



UNIVERSITÀ DI PISA

Project Report

Business Process Modelling

Data Science and Business Informatics

Elisa Pashku **(628389)**

Matteo Garro **(620025)**

Academic Year 2021/2022

1 Introduction

The scenario presented in this report consists of the design of probable processes in an **online music store** and analyzing these processes to validate them and verify the performance of the model created. The process modeling will be presented using the *BPMN language* for providing a simplified dynamic of the whole process, and *Petri Nets* for the soundness analysis.

1.1 Scenario - Project 41

Before starting the analysis, it is important to identify all actors involved in the process.

- *The customer*
- *The music store* (online system)
- *The bank*

Each actor interacts directly with the others, therefore, message flow arrows are employed to represent the flow of information between them. Artifacts are instead used for visually representing objects outside the main process (database artifacts in online music store pool). Some processes represented in the pools are dependent on one another, therefore in order to control the flow of activities decision base gateways are used.

1.2 Steps followed during the project

1. Implementation of the proposed solution using BPMN notation, that allows to formulate the first hypotheses and to define the processes of each actor.
2. Subsequently, the resulting graph is converted to Petri Net, which facilitates in-depth analysis and allows verification of the basic properties of the process. In particular, the process of each participant is first converted into a workflow and then the entire scene is represented through a workflow system.
3. Analysis of the behavior and structure of each actor's module and general system.

The BPMN diagrams were designed in Signavio (now part of SAP), whereas the Petri nets analysis was conducted on both Woped and Woflan.

In the next section, with the aid of a BPMN diagram is described the sequence of activities that are performed in each pool.

2 Designing the process in BPMN

The use of the BPMN notation provides, first of all, a simple but nevertheless complete notation thanks to various decorations to make the meaning of each symbol more detailed and secondly the possibility of managing both the orchestration and choreography among the various pools. In the next part, the three pools of actors involved in the process will be illustrated.

2.1 Customer Pool

The main pool is represented by the Customer Pool, which is the one that starts the whole process. The process starts with the task “*Request Login*” and sequentially the user’s credentials are sent to the Online Music Store Pool to be verified. This request is fundamental because it activated the Online Music Store Pool. At this point, the best way to represent the control held by the store (that can either approve or deny the credentials), is to add an *event-based gateway*. If the connection is denied, the **XOR-join gate** allows the repetition of this sequence of activities. If the connection is successful (login is accepted), the user can “*Check Catalog*” and then can either “*Filter by Author*” or “*Filter by Title*”. After “*Check Results*”, the user can decide through the **inclusive gateway-OR** to “*Play Demo*” (play a 45’ excerpt in streaming) or “*Add song to cart*”.

