Bigbank AS

Vadim Kööp

**System Analyst homework**

Tallinn 2025

**Table of contents**

[1 Case 1 6](#_Toc191633916)

[1.1 Introduction 6](#_Toc191633917)

[1.2 Tasks 6](#_Toc191633918)

[2 Theoretical Background Analysis of Case 1 7](#_Toc191633919)

[2.1 Current Debt Repayment Process ("As-Is") 7](#_Toc191633920)

[2.1.1 Overview of the Debt Repayment Mechanism 7](#_Toc191633921)

[2.1.2 Loan Repayment Cycle 7](#_Toc191633922)

[2.1.3 Priority Rules for Debt Repayment 7](#_Toc191633923)

[2.1.4 Potential Limitations in the Current Process 8](#_Toc191633924)

[2.2 Process Diagram "As-Is Process Diagram of Automated Loan Repayment System" 9](#_Toc191633925)

[2.2.1 Overview of the Process Diagram 10](#_Toc191633926)

[2.2.2 Description of Key Process Steps 10](#_Toc191633927)

[2.2.3 Diagram Representation 11](#_Toc191633928)

[2.2.4 Explanation of Symbols Used in the Diagram 11](#_Toc191633929)

[2.2.5 Observations and Possible Issues Identified in the Diagram 11](#_Toc191633930)

[2.3 Assumptions and Сonsiderations 12](#_Toc191633931)

[2.3.1 Introduction 12](#_Toc191633932)

[2.3.2 Assumptions Made 12](#_Toc191633933)

[2.3.3 Considerations for Further Clarification 14](#_Toc191633934)

[2.3.4 Considerations for Further Clarification 14](#_Toc191633935)

[2.4 Questions for process clarification 15](#_Toc191633936)

[2.4.1 Introduction 15](#_Toc191633937)

[2.4.2 Loan Repayment Process 15](#_Toc191633938)

[2.4.3 Account and Fund Availability 15](#_Toc191633939)

[2.4.4 System Execution and Scheduling 15](#_Toc191633940)

[2.4.5 Customer Interaction and Notifications 15](#_Toc191633941)

[2.4.6 Exception Handling and Business Rules 15](#_Toc191633942)

[2.5 Analysis of Inefficiencies and Issues 16](#_Toc191633943)

[2.5.1 Introduction 16](#_Toc191633944)

[2.5.2 Identified Inefficiencies 16](#_Toc191633945)

[2.5.3 Business Risks and Consequences 18](#_Toc191633946)

[2.5.4 Conclusion 18](#_Toc191633947)

[2.6 Proposed Improvements and Solutions 19](#_Toc191633948)

[2.6.1 Introduction 19](#_Toc191633949)

[2.6.2 Key Improvements 19](#_Toc191633950)

[2.6.3 Summary of Proposed Solutions and Benefits 21](#_Toc191633951)

[2.7 Conclusion 22](#_Toc191633952)

[2.7.1 Summary of Findings 22](#_Toc191633953)

[2.7.2 Proposed Solutions 22](#_Toc191633954)

[2.7.3 Expected Benefits 22](#_Toc191633955)

[2.7.4 Final Thoughts 23](#_Toc191633956)

[3 Case 2 24](#_Toc191633957)

[3.1 Introduction 24](#_Toc191633958)

[3.2 Tasks 25](#_Toc191633959)

[4 Theoretical Background Analysis of Case 2 26](#_Toc191633960)

[4.1 Introduction 26](#_Toc191633961)

[4.2 Information Gathering 26](#_Toc191633962)

[4.3 Analysis of Potential Causes 26](#_Toc191633963)

[4.4 Hypothesis Selection and Solution Development 27](#_Toc191633964)

[4.5 Task for the Technical Team 28](#_Toc191633965)

[4.6 Response to the Stakeholders 28](#_Toc191633966)

[4.7 Conclusion 28](#_Toc191633967)

**List of figures**

Figure 1. As-Is Process Diagram of Automated Loan Repayment System. ..................................... 9

Figure 2. A repayment schedule type calculator ............................................................................. 24

Figure 3. Annuity payment formula. ............................................................................................... 27

**List of tables**

Table 1. Business risks and consequences. .................................................................................. 18

Table 2. Proposed Solutions and Benefits. ................................................................................... 21

# 1 Case 1

### 1.1 Introduction

There is a bank called SmallBank OÜ, which is providing loans and accounts for their customers.

