

Assignment 3 - Process Mining

Course: 5,131: Introduction to Business Process Automation and Process Mining

Format: Group assignment, Group size: 1-2 people, Group set "A3-Groups" in Canvas

Weight: 35% of the grade

Deadline: January 6th, 2026, 19:00 CET

Background

Dhana is a European bank providing a range of financial services, including home and business banking, checking and savings account, debit and credit cards, investment management and loans.

The Loan Application Process

One of the core services provided by Dhana is their loan services, provided via their loan application process. This process supports different types of loans:

Courtesy of Dhana, we have obtained an event log containing all loan applications filed with the bank in 2016. The event log has been given to you separately. The event log contains about 761K events pertaining to 20.3K loan applications, handled by 146 bank employees. For these applications, a total of 26.8K offers were created. There are three types of events: Application state changes (starting with "A"), Offer state changes (starting with "O") and Workflow activity instances (starting with "W"). The activity instances have start and end timestamps. The events corresponding to state changes only have one timestamp.

For each loan application, the following data is available in the log:

- Requested loan amount (in Euros),
- The application type,
- The reason the loan was applied for (LoanGoal),
- An application ID.

For each offer, the following data is available:

- An offer ID,
- The offered amount,
- The first withdrawal amount,
- The number of payback terms,

- The monthly repayment cost,
- The credit score of the customer,
- The employee who created the offer (Resource),
- Whether the offer was selected,
- Whether the offer was accepted by the customer.

There may be multiple offers per loan application but at most one offer is accepted per loan application.

A loan application goes through a lifecycle consisting of the following states:

- **Application Created:** a new application has been created via the website.
- **Submitted:** a customer has submitted the application via the website. If the new application is created directly by the bank, this state is skipped.
- **Concept:** the application is in the concept state, that means that the customer just submitted it (or the bank started it), and a first assessment has been done automatically. An employee calls the customer to complete the application.
- **Accepted:** after the call with the customer, the application is completed and assessed again. If there is a possibility to make an offer, the state is accepted. The employee now creates one or more offers.
- **Complete:** the offers have been sent to the customer and the bank waits for the customer to return a signed offer along with the rest of the documents (payslip, ID etc)
- **Validating:** the offer and documents are received and checked.
- **Incomplete:** if documents are not correct or some documents are still missing, the application state is set to 'incomplete', which means the customers needs to send in further documents.
- **Pending:** if all documents are received and the assessment is positive, the loan is final and the customer is payed.
- **Denied:** if somewhere in the process the loan cannot be offered to the customer, because the application does not fit the acceptance criteria, the application is declined, which results in the state 'denied'.
- **Cancelled:** if the customer never sends in their documents or calls to tell they do not need the loan anymore, the application is cancelled.

Tasks

Write a report addressing each of the following tasks individually. Include for each task screenshots from the tools you used to document your evidence.

Important Notes

The log is extracted from a process automation system (a.k.a. Workflow system) similar to *Camunda*. You can see values for the attribute "*lifecycle:transition*" other than *complete*, e.g. *schedule* (the task becomes available to a process worker), *start* (the process worker has picked the task and starts working on it), *suspend*, *resume*, etc.

When you load the data in Disco or another tool, you see under "Activity" activity names such as "A_Create Application" as well as state names such as "A_Submitted". We will henceforth refer to those uniformly as *activities*.

In case you run into performance problems with the large log (e.g. with a Marimo notebook) or usage limitations (Celonis, Apromore), a second event log is provided that contains only the cases of the large log that complete before end of July. Please state in your report whenever you have used the smaller log to obtain a particular result.

Exploratory Analysis

1. Schemas:
 - a. Which data schema is associated with the activity "A_Create Application"?
 - b. Which attributes of this schema have missing values?
 - c. Does the log have any attribute that is re-used across different activities?
2. Inspect the case "Application_177461216". Describe what happens in that case in a few sentences.
3. What are the top three loan goals (with respect to their number of cases)?
4. Visualize the distribution of the requested amount per case, write down at least three interesting observations.
5. Visualize the relationship between the requested amount and the maximum offered amount per case. Write down at least two interesting observations.
6. Pick two other numeric attributes per case that you find interesting, visualize their relationship, and state some interesting observations.

Discovery

7. Sequential variants:
 - a. What is the number of sequential variants of the log?

- b. What is the minimum percentage of sequential variants that is needed to cover 50% of the cases? (see *Variation* filter in *Disco*)
 - c. Use an event filter by keeping only the *Loan Application* events, i.e., all events with an Activity starting with "A_". (We call the remaining subprocess also the *Loan Application lifecycle*.) Again, determine the number of sequential variants and the minimum percentage of variants to cover 50% of the cases.
8. Keep the event filter to visualize only the *Loan Application lifecycle* in the process map. Set all sliders to 100%. Study the process map and write down at least three interesting observations.
9. Repeat Task 8 for the *Offer lifecycle*, i.e., filter to keep all activities starting with "O_", but not "O_Create Offer".