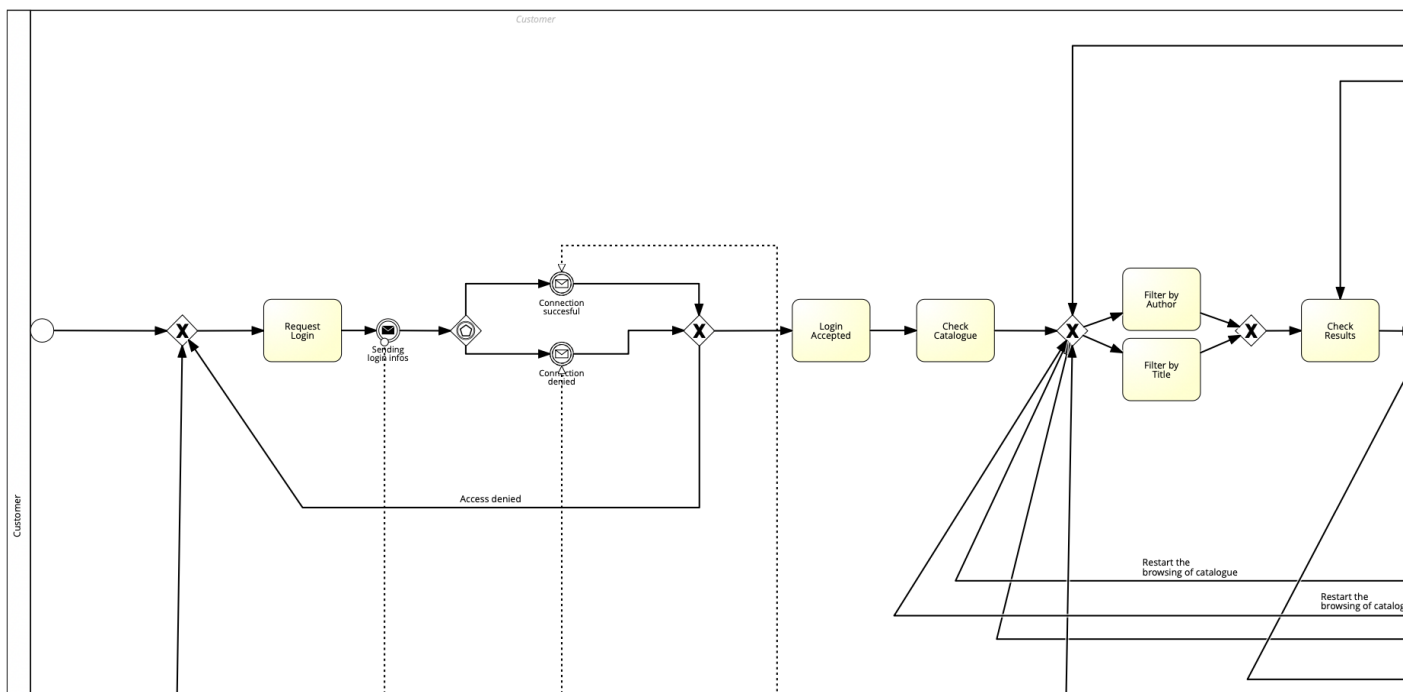


Figure 1 - Start of Customer Pool

Before “*Add song to cart*”, a request “*Check availability and license*” (added feature - not part of the project description) is sent to the Online Music Store Pool to inspect if for example the song’s license is available in the user’s region. If the song is not available, the process is restarted back at the “*Check Catalog*”. If the song is available, it is added to the cart and at this moment, the customer has the possibility through the **inclusive gateway-OR** to repeat the flow of activities (keep browsing), or “*Remove song from cart*” (which is a subprocess in itself), or “*Proceed to checkout*”.