Each loan contract is connected to an account from which loans are paid out to customers, and loan repayments to these accounts are matched to loan contracts to cover debts. Multiple contracts may use the same account.

Each loan contract in debt is checking its account for available funds every 15 minutes. If there are sufficient funds to cover the debt, a transaction is created from the account to the loan contract to cover the debt. If there are multiple loans in debt, oldest debt is covered first.

### 1.2 Tasks

• Create an as-is process diagram of the debt matching process.

• Feel free to make assumptions about parts not described, but make sure to document any assumptions made.

• Prepare questions to better understand the parts of the process which have not been sufficiently described.

• Based on limited information available, highlight the inefficient parts of the operation, and any business issues that may arise from these inefficiencies.

• Propose solutions to highlighted issues (the goal is not to describe a full solution, but to initiate a discussion on ways the process could be improved)

# 2 Theoretical Background Analysis of Case 1

## 2.1 Current Debt Repayment Process ("As-Is")

### 2.1.1 Overview of the Debt Repayment Mechanism

SmallBank OÜ provides loans to its customers, and each loan contract is linked to a specific account. This account is used for loan disbursement as well as for loan repayment. Customers deposit funds into their accounts, and these funds are then used to cover their outstanding loan debts.

The bank has an automated debt repayment process that checks the account balance at regular intervals (15 minutes) to determine if there are sufficient funds to cover outstanding debts. If funds are available, a repayment transaction is initiated (oldest debt is covered first).

### 2.1.2 Loan Repayment Cycle

The debt repayment process operates on a scheduled basis with the following key steps:

1. **Periodic Debt Check** **-** every 15 minutes, an automated system checks the linked accounts for available funds.
2. **Funds Availability Verification** - if sufficient funds are available, the system proceeds with repayment.
3. **Transaction Creation** **-** a repayment transaction is created from the account to the respective loan contract.
4. **Debt Settlement Priority** - if multiple loans are due, the system prioritizes older debts first.

This cycle repeats every 15 minutes to ensure outstanding debts are covered as soon as funds become available.

### 2.1.3 Priority Rules for Debt Repayment

If a customer has multiple outstanding loan contracts associated with the same account, the system follows a predefined priority rule:

1. **Oldest Debt First** **-** the system pays off the oldest debt before processing newer ones.
2. **Sequential Processing** **-** each transaction is processed sequentially to avoid partial payments across multiple loans.

This ensures that long-standing debts are cleared before newer obligations.

### 2.1.4 Potential Limitations in the Current Process

While the process is automated, there are inherent limitations, including:

1. **Fixed 15-minute Check Interval** **-** if a customer deposits funds immediately after a check, the system will not recognize them until the next cycle, delaying repayment.
2. **No Partial Payment Mechanism** **-** if the account balance is insufficient to cover a full repayment, no partial payment is made, potentially causing delays in debt reduction.
3. **Strict Sequential Processing** **-** prioritizing older debts may not always align with customer preferences or business goals.

These limitations may lead to inefficiencies in fund utilization and debt clearance timing.

