: a mechanism should be in place that allows the certification of the link between the electronic identity and the physical identity. (For example usage of a copy of the identity card, a signature and a secret key that is sent to the address of the person, ...)

This needs to be done for employees working with the system and for customers of the system.

User Management The user management process takes care of the full lifetime of the user in the system.

This extra functionality introduces new security requirements on top of the security requirements of the (regular) functionality of the e-finance system. In particular, access restrictions apply to the creation and modification of user accounts. We refer to the access control policy for relevant rules (see section 3.3.2.2). Also, no information should be deleted out of the system because this information may be required for audit activities or future forensics research, etc. Some information can be archived after period T. We refer to the audit policy for relevant rules (see section 3.3.6).

3.3.1.2 Authentication in the system

Entity authentication As mentioned above, when a transaction between a user and the bank system takes place, both entities should identify each other. For usability reasons, we want to achieve a single-sign-on system. Whenever

a user wants to use the system, he should authenticate himself at the start of his session. This initial authentication process should be sufficient for the rest of the session. On the other hand, the bank system will also identify itself to the user so he is certain that he is communicating with the bank system. This requirement applies to all possible communication in the system between a (remote) client and the main bank system. These remote clients could be: customers using home banking software, customers at self-banking terminals, personnel at workstations in the branch offices and the stock market systems.

Data authentication The information involved in a transaction between a user and the bank system should be authenticated for its origin. Data origin authentication ensures that the information is provided by who claims to have done so.

3.3.1.3 Authentication policy

1. Before a person can become a customer or employee, he has to be identified by means of his e-ID.
2. For user registration purposes, personal information has to be signed by the future user using his e-ID.
3. Before a person can use the system he has to identify himself as a registered user at the start of his session. Single sign-on has to be supported in case of multiple, distributed services.
4. Only one parallel session at a time is allowed for a user.
5. Customers in the branch office initially need to identify themselves to the employee, by showing their e-ID. This is a initial check to verify the person's identity as soon as possible.
6. Each operation requested by the customer at the branch office needs to be securely linkable to the requesting customer.
7. Each operation requested by the personnel needs to be securely linkable to the requesting employee.
8. Each operation requested by a customer who accesses the system directly (without intervention of an employee) needs to be securely linkable to the following entities:
 - (a) The requesting customer
 - (b) The trusted (front-end) system used by the customer to connect to the banking system (E. g. an ATM or web banking server)

3.3.2 Access control

3.3.2.1 Basic principles

The access control policy obeys to several basic security principles:

Default deny all actions that are not explicitly allowed by the policy should be denied. As a result, a person will never get access to a service 'by accident'.

Separation of duty its primary objective is the prevention of fraud and errors. This objective is achieved by disseminating the tasks and associated privileges for a specific business process among multiple users. For example, creating an account should be confirmed by the bank clerk and the customer.

The principle of least privilege a user should have only the authority he needs to accomplish his task. For example, the system administrator should not be given access to the account information of customers.

Further, the policy is based on the fact that information within the bank system cannot be accessed directly without using the bank system software services (see also the confidentiality policy).

3.3.2.2 Policy

1. (All actions should be logged.)
2. All users can inspect the stock market catalogue.
3. Bank clerks and office managers can inspect customer information (personal, as well as financial) and can inspect all stock market information. The personal information of a customer includes name, social security number, address, and birth date. The financial information of a customer includes account numbers, the balances, the transactions, market orders, the realization depot, and all other information of the financial products of the customer, offered by the bank.
4. Upon authorization by a particular customer, a bank clerk or an office manager from the customer's home branch office can provide the following services to this customer: basic and custody administrative operations (create customer, edit customer's personal information, open/close account, open/close position), custody financial transactions (transfer) and stock transactions (orders and options).¹ Basic financial transactions (deposit/withdraw/transfer) can be performed by clerks and managers from all offices.
Authorization by the customer is achieved by digitally signing the transaction request (e.g., by bank card or e-ID card for opening an account).
5. The office manager of the customer's home office can (un)block an account of this particular customer.

¹We expect the rule of home branch office to be easily evolvable.