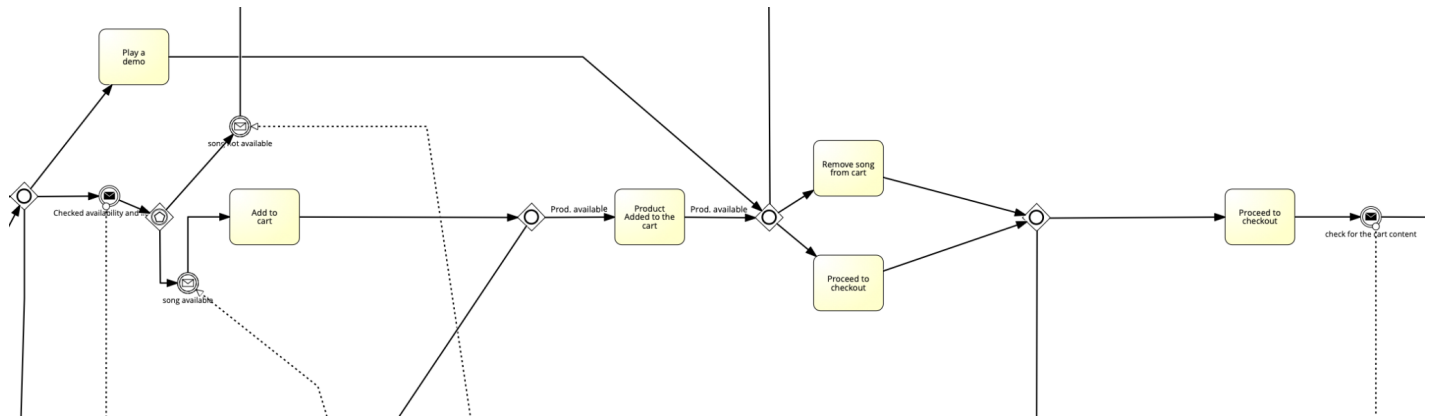


Figure 2 - Checkout of Customer

Prior to real check-out, the customer sends a request to the store which assures that the cart is non-empty. On that occasion, the customer “*Fills credit card information*” and starts communication with the bank. The payment is done by an electronic payment, and the money transfer confirmation is represented by a message flow arrow labeled “*Positive reply*”. In case of a “*Negative reply*”, the payment is denied and the customer can enter another credit card. Once “*Payment accepted*”, parallelly the customer can “*Download Songs*” and receives a “*Confirmation email*” by the Online Music Store Pool. The **XOR-join gate** signifies the completion of checkout and the customer can either logout or continue browsing the catalog. If choosing “*Logout*”, a “*Notify termination*” message is sent to the store. The connection between **XOR-join** at the beginning and **XOR-split** at the end, are necessary to guarantee the possibility for the process to be repeated.

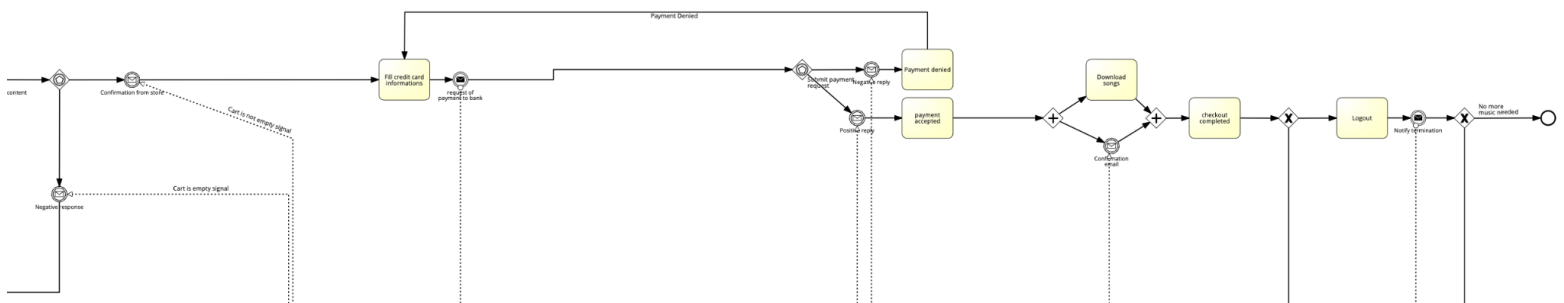


Figure 3 - Ending part of Customer Pool

2.2 Online Music Store Pool

The process starts after an incoming message “*Receive login credentials*” from the Customer Pool. Then “*Check validity of credentials*” searches the credentials in the “*Registered customer*” database. Based on the validity of credentials, the store can deny access (the **XOR-join** allows this process to be executed many times until success - The option of limiting the number of trials is not considered) or confirm by a message and continue the session by waiting for customer interaction.

In case of receiving a “*Request for check*”, the store checks song availability by searching the “*Song and licenses*” database. Immediately when receiving the “*Request for checkout*” by the customer, the store checks the cart content. On condition that the cart is not empty, to represent the waiting time of the bank that checks the credit card balance, an **intermediate time event** is added. Only if the “*Transaction approved*” message is received from the bank, the store sends a “*Send confirmation email*” and the order is saved in the “*Orders*” database. Afterwards, the customer is informed about the termination, and a timer is set to count for 7 days.

After 7 days, the whole process of the store finishes with the “*Send email for review of songs*” activity.