## 2.2 Process Diagram "As-Is Process Diagram of Automated Loan Repayment System"

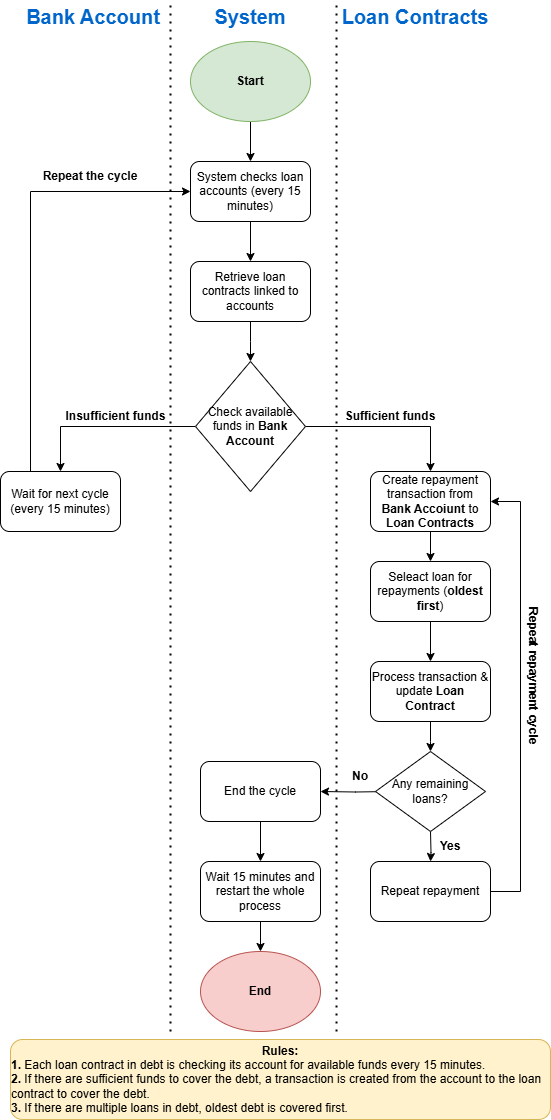


Figure 1. As-Is Process Diagram of Automated Loan Repayment System.

### ****2.2.1 Overview of the Process Diagram****

The "As-Is" process diagram visually represents the current debt repayment mechanism at SmallBank OÜ. It illustrates how loan accounts are monitored, how funds are allocated to cover debts, and the sequence of operations involved in the repayment process. This diagram helps identify inefficiencies and potential areas for improvement.

### ****2.2.2 Description of Key Process Steps****

The debt repayment process consists of the following key steps:

1. **Initiation of the Checking Process** 
   1. Every 15 minutes, the system triggers a process to check all loan accounts for available funds.
   2. The system retrieves a list of outstanding loan contracts associated with each account.
2. **Funds Verification**
   1. For each loan contract, the system checks whether the linked account has sufficient funds to cover the debt.
   2. If the available balance is greater than or equal to the outstanding amount, the system proceeds with the repayment transaction.
   3. If funds are insufficient, no repayment occurs, and the system waits until the next check.
3. **Debt Repayment Processing**
   1. If multiple loan contracts exist for the same account, the system prioritizes repayment based on the oldest outstanding debt.
   2. A transaction is created to transfer funds from the account to the corresponding loan contract.
4. **Transaction Execution**
   1. The system registers the repayment transaction and updates the loan balance accordingly.
   2. If the loan is fully repaid, it is marked as closed.
   3. If there are remaining funds in the account, the system repeats the process for the next outstanding loan.
5. **End of Cycle and Repetition**
   1. Once all eligible repayments have been processed, the system completes the current cycle.
   2. The process restarts after 15 minutes.

### ****2.2.3 Diagram Representation****

The following diagram (to be created) will visually represent the flow of the process:

1. **Start** → System triggers loan account check every 15 minutes
2. **Retrieve Loan Contracts** → Identify all outstanding loans for each account
3. **Check** **Available Funds**
   1. If **sufficient funds** → Proceed to repayment
   2. If **insufficient funds** → Wait for the next cycle
4. **Select** **Loan for Repayment (Oldest First)**
5. **Process** **Transaction & Update Loan Balance**
6. **Check** **for Remaining Loans**
   1. If **yes**, repeat from step 4
   2. If **no**, end the cycle
7. **Wait** **15 minutes and restart process**

### 2.**2.4 Explanation of Symbols Used in the Diagram**

**Oval (Start/End Nodes)** **-** represents the beginning and end of the process.

**Rectangle (Process Step)** **-** denotes an operation performed by the system.

**Square (Decision Point)** **-** represents a conditional check in the process.

**Arrows** **-** indicate the flow of the process from one step to another.

### 2.**2.5 Observations and Possible Issues Identified in the Diagram**

**Delay in Fund Utilization** **-** since the system operates in fixed 15-minute intervals, a deposit made just after a check is ignored until the next cycle.

**Inefficient Prioritization** **-** the system prioritizes the oldest debt first, which may not align with customer repayment preferences.

**No Partial Payments** **-** the system does not attempt to make a partial payment if the full amount is unavailable, leading to delayed debt reduction.

The diagram serves as a foundation for analyzing inefficiencies and proposing optimizations in the later sections.