6. A customer can inspect his personal financial information (basic and custody accounts, orders, depot, ...), and he can modify his own personal information (i.e., address). He can also perform financial transactions on his own accounts, depot and stock market (in his own name).
7. A clerk cannot access his own accounts as a clerk, only as a customer.
8. A withdraw or transfer cannot exceed the credit limit.
9. An account cannot be closed with a negative balance.
10. High risk cash withdrawals can only take place at a branch office and have to be approved by the office manager. This includes a (set of) transaction(s) that exceeds 80% of the credit limit in 48 hours. Additionally, a regular withdrawal transaction cannot exceed a fixed amount (now set to 500 EUR).²
11. User account managers can manage (i.e., add or remove) users in the system. The management of user rights is performed by a security manager. Both roles cannot be assigned to a single person (static SoD).
12. The stock market can give feedback about the orders which are handled by himself.
13. Auditors can inspect within the system all information but the customer's personal information.
14. All operations not described by this policy are not permitted.

²Instead of red flagging.

3.3.3 Privacy and confidentiality

Information should be kept secret from all but those who are authorized to see it. This is partially covered in the requirement about access control. However, also direct access to the clear data on storage (databases, logs, backups, ...) by circumventing the software layer should not be possible. This is particularly true for all data marked sensitive in the domain model, and all data that is related to the security services.

This requirement also includes that the information involved in a transaction between the user and the bank should be kept secret while it is on the communication channel. As a matter of fact, all information can be transported over a public untrusted channel: home banking can use the Internet, branch offices can be connected through a leased line, stock markets also could use the Internet. Self-banking terminals can use the public telephone network.

Consequently, the policy about confidentiality states that :

1. No direct access to data on the storage should be allowed. All data should only be accessible through the software system.
2. All information transported over a untrusted network should be kept confidential to the parties not involved in the transaction.
3. Confidentiality of data transported over an untrusted network is enforced by cryptographic means of which the strength should be sufficient to cover the financial risk of the transaction.
4. All sensitive cryptographic data, such as private and secret keys, should be stored in a secure way.

3.3.4 Data integrity

Data integrity addresses the unauthorized alteration of data. To assure data integrity, one must have the ability to detect data manipulation by unauthorized parties. This data manipulation includes among others insertion, deletion, and substitution.

Although access control also protects against data manipulation, it is mainly applied at the service level. Data integrity however, also includes the ability to detect manipulation without using the system services, but by directly attacking the data in transit or in storage. Hence, data integrity applies to two kinds of data: the information involved in communication between a user and the bank system and the data in the persistent storage of the bank system.

Policy

1. Appropriate protection measures should support both information alteration by hardware failure or by manipulation through an attack.
2. It must be possible to check all information on the storage device for its integrity.
3. All corrupted information on storage should be recoverable.
4. All information received over a remote channel, should be checkable on integrity.

3.3.5 Non-repudiation and accountability

Accountability is the requirement that a certain operation or event cannot be denied by one or more of the parties that are involved. Accountability ensures that a person can be identified for his actions. Non-repudiation also delivers cryptographic proof of those actions. The legal value of cryptographic proof depends on many extra conditions such as timestamping, secure storage and audits of the cryptographic services. We will discuss this further when elaborating on the security architecture.

In general, accountability with non-repudiation applies for all transactions that modify data and that could lead to a dispute between the involved parties. For all these transactions it must be possible to provide proof about the involvement of the different parties so they can be held accountable for it. Within the banking system this requirement applies to multiple scenarios and involves different parties that have to commit undeniably.

The remainder of this section discusses the different classes of transactions that require accountability and non-repudiation, and what kind of proof is needed. Afterwards, we will elaborate on the trust model that applies to the e-finance case and finally, we will specify the concrete policy.

3.3.5.1 Classification and illustration

Within an e-finance system, we can distinguish two kinds of transactions: two-party transactions, and multi-party transactions.