2.3 Bank Pool

The process starts after an incoming message from the customer. The bank “*Checks availability of funds*”. Using the **XOR-split gateway**, if the customer’s funds are not enough, two messages are sent parallelly to notify customer and store and the process (connection) finishes (is interrupted).

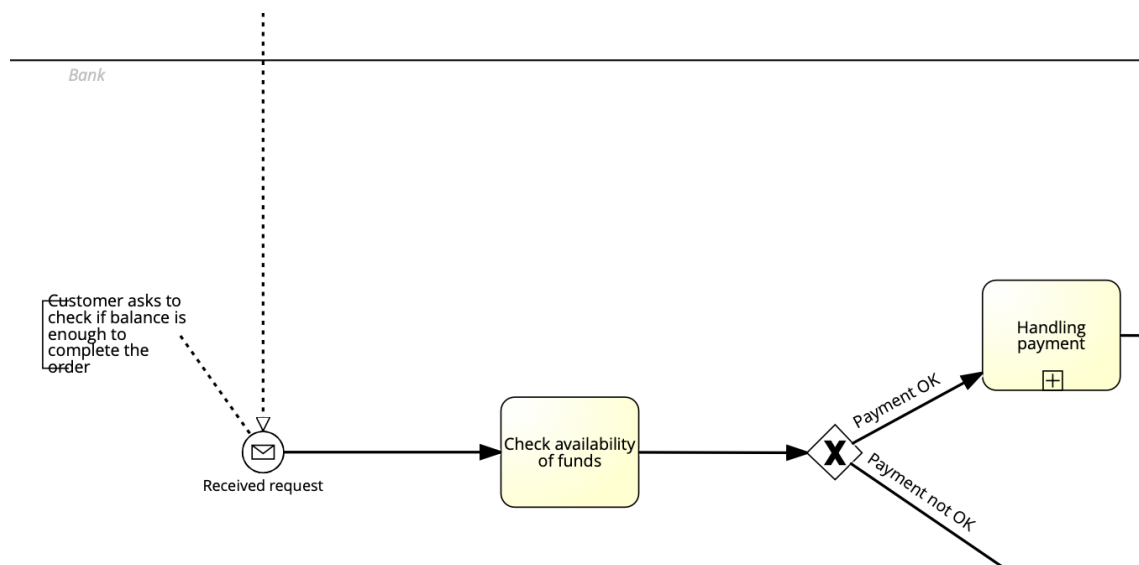
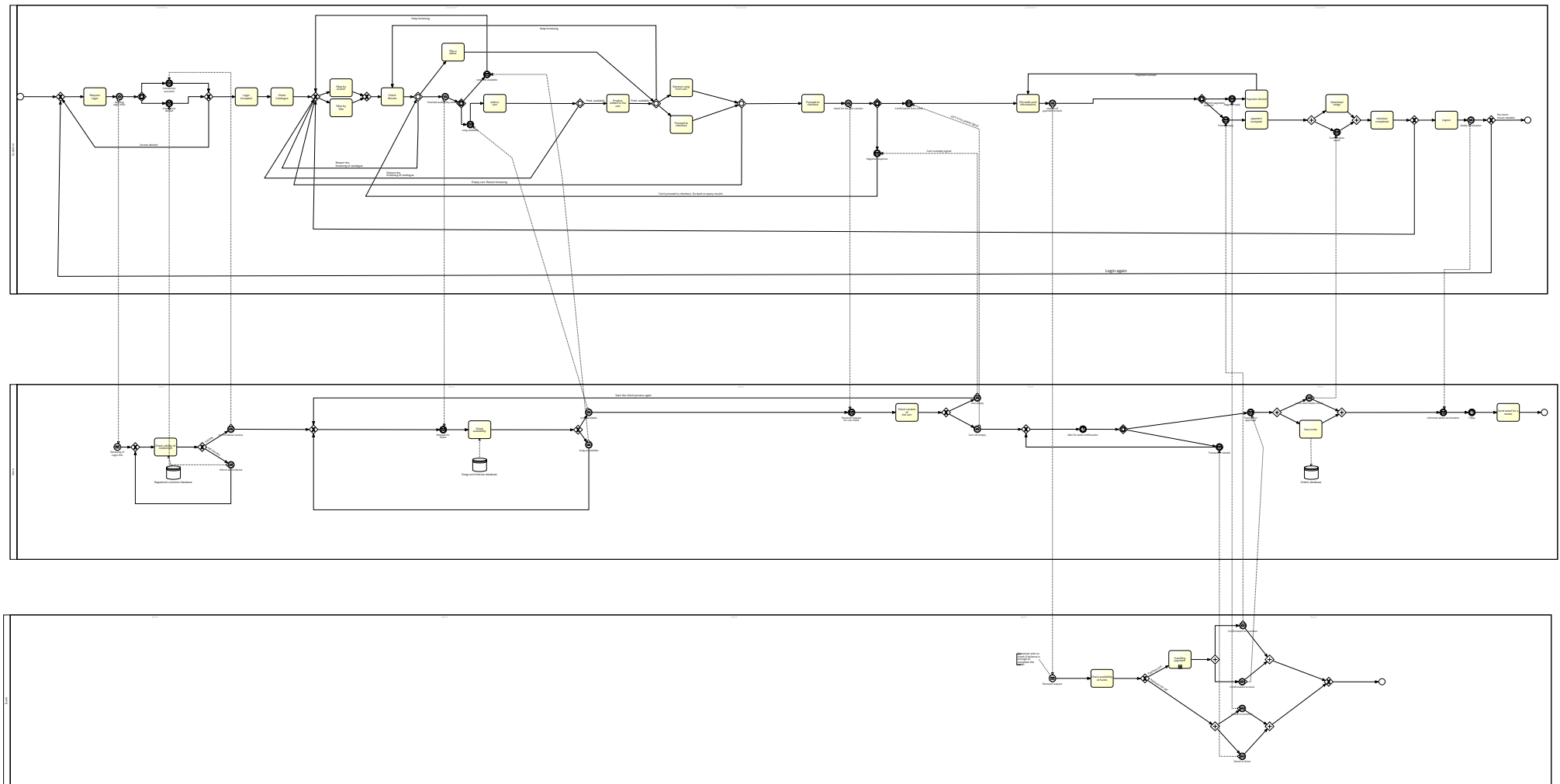