## 2.3 Assumptions and Сonsiderations

### 2.3.1 Introduction

Since the provided case study lacks some details about the debt repayment process, certain assumptions have been made to construct a more complete understanding of how the system operates. These assumptions are necessary for identifying inefficiencies and proposing improvements. The following section outlines these assumptions and the reasoning behind them.

### 2.3.2 Assumptions Made

1. **Loan and Account Relationships**
   1. Each loan contract is linked to a bank account that is used for loan disbursement and repayment.
   2. Multiple loan contracts can be associated with the same bank account.
   3. The same account may be used for purposes other than loan repayments, such as receiving deposits and handling customer transactions.
   4. The loan contract does not have a predefined repayment schedule, meaning repayments depend on account balance availability rather than fixed installment dates.
2. **Debt Checking and Repayment Process**
   1. The system checks all loan accounts for available funds every 15 minutes without exception.
   2. If sufficient funds are available, a repayment transaction is triggered immediately during the check.
   3. Loan repayment transactions are processed sequentially, meaning that the system does not process multiple loans simultaneously.
   4. If an account has multiple outstanding loan contracts, the oldest loan is repaid first before newer debts are considered.
   5. If there are insufficient funds, no partial payments are made, and the system waits until the next cycle to check again.
3. **System Behavior and Constraints**
   1. The system does not notify customers about pending debts or missed payments, it only executes automatic checks.
   2. Loan contracts do not have a specific grace period for repayments, meaning overdue debts remain until the account balance is sufficient.
   3. There are no penalties or escalation processes mentioned in case of prolonged non-payment.
   4. The system does not attempt to reallocate funds from other accounts owned by the same customer to cover debts.
   5. The process is fully automated, meaning no manual intervention is required for repayment execution.

### 2.3.3 Considerations for Further Clarification

While these assumptions provide a basis for understanding the system, certain aspects remain unclear and require additional clarification from stakeholders:

1. **Loan Repayment Priority Rules**

Does the system strictly follow an "oldest debt first" approach, or do different loans have different repayment priorities based on interest rates or contract conditions?

1. **Handling of Partial Payments**

Can the system process partial payments when the account has insufficient funds, or does it always wait for a full repayment amount?

1. **Account Balance Management**

Does the system consider reserved funds for other transactions when determining available balance for loan repayment?

1. **Customer Notifications and Alerts**

Are customers notified when their debt is checked but remains unpaid due to insufficient funds?

1. **Exception Handling and Failure Scenarios**

What happens if a repayment transaction fails due to a system error or banking restrictions?

Is there a retry mechanism, or does the system wait for the next 15-minute cycle?

### 2.3.4 Considerations for Further Clarification

The assumptions outlined above directly impact the process analysis and efficiency evaluation. For example:

1. If partial payments are not allowed, debt clearance is delayed, potentially increasing outstanding loan amounts.
2. If the system only checks every 15 minutes, funds deposited immediately after a check remain unused until the next cycle.
3. If no notifications are sent, customers may remain unaware of their pending debts, leading to prolonged overdue payments.

These considerations highlight the need for additional clarification before proposing system optimizations.

## 2.4 Questions for process clarification

### 2.4.1 Introduction

To fully understand the debt repayment process at SmallBank OÜ, additional details are needed. The following questions aim to clarify missing aspects and ensure a complete analysis of the system.

### 2.4.2 Loan Repayment Process

1. How is the priority for debt repayment determined?
2. Does the system always prioritize the oldest debt, or are there other factors such as interest rates or due dates?
3. Can the system handle partial payments?
4. If an account has some funds but not enough to cover the full debt, can a partial repayment be made?

### 2.4.3 Account and Fund Availability

1. How does the system define "available funds"?
2. Does it consider only the current account balance, or does it account for pending transactions or reserved funds?
3. Are there any restrictions on fund usage for repayments?
4. Can a customer manually reserve funds in the account to prevent automatic repayments?

### 2.4.4 System Execution and Scheduling

1. Is the 15-minute debt check interval fixed, or can it be adjusted?
2. Would it be possible to trigger a repayment immediately when funds become available?
3. What happens if the system fails to process a repayment transaction?
4. Does it retry automatically, or does the system wait until the next 15-minute cycle?

### 2.4.5 Customer Interaction and Notifications

1. Are customers notified when a repayment fails due to insufficient funds?
2. If yes, what method is used (SMS, email, app notification)?
3. Can customers track real-time repayment status?
4. Is there a way for customers to see which loan is being repaid next and how much is due?