Two-party transactions These are requests of the customers for a certain service of the bank, that only involves the customer and the bank. Both the customer and the bank needs to confirm (undeniably) the request for this service. The customer needs to confirm that he requested the service (non-repudiation of origin) and the bank needs to confirm it received this request (non-repudiation of receipt).³ Typical scenario's where this applies include:

- The customer requests the creation of a current, savings or custody account.
- The customer requests for a financial transaction, such as a transfer between his own account and another account at that bank, a withdrawal of his own account or a deposit on his own account.

Remark that non-repudiation of origin of the depositing person was formerly not needed for a deposit on an account, because denying a deposit on an account didn't give him any advantage. But, with regards to the new regulations on money laundry, deposits can only be done by registered customers of a bank (see access control policy), and hence non-repudiation of origin is required for every transaction.

³Remark that the terms non-repudiation of origin and receipt are typically used for proof on communication. In the context of this case, we use them for proving that a request took place. Whether this request actually maps one-to-one with a particular communication exchange is another technical discussion that we will postpone to the actual design of the e-finance system.

Multi-party transactions These are requests of the customers for a certain service that involves another financial institution (like another bank or a stock market).

As for two-party transactions, the customer needs to confirm that he requested the service (non-repudiation of origin) and the bank needs to confirm it received this request (non-repudiation of receipt). The same applies to the transaction between the bank and the other financial institution. We illustrate this with two examples.

In case of a money transfer the other financial institution does not need proof concerning non-repudiation of origin of the customer. But the customer needs proof that the bank indeed has transferred the money to the other bank. Therefore the bank will add the proof about non-repudiation of receipt to the customer's transactions.

In case of a stock order the bank also needs to proof to the stock market that the order is as the customer requested it.⁴ Therefore, proof concerning non-repudiation of origin from the customer towards the bank needs to be included when sending the order to the market.

In summary, there has to be non-repudiation of origin from the customer towards the bank and from the bank towards the other financial institution. There has to be non-repudiation of receipt from the bank towards the customer and from the other financial institution towards the bank.

3.3.5.2 Trust Model

We discuss the different trusted third parties that will be involved for fair non-repudiation. Then, we elaborate on the trust model, and in particular the differences between online electronic transactions and manual transactions at a branch office of the bank. Finally, the chain of evidence in the different transactions is discussed.

Trusted third parties Three trusted third parties are important for the non-repudiation requirements of the e-finance application: certificate authorities of the financial institutions, the government as certificate authority for the e-ID card and a fair non-repudiation service.

Certificate authorities Certificate authorities guarantee that the bank is who it claims to be. The set of root certificate authorities, that issue the certificates of the different financial institutions, needs to be trusted by all participants in the transactions. Remark that self-signed certificates are a problem for electronic banking. A customer can never be sure, that the banking service he is communicating with, is the bank that it claims to be. The channel can never be properly authenticated, and therefore the channel is vulnerable for man-in-the-middle attacks. The proof of origin or receipt received from a bank, is based on digital signatures and suffers the same problem due to the self-signed certificate. Cross-signed root certificates can help in case the certificate authorities of different parties are not identical.

⁴Stock brokers are not allowed to group orders or alter them.

Non-repudiation service To achieve fair mutual non-repudiation, a trusted third party is required. This trusted third party could be an online service (that is always involved in the transaction) or an offline service (that is only involved when a dispute arises). We refer to [13] for more information on this matter.

Government The government, acting as a certificate agency for the e-ID card, guarantees that the identity of a customer creating a signature can be securely verified. The government could install a proper root certificate authority, or make use of another root certificate authority to issue its own certificates.

The cryptographic requirements for digital signature should conform to the relevant standards.

Trust model In case of an *electronic banking environment*, we assume that both parties in a transaction, bank to customer or bank to bank, do not trust each other (but that an authenticated channel between the two parties can be established using a trusted third party). We assume that the bank and the customer trust the government as the issuer of the e-ID and that both also trust a well-known certificate authority that issued the bank's certificate.

Because the two parties do not trust each other, another trusted third party will be needed to establish a fair non-repudiable transaction between the two parties. This trusted third party is needed to function as a delivery office for the signatures of transaction requests.