KPIs and Outcomes

10. Create a visualization that shows the case outcome distribution of the log. Make sure that your set of outcomes is pairwise mutually exclusive and complete. Briefly reflect the result.
11. Suggest two charts for an operational dashboard for the investigated process. Describe briefly each chart and its configuration and why it could be useful for process operations. Make sure that each chart could be build with the data that is included in the event log, i.e., does not require additional events or attributes.

Conformance Checking

12. Check the following rules using *Disco* filters or *Declare* constraints. State, for each rule, the number of violating cases. Name the case id of a violating case (if one exists), inspect the case, and briefly interpret the violation in that case.
- a. When the loan application reaches the *pending* state, then an offer was previously accepted.
 - b. If a loan application is cancelled, then at least one associated offer is cancelled.
 - c. If a loan application is denied, then at least one offer is refused.
 - d. Whenever the application reaches the *incomplete* state, then a "*call incomplete*" workflow activity takes place.
 - e. Each *offer sent (mail only)* is followed by a *call after offers*.
13. Determine the percentage of cases that violate the following service level objectives (SLOs):
- a. If a positive decision is reached (*A_Pending*) on a loan application, then within 28 days.

- b. If a negative decision is reached (*A_Denied*) on a loan application with goal "existing loan takeover", then within 14 days.
14. Each individual offer has a lifecycle similar to a loan application.
- a. Given your preliminary knowledge of the process, draw a normative model of what you think the lifecycle of an offer should be. Use BPMN with start events, end events, activities and only exclusive gateways. A template that includes all activity names is attached. Save the model as *bpmn* file and include a screenshot in your report.
 - b. The *offer event log* has been extracted from the original event log and is provided to you separately. This event log shows every offer as a separate case (irrespective to which loan application it belongs) with the attribute "*OfferID*" being the Case Id. Perform a conformance check of this offer event log against the normative model you have drawn. State the number of violating cases (a.k.a. traces). You can use *Python* or *Celonis*. (Update: *Apromore* appeared to me unreliable recently. So I do not recommend it anymore. You can certainly hand in a solution done with *Apromore*.)
 - c. Pick one violating case, state its sequence of activities and interpret whether you consider that as a log error, i.e., a normative violation, or a model error, i.e., an incomplete normative model and why.

Performance Mining

15. Analyze the rework on the loan application lifecycle.
- a. Which activities are involved in rework?
 - b. What percentage of cases exhibit rework?
 - c. How does the median/mean cycle time of all successful (i.e., loan granted) cases compare to those of all successful cases with rework?
 - d. Suggest some potential improvements to the process based on the rework analysis.
16. Analyze the rework for the offer lifecycle.
- a. Which activities of the offer lifecycle (cf. Task 14b) has repetitions, if any?
 - b. What is the percentage of all offers that get accepted, canceled, and refused?
 - c. How many offers exist per loan application? Visualize or briefly describe the corresponding distribution.
17. Include '*/lifecycle:transition*' in the rework analysis (in the corresponding *Marimo* notebook) and now focus on *workflow activities* (= activities starting with "*W_*"). Write down your two most interesting observations.
18. Study case arrival, -closure, -load, and event load:
- a. What patterns or interesting observations regarding case arrival do you see?

- b. Write down your most interesting observation regarding case closure.
 - c. Write down our most interesting observation regarding case load.
 - d. Which activities are occasionally executed in batches? Interpret the batching for each such activity.
19. Cycle time
- a. Briefly describe the distribution of the cycle time (a.k.a. case duration) of the log.
 - b. What is a characteristic of many long-running cases?
 - c. Determine whether the cycle time for successful applications is dominated by the time spent inside the bank (waiting for a resource or completing work) or by waiting for the customer to return information.
 - d. Can you find any activity bottlenecks?
 - e. Check the distribution of the delay between the last contact the bank had or had attempted with the customer and the cancellation of an application. What does it tell you? Do you have any suggestions for improvement?
20. Determine for each of the following conditions, how many cases meet the condition. For each condition, inspect a case, mention (i) its case id, (ii) an idea for a process change that could increase ROI for that specific case, and (iii) a potential risk or trade-off associated with that process change.
- a. a denied loan application,
 - b. a cancelled loan application,
 - c. a case that includes a potential fraud investigation,
 - d. a successful application where multiple offers are created with at least 1 week in between.
 - e. a cancelled application where a *call after offers* is conducted (or attempted) by different loan officers (NB. the lifecycle transition "*ate_abort*" indicates that the task was automatically terminated by the system, i.e., such an event is not done by a loan officer.)

Submission Instructions

Submit a report in *pdf* format. Name the file *Name1_Name2.pdf* Include a screenshot for each of your results. If not clear from the screenshot, document the parameters that you have used to obtain your result to allow me to reproduce your result. Submit your report on *Canvas* by the deadline mentioned above.