### 2.4.6 Exception Handling and Business Rules

1. What happens if a loan remains unpaid for an extended period?
2. Are there penalties, escalations, or manual interventions?
3. Can funds from other accounts owned by the same customer be used to cover the debt?
4. If a customer has multiple accounts, does the system check other balances for repayment?

## 2.5 Analysis of Inefficiencies and Issues

### 2.5.1 Introduction

The current debt repayment process at SmallBank OÜ is automated and follows a structured workflow. However, certain inefficiencies and potential business issues could impact operational effectiveness, customer satisfaction, and financial performance. This section identifies key inefficiencies and their consequences.

### 2.5.2 Identified Inefficiencies

1. **Fixed 15-Minute Check Interval**
   1. The system checks for available funds only every 15 minutes. If a customer deposits funds just after a check, repayment is delayed until the next cycle.
   2. This may lead to unnecessary delays in debt clearance, especially for customers who intend to pay off their loans immediately.

**Potential Impact:**

* 1. Customers may experience frustration due to delayed debt processing.
  2. Funds remain unused for up to 15 minutes, impacting cash flow optimization.
  3. If multiple transactions happen within this window, prioritization may become inefficient.

1. **No Partial Payments Allowed**
   1. If a customer has some but not all of the required funds for a repayment, the system does not process a partial payment.
   2. The debt remains unpaid until the full amount is available.

**Potential Impact:**

* 1. The outstanding debt does not decrease, even when partial funds are available.
  2. Customers may accumulate late fees or penalties due to rigid repayment conditions.
  3. The bank could miss opportunities to gradually recover debts.

1. **Sequential Loan Processing (Oldest First Rule)**
   1. The system strictly follows the "oldest debt first" rule, meaning newer debts are ignored until older ones are fully repaid.
   2. Some customers might prefer to pay off smaller debts first or prioritize high-interest loans.

**Potential Impact:**

* 1. Customers have no flexibility in how their payments are allocated.
  2. This may not be the most financially efficient strategy for either the bank or the customer.
  3. Loans with higher interest rates could remain unpaid longer, increasing financial risks.

1. **No Customer Notifications for Failed Payments**
   1. If a repayment fails due to insufficient funds, the system does not notify the customer.
   2. Customers may not be aware of missed payments, leading to unexpected penalties or loan escalations.

**Potential Impact:**

* 1. Customers may assume their debt was repaid, only to later discover overdue payments.
  2. Increased customer complaints and support requests.
  3. Higher likelihood of non-performing loans due to unaddressed failed transactions.

1. **No Exception Handling for Failed Transactions**
   1. If a repayment transaction fails due to technical issues (system errors, network problems, bank restrictions), the system waits until the next 15-minute cycle rather than retrying immediately.
   2. No automatic retry mechanism exists to ensure faster issue resolution.

**Potential Impact:**

* 1. Delays in processing lead to unnecessary backlogs of pending transaction.
  2. Possible financial losses if repayments are not processed on time due to system errors.
  3. Increased operational burden due to manual interventions required to fix issues.

1. **No Alternative Fund Sources Considered**
   1. The system only checks the assigned bank account for repayment funds.
   2. If a customer has other accounts with sufficient balance, those funds are not used to cover the debt.

**Potential Impact:**

* 1. Missed opportunity to prevent overdue payments.
  2. Increased risk of loan defaults if customers fail to manually transfer funds.

### 2.5.3 Business Risks and Consequences

|  |  |
| --- | --- |
| **Issue** | **Potential Business Impact** |
| Delayed repayments due to fixed check intervals | Customer dissatisfaction, delayed cash flow |
| No partial payments | Higher overdue balances, increased default risks |
| Strict "oldest debt first" rule | Inefficient financial planning for customers |
| No failed payment notifications | Higher risk of non-performing loans |
| No immediate retries for failed transactions | Slower debt recovery, operational inefficiencies |
| No alternative fund sources checked | Missed repayment opportunities, increased loan defaults |

Table 1. Business risks and consequences.

### 2.5.4 Conclusion

The current system, while structured, contains several inefficiencies that impact both customers and the bank. Addressing these issues through process automation, customer communication improvements, and repayment flexibility could enhance efficiency and reduce financial risks.