In case of a *manual transaction at the branch office*, the same requirements apply as for the electronic banking. But, some extra precautions need to be taken to secure the transactions that involve cash money (e.g., withdrawals and deposits). It is always possible that one of the parties, bank clerk or customer, denies that it received the cash money from the other party. We assume that office managers, surveillance cameras recording on video or witnesses can bring resolution to this kind of conflicts between customer and employee. The history of customer or employee can also contribute to resolving a conflict. The profit that could be achieved by the customer or employee through fraud in a single transaction does not compare to the consequences. In case of the employee, he can lose his job and in case of the customer he can be blacklisted at the national bank.

Chain of evidence We explain three important situations where a chain of evidence to a trusted third party is crucial: certificates of the transaction participants, proof of delivery and proof of customer initiation.

Certificates of the transaction participants The signatures of the different parties in a transaction need to be verified by means of a certificate, that uniquely links an identity with the verification data (e.g., a public key). This certificate needs to be issued by an internationally recognized root certificate authority, or by a registration authority that is granted this function by the root certificate authority. Digital signatures of the certificates form a chain of evidence from the certificate of one of the transaction participants to this root certificate authority (see PKI based on X.509 certificates for more information).

The certificates of the root certificate authorities need to be present on each of the systems that is involved in the transaction. These certificates must be gathered through a non-remote channel and from a trusted party. For example, read-only digital media from a trusted software vendor or government office. It is obvious that this approach has scalability issues in case a lot of root certificate authorities are involved. Therefore, another possibility is to use cross-signing of the certificates of the root certificate authorities. Then one root certificate on a system is sufficient to disseminate to others in a secure way through a remote channel.

Proof of delivery A proof of delivery is provided by a bank towards a customer, when another financial institution is involved. It ensures that the bank indeed involved the other financial institution for the transaction requested by the customer. This proof of delivery is the proof of receipt delivered by the other financial institution to the bank of the customer. That proof of receipt is the signature of the other financial institution and this signature should be verifiable for the customer, similar to the situation described in the previous paragraph.

Proof of customer initiation The bank needs to prove to the market that the customer indeed has requested the transaction at it is sent by the bank to the market. Therefore, the bank will include the proof of origin of the customer, created when the customer requested the transaction. Again, this proof of receipt is verifiable by the market, because the e-ID of the customer contains a certificate that is issued by the government. We assume that the government acts as a certificate agency.

3.3.5.3 Non-repudiation policy

The first set of rules is about the cryptographic technology used for achieving non-repudiation proof, the second set defines the required proof for transactions between customer and bank. The third set of rules defines the required proof for transactions between the bank and another financial institution. We conclude with the rules about fairness of proof delivery.

1. cryptographic services

- (a) Proof of origin is achieved by a digital signature of the (time stamped) transaction, placed by the person who originates a transaction:
 - i. In case of the customer : by means of his e-ID, issued by the government
 - ii. In case of a financial institution: by means of a certificate issued by an internationally approved certificate authority.
- (b) Proof of receipt is achieved by a digital signature of the transaction content, placed by the person who receives a transaction.
 - i. In case of the customer : by means of his e-ID, issued by the government
 - ii. In case of a financial institution: by means of a certificate issued by an internationally approved certificate authority.

- (c) The certificate of a financial institution has a chain of evidence to a root certificate agency, whose certificate has been installed on the different systems through a non remote channel.
- (d) The non-repudiation proof needs to be timestamped.
- (e) The non-repudiation proof needs to be revalidated when new minimal cryptographic requirements for digital signatures are defined.
- (f) The non-repudiation proof needs to be stored in a secure way.

2. Transactions between bank and customer

- (a) Customers need to provide a proof of origin for each requested transaction
 - i. A transaction on a current account or savings account: open, close, withdraw, deposit or transfer
 - ii. A stock market transaction: direct orders, limited orders, or option execution
- (b) A bank needs to provide a proof of receipt to the customer for each transaction requested by this customer.