## 2.6 Proposed Improvements and Solutions

### 2.6.1 Introduction

To enhance the efficiency of the debt repayment process at SmallBank OÜ, several improvements can be made. The proposed solutions focus on reducing delays, increasing automation, enhancing customer experience, and optimizing repayment processing.

### 2.6.2 Key Improvements

1. **Dynamic Fund Checks Instead of Fixed 15-Minute Intervals**

**Current Issue:**

* 1. The system checks for available funds every 15 minutes, causing delays if a customer deposits funds just after a check.

**Proposed Solution:**

* 1. Implement an event-driven system where fund availability is checked immediately after a deposit is made.
  2. Combine this with scheduled checks to ensure repayments are processed as soon as possible.

**Expected Benefits:**

* 1. Faster debt repayment processing.
  2. Reduced waiting time for customers.
  3. More efficient cash flow management.

1. **Allow Partial Payments**

**Current Issue:**

1. If the account does not have enough funds for a full repayment, the system waits until the full amount is available rather than making a partial payment.

**Proposed Solution:**

1. Enable partial payments, where available funds are used to reduce debt even if they do not cover the full amount.
2. Adjust the loan contract balance accordingly and notify the customer about the remaining debt.

**Expected Benefits:**

1. Helps customers gradually reduce their debt.
2. Reduces overdue balances and potential penalties.
3. Improves loan recovery rates for the bank.
4. **Flexible Loan Repayment Prioritization**

**Current Issue:**

1. The system always repays the oldest debt first, even if the customer prefers to pay off a smaller or high-interest loan first.

**Proposed Solution:**

1. Allow customers to choose which loan to prioritize (oldest, highest interest rate, lowest balance).
2. Alternatively, implement an intelligent algorithm that optimizes repayments for both the bank and the customer.

**Expected Benefits:**

1. Increased customer control over loan repayment strategy.
2. Faster clearance of high-interest debts, reducing overall loan costs.
3. More efficient fund allocation.
4. **Automated Customer Notifications for Failed Payments**

**Current Issue:**

1. Customers do not receive notifications if a repayment fails due to insufficient funds.

**Proposed Solution:**

1. Implement real-time notifications via email, SMS, or mobile banking app.
2. Include a summary of the failed transaction and a suggested action (deposit required amount).

**Expected Benefits:**

1. Customers stay informed about their loan status.
2. Reduces overdue payments caused by lack of awareness.
3. Improves customer satisfaction and trust in the bank.
4. **Immediate Retry Mechanism for Failed Transactions**

**Current Issue:**

1. If a transaction fails due to technical errors or network issues, the system waits until the next 15-minute cycle instead of retrying immediately.

**Proposed Solution:**

1. Introduce an automatic retry mechanism that attempts repayment multiple times within the same cycle before delaying to the next interval.

**Expected Benefits:**

1. Higher success rate for transactions.
2. Reduced risk of unnecessary repayment delays.
3. Less manual intervention needed for failed payments.
4. **Use Alternative Account Balances for Repayment**

**Current Issue:**

1. The system only checks the primary loan account, even if the customer has funds in another linked account.

**Proposed Solution:**

1. Enable the system to check all linked accounts and use available balances for loan repayments.
2. Customers can set preferences for which accounts should be used first.

**Expected Benefits:**

1. Fewer missed payments due to lack of funds in a single account.
2. Increased repayment success rate.
3. Lower default risks.

### 2.6.3 Summary of Proposed Solutions and Benefits

|  |  |  |
| --- | --- | --- |
| **Proposed Improvement** | **Current Issue** | **Expected Benefits** |
| **Dynamic fund checks** | Fixed 15-minute interval causes delays | Faster processing, immediate repayments |
| **Allow partial payments** | System waits for full amount | Reduced overdue balances, gradual debt reduction |
| **Flexible repayment priorities** | Oldest debt is always paid first | More control for customers, better financial planning |
| **Automated notifications** | Customers are unaware of failed payments | Fewer missed payments, better communication |
| **Immediate retry mechanism** | Failed transactions wait until next cycle | Faster issue resolution, fewer delays |
| **Use alternative account balances** | Only one account is checked for funds | Higher repayment success, fewer loan defaults |

Table 2. Proposed Solutions and Benefits.

## 2.7 Conclusion

### 2.7.1 Summary of Findings

The current debt repayment process at SmallBank OÜ is structured and automated, ensuring that outstanding loan balances are checked every 15 minutes and repaid when sufficient funds are available. However, several inefficiencies have been identified that may cause delays, missed payments, and customer dissatisfaction. Key issues include:

1. Fixed 15-minute fund checks, which can delay repayments.
2. No partial payments, meaning debts remain unpaid even when partial funds are available.
3. Strict repayment prioritization (oldest debt first), limiting flexibility.
4. Lack of customer notifications for failed payments.
5. No immediate retry mechanism for failed transactions.
6. No use of alternative account balances for repayments.

These inefficiencies not only impact operational efficiency but also pose potential risks such as delayed cash flow, increased overdue debts, and higher customer complaints.

### 2.7.2 Proposed Solutions

To address these inefficiencies, several key improvements were proposed:

1. Implement dynamic fund checks to trigger repayments immediately when funds become available.
2. Enable partial payments to reduce outstanding debt gradually.
3. Introduce flexible repayment prioritization, allowing customers to choose which loans to repay first.
4. Automate customer notifications to inform them of failed transactions and necessary actions.
5. Develop an immediate retry mechanism for failed payments to prevent unnecessary delays.
6. Allow repayments from alternative account balances to increase the chances of successful transactions.

These solutions will increase efficiency, enhance customer experience, and improve overall debt recovery rates.

### 2.7.3 Expected Benefits

By implementing these improvements, SmallBank OÜ can expect:

1. Faster and more efficient debt repayment processing.
2. Reduced overdue debts and better loan management.
3. Higher customer satisfaction through increased transparency and flexibility.
4. Stronger financial performance with improved cash flow and lower default risks.
5. Less manual intervention, reducing operational workload and potential errors.

### 2.7.4 Final Thoughts

The proposed changes will transform the debt repayment system into a more dynamic, customer-friendly, and optimized process. By leveraging automation and flexibility, SmallBank OÜ can minimize inefficiencies, enhance customer trust, and ensure a more effective debt recovery strategy. Implementing these recommendations will position the bank for long-term success while improving both customer and business outcomes.

# 3 Case 2

### 3.1 Introduction

You receive a request from a stakeholder: “We are comparing our internal system loan monthly payment calculator with external calculators, and it seems our calculator is working incorrectly. External calculators give a much higher monthly payment!

In the internal calculator we use:

• Loan amount 59 900 €

• Interest rate marginal 3% + EURIBOR

• Term: 300 months = 20 years

• Loan purpose: xxx

• Payment date: 5

• Schedule type: annuity

The calculator gives a result of 366,69€.

External calculator gives 410.96€ (see attached picture) Please fix the internal calculator ASAP!”

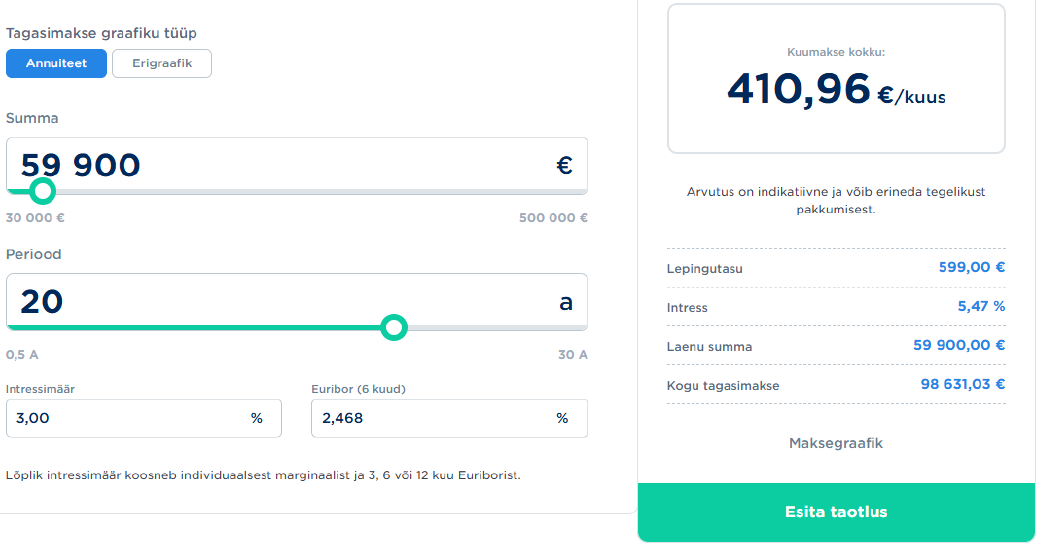


Figure 2. A repayment schedule type calculator.