3. Transactions between bank and another financial institution

- (a) A bank needs to provide a proof of origin and receive a proof of receipt for transactions with other financial institutions
 - i. when transferring money to another bank.
 - ii. when submitting stock transactions to the market.
- (b) A bank needs to provide to the market the customer's proof of origin about stock transactions.
- (c) A bank needs to receive a proof of origin and provide a proof of receipt when receiving a transaction from other financial institutions
 - i. A money transfer
 - ii. A confirmation of a selling order
 - iii. A confirmation of a buying order
 - iv. An order denial
 - v. An invoice for a trading transaction

4. Fairness in the delivery of non-repudiation proof

- (a) *(Un)fair delivery for (manual) branch office transactions* a bank only provides a proof of receipt to customers, after the customer provided the proof of origin.⁵
- (b) *Fair delivery for online electronic transactions* proof of origin and receipt need to be delivered in a fair way to both parties, monitored by a internationally approved trusted third party (proof delivery service).

⁵This can be considered as an unfair situation, but it relates to the credibility of the bank: an employee that does not provide the proof of receipt would be sanctioned by the office manager.

3.3.6 Audit

Two important audit requirements at the business level drive the audit requirements at the technical level:

- a financial institution wants compliance with a set of controls defined by an external organization, to achieve the certification of its system. Therefore, a lot of information about the operations of the employees and the customer transactions needs to be stored.
- by legislation and government regulations, a financial institution is obligated to keep track of the transactions of the customers for at least five years. The institution also needs to keep track of all identification information of the customer.

The audit service of the e-finance application does depend on several other security services in order to keep track of a valid audit trail. But it is also important that this stored audit information does not compromise the other security requirements (e.g., privacy). We summarize these related security requirements in a separate security policy for the audit service.

Audit policy

1. All operations of the employees should be stored, with all involved information.
2. All transactions (i.e., operations that modify data) of the customers should be stored, with all involved information.
3. All non-repudiation information about transactions (proof of origin and proof of receipt) should be stored with the transaction information).

Audit service: security policy

1. Only authenticated employees of the internal audit group can use the audit service and only to inspect the data in the audit trail.
2. Adding entries to the audit trail can only be achieved using the audit service interface.
3. Only the e-finance software services can add entries to the audit trail.
4. All rules of the data integrity policy also apply to audit information.
5. All entries in the audit trail must be securely timestamped. This timestamp has to be verifiable by certification organizations (for audit purposes), by the government (for legal purposes) and by the customers (in case of a dispute with the bank).
6. Audit entries must be revalidated by the bank when new cryptographic standards are required.

3.3.7 Recovery

Two kinds of recovery requirements apply to e-finance IT systems: recovery after downtime of systems and recovery after data loss. The recovery requirements are part of the availability requirements. Quality scenarios in Chapter 4 explain the requirements about high availability of the IT systems. These scenarios also describe data recovery in case of data loss due to hardware malfunctioning, an attack or a disaster.

Chapter 4

Other non-functional requirements

First we will give a short introduction about business continuity. An important requirement within business continuity is availability. In this introduction we will discuss some business level issues about availability to illustrate the importance.

In the second section we will elaborate on more technical non functional requirements. We will use quality attribute scenarios to characterize the desired quality attributes of the system.

4.1 Business Continuity

A bank has typically the objective of developing a business continuity plan. This plan is developed in different phases starting with a Business Impact Assessment. We will show some example scenarios.

Availability

- What if Headquarters or a branch office is lost?
 - Can the balance of an accounts being checked before transaction are allowed?
 - Is the repository for the user id's and access rights replicated in the different branch offices
 - Can the branch office operate independently of Headquarters? For how long? For what functionality, e.g. is withdrawal of money still possible?
 - Can the customers' identity being verified in other offices (signature, copy of identity card, ...)
- What if the external processing facility (inter banking organization) is lost?
 - Are there local processing facilities for transactions in case the link to the inter-banking organization is broken

- Is it possible to support stock trading? How are customers informed in case they have correctly signed a transaction for stock trading but at the moment the transaction is in process the system fails? Since the stock market changes rapidly, it is not possible to delay the transaction with 1 hour for example.
- What if critical systems or IT facilities are broken (e.g. database systems, network connection)?
 - Technical point of view: transactional integrity . . .
 - Procedural point of view: what secrets, access rights, etc. must be available for recovery from failure.
 - In case a backup solution is in place, it needs to be checked for new security issues.
 - This should be analyzed as soon as the solution architecture is defined in terms of software, and of deployment.