### 3.2 Tasks

• Which information would you need, and from whom, to diagnose the problem further? You may address any question to any person within the organization (technical teams responsible for the internal calculators, business stakeholders using the calculators in their day-to-day work, legal team to find out legal requirements for the calculator etc), but not people outside the organization (responsible for the external calculator). List down specific questions with specific recipients.

• Based on limited information available, try to speculate what the potential root causes of the discrepancy may be. Choose one of those hypotheses and assume this is the root cause. How would you fix or improve the internal calculator to address this issue? Create an initial draft for a placeholder ticket to the technical team.

• After the previous steps are completed, compile a response to the stakeholder who raised the issue, outlining the next steps and action plan.

# 4 Theoretical Background Analysis of Case 2

### Introduction

The internal loan monthly payment calculator shows a result of **€366.69**, while the external calculator gives **€410.96**. The discrepancy must be analyzed, and a solution proposed.

### Information Gathering

**Required data and key questions by category:**

1. **Technical Team (developers of the internal calculator):**
   1. What formula is used for the payment calculation?
   2. How is the EURIBOR rate applied in the calculations?
   3. What rounding rules are implemented?
   4. How are data stored and processed?
2. **Business Analysts & Finance Department:**
   1. What are the official requirements for calculating monthly payments?
   2. Is there a methodological difference between the internal and external calculators?
   3. What key parameters are considered in the calculations?
3. **Legal Department:**
   1. Are there any regulatory requirements for loan payment calculations?
   2. Are there specific formulas mandated by financial authorities?

### Analysis of Potential Causes

**Possible reasons for the discrepancy:**

1. **Incorrect annuity payment formula**
   1. The internal calculator may be using an incorrect formula.
2. **Improper EURIBOR application**
   1. The floating interest rate might be miscalculated or derived from an incorrect data source.
3. **Different rounding methods**
   1. The internal calculator may round intermediate values differently than the external one.
4. **Incorrect time period usage** **a. 300 kuud ei ole 20 aastat**
   1. 300 months is not equivalent to 20 years, 300 months is equivalent to 25 years.
   2. The internal calculator might assume a different year length (360 days instead of 365/366).
5. **Incorrect coefficients or additional parameters**
   1. Hidden fees or incorrect coefficients may be applied in one of the calculators.

### Hypothesis Selection and Solution Development

We assume the "Incorrect annuity payment formula" hypothesis, as it is the most probable cause.

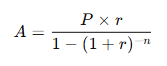


Figure 3. Annuity payment formula.

**Where:**

* A **-** Monthly payment
* P **-** Loan amount
* r **-** Monthly interest rate
* n **-** Number of payments

**Fix proposal:**

* Verify whether the formula is correctly implemented.
* Ensure that the monthly interest rate is calculated as (Annual Rate / 12).
* Conduct testing with different input scenarios.

### Task for the Technical Team

**Title:** Fix annuity payment calculation

**Description:**

A discrepancy has been identified between the internal and external loan payment calculators. The issue seems to be related to the incorrect application of the annuity formula. The development team needs to verify and correct the formula to align with financial standards.

**Steps:**

1. Review the current implementation of the annuity formula.
2. Fix the formula if inconsistencies are found.
3. Perform testing with various loan conditions.

**Priority:** High

### Response to the Stakeholders

**Subject:** Loan Payment Calculation Discrepancy – Status & Next Steps

**Message:**

Dear [Stakeholder],

We have analyzed the issue with the loan payment calculation and identified a potential cause: an incorrect implementation of the annuity payment formula.

**Next steps:**

1. The development team will verify and adjust the calculation formula.
2. Further testing will be conducted to ensure accuracy.
3. The fix is expected to be implemented soon.

We will keep you updated on progress.

Best regards,

Vadim Kööp

### Conclusion

This report comprehensively covers the issue, including information gathering, analysis of potential causes, problem resolution, and stakeholder communication.