A Business Impact Assessment goes further than a disaster recovery plan, which typically is focused on IT infrastructure. For example: the systems running the e-finance application (i.e. hardware and software) should be available for 99.99%. This can be further defined in terms of planned versus non-planned downtime, number of maintenance windows, differences between front-end and back-end, etc. The system is defined as available if it responds in *well defined* response times. These response times can be different per activity, etc

Non compliance with the regulations and legislations The license for performing banking activities can be revoked in case not all procedures are in place that are required by the regulations and legislations that apply. These regulations and legislations are summarized in the introduction of the chapter about security requirements.

4.2 Quality attributes of the IT infrastructure

We will use quality scenarios to define the non-functional requirements at the technical level. These are very specific technical requirements for the IT infrastructure of the e-finance case study.

4.2.1 Availability

4.2.1.1 General quality scenarios

We can divide the general availability scenarios in two parts : availability of systems and availability of data.

Availability of systems

1. **Source:** Internal.
Stimulus: Zero downtime of all banking services.
Artifact: Bank system
Environment: Runtime.

Response: Downtime shall be avoided. The bank system is deployed on multiple servers with passive redundancy. The usage of transactional operations and stateless components between transactions grants consistency between redundant servers.

Response measure: virtual zero downtime.

2. **Source:** External.

Stimulus: Failure of external systems.

Artifact: External Communication

Environment: Runtime.

Response: Pre-emptive caching of important external data and gradual degradation of banking services relative to external systems failure.

Response measure: virtual zero downtime with degraded services.

Availability of data Availability of data to a software service can be threatened by three causes : hardware failure or attacks hardware failures : hard drive cannot deliver requested data or hardware delivers corrupted data attacks : data is corrupt due to an attack

4.2.1.2 Specific quality scenarios

1. **Source:** Internal.

Stimulus: Bank system application server crash.

Artifact: Bank system application server.

Environment: runtime.

Response: System administrator is informed to investigate cause. The redundant application server takes over. Transactional execution of operations ensures consistency in the data.

Response measure: virtual zero downtime.

2. **Source:** Internal.

Stimulus: Data server unreachable for application server

Artifact: Bank system data server.

Environment: runtime.

Response: The system administrator is informed to investigate cause. The redundant data server takes over. Transactional execution of operations ensures consistency in the data.

Response measure: virtual zero downtime.

3. **Source:** External.

Stimulus: Some stock market is unreachable

Artifact: Investment services.

Environment: runtime.

Response: The system administrator is informed to investigate cause. The bank system goes to degraded mode on investment services based on the unreachable stock market. Inspection of local data is possible. Cached information on stocks is clearly timestamped. Order requests can be made. Sending the orders to the market will be post-poned until the market is reachable.

Response measure:

4.2.2 Performance

This requirement is taken in account because it interferes with the security requirements. It will be worked out later.

4.2.3 Usability

An important usability requirement related to security is auto-adaptation of the user interface according to the possible actions allowed for the user. This software architecture document focuses on the functional (business) layer of software for retail banking services. The usability requirement requirement is about the presentation layer, so we do not design this in detail. Though, it is an interesting problem and we will shortly sketch a possible strategy to achieve this requirement.

4.2.4 Security

Security requirements are discussed in a separate chapter after this one.

Chapter 5

Conclusion

In this document, which is a result of the SoBeNeT project, we have described the requirements and analysis of a case study in the world of e-finance, including basic banking services and more advanced retail banking services concerning private investments.

The functional requirements have been described using actors, use cases and a domain model. For the non-functional requirements we have mainly focused on security. After an introduction to relevant regulations and applicable legislation, the security analysis has elaborated on extra actors introduced for the purpose of security, misuse cases and data sensitivity, to finally arrive to the actual security policy for the most important security requirements (identification and authentication, access control, confidentiality and integrity, accountability and non-repudiation, and audit). Other non-functional requirements have been (superficially) included as well, among others availability and performance.

In a follow-up document, the software architecture of this case will be further described.

Bibliography

- [1] Banking, Finance and Insurance Commission, Koninklijk besluit tot goedkeuring van het reglement van de Commissie voor het Bank-, Financien en Assurantiewezen betreffende de voorkoming van het witwassen van geld en de financiering van terrorisme, 8 October 2004 (In Dutch or French) http://www.cbfa.be/nl/ki/wg/pdf/rd_08-10-2004.pdf
- [2] BASEL II, Basel Committee on Banking Supervision, Bank for international settlements, <http://www.bis.org/publ/bcbsca.htm>
- [3] Wet tot bescherming van de persoonlijke levenssfeer ten opzichte van de verwerking van persoonsgegevens. 8 december 1992 met raadpleging van vroegere versies vanaf 18-03-1993 en tekstbijwerking tot 07-08-2003 <http://www.juridat.be>
- [4] Directives and regulations of the European Community:
 - Directive 2002/58/EC of the European Parliament and of the Council of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector (*Directive on privacy and electronic communications*)
 - Directive 2002/22/EC of the European Parliament and of the Council of 7 March 2002 on universal service and users rights relating to electronic communications networks and services (*Universal Service Directive*)
 - Directive 2002/21/EC of the European Parliament and of the Council of 7 March 2002 on a common regulatory framework for electronic communications networks and services (*Framework Directive*)
 - Regulation No 45/2001 of the European Parliament and of the Council of 18 December 2000 on the protection of individuals with regard to the processing of personal data by the Community institutions and bodies and on the free movements of such data.
 - Directive 2000/31/EC of the European Parliament and of the Council of 8 June 2000 on certain legal aspects of information society services, in particular electronic commerce, in the Internal Market (*Directive on electronic commerce*)
 - Directive 1997/66/EC of the European Parliament and of the Council of 15 December 1997 concerning the processing of personal data and the protection of privacy in the electronic communications sector (replaced by Directive 2002/58/EC)

- Directive 1995/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movements of such data. All directives are published in the Official Journal of the European Communities.
- [5] Methodology for Assessing Compliance with the FATF 40 Recommendations and the FATF 8 Special Recommendations, Financial Action Task Force on Money Laundering, 27 February 2004, <http://www.fatf-gafi.org>
- [6] DUMORTIER, J., 'Legal Status of Qualified Electronic Signatures in Europe', in ISSE 2004-Securing Electronic Business Processes, S. Paulus, N. Pohlmann, H. Reimer, uit. Vieweg, 2004, p. 281-289 "Het recht rond elektronische handtekeningen: Richtlijn 1999/93/EG en de omzetting in België en Nederland", Deventer, Kluwer, 2005, 155 blz.;
- [7] Wet tot invoering van het gebruik van telecommunicatiemiddelen en van de elektronische handtekening in de gerechtelijke en de buitengerechtelijke procedure. 20 oktober 2000, Belgisch Staatsblad van 22/12/2000. <http://www.juridat.be>
- [8] Wet houdende vaststelling van bepaalde regels in verband met het juridisch kader voor elektronische handtekeningen en certificatediensten. 9 juli 2001, Belgisch Staatsblad van 29/09/2001. <http://www.juridat.be>
- [9] Koninklijk besluit houdende organisatie van de controle en de accreditatie van de certificatedienstverleners die gekwalificeerde certificaten afleveren. 6 december 2002, Belgisch Staatsblad van 17/01/2003. <http://www.juridat.be>
- [10] Directive 1999/93/EG of the European Parliament and of the Council of 13 December 1999 on a Community framework for electronic signatures.
- [11] Guttorm Sindre, Andreas L. Opdahl. Templates for Misuse case Description. In Proceedings of the seventh international workshop on Requirements Engineering: Foundation for Software Quality, June 4-5 2001, Interlaken, Switzerland
- [12] Guttorm Sindre, Andreas L. Opdahl. Capturing Security Requirements through Misuse Cases. In Proceedings of the Norsk Informatikkonferanse NIK, November 26-28 2001, Tromsø, Norway
- [13] Steve Kremer, Olivier Markowitch, and Jianying Zhou. An intensive survey of non-repudiation protocols. *Computer Communications*, 25(17):1606-1621, November 2002.