Figure 4 - Start of Bank Pool

If the customer’s funds are enough to cover the buying, “*Handling payment*” subprocess (involves many actions that were decided not to be taken into account in this project) is sequentially followed by the sending of confirmation to both customer and store. Eventually the process is finished successfully.

BPM



3 From BPMN to Petri Net

Once the model was implemented in BPMN, the aforementioned models are transformed into Petri Nets, following the rules for replacing the various "modules" of the model, in order to perform a workflow analysis, with the aim of identifying the fundamental characteristics network (*s-invariant*, *t-invariant*, *liveness*, *free-choice*, *soundness*, etc).

3.1 Translations Carried out

In this subsection, the meaning of the nodes used in the BPMN model and the respective translation in Petri Net are illustrated in detail. The main changes are:

- Adding an initial and final place for evaluating it in terms of Workflow net
- Replacing arcs with places
- Encoding the gateways in Petri net notation & de-sugaring the net

Initial Place

The *Start node* represents the beginning of the interaction process between *Customer* and *Online Store*. The translation will be carried out by inserting a place whose post-set will consist of a transition that will be enabled and activated as soon as the process begins.

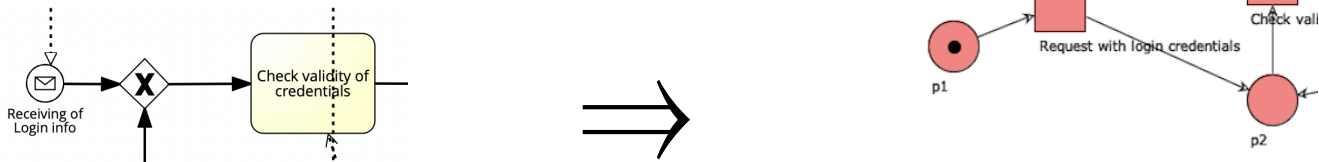


Figure 5 - Initial Place (Start) Conversion

XOR-Gate

Is modified as a place that creates 2 or more branches with respective transitions.

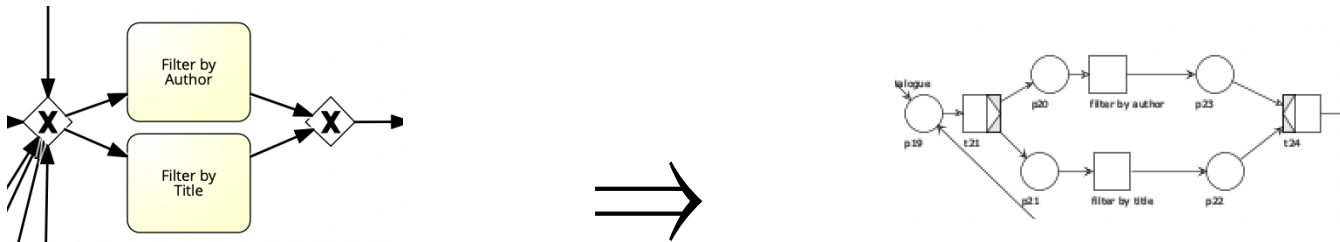


Figure 6 - XOR-Gate Conversion

Event Based Decision Gate

Is replaced with a place and each event in the post-set with a transition.

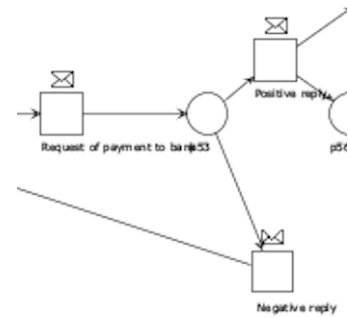
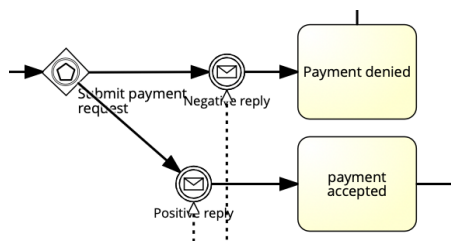


Figure 7 - Event Based Decision Gate Conversion

AND-Gate

Is represented by a transition that has as post-set 2 places, which then will be connected back to a single transition.

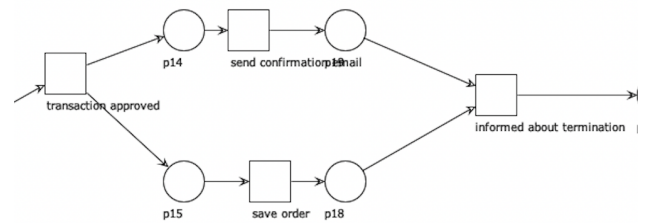
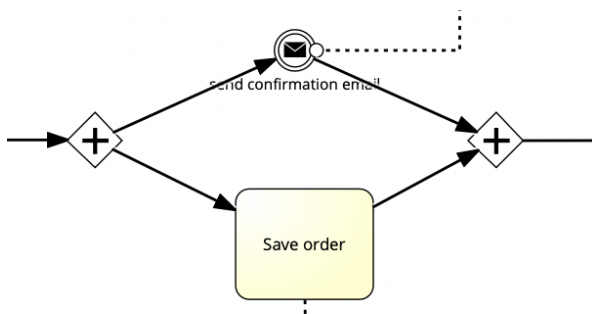


Figure 8 - AND-Gate Conversion

Implicit OR-Gate

Is remodeled by a place which produces 2 or more transitions but then they connect back in one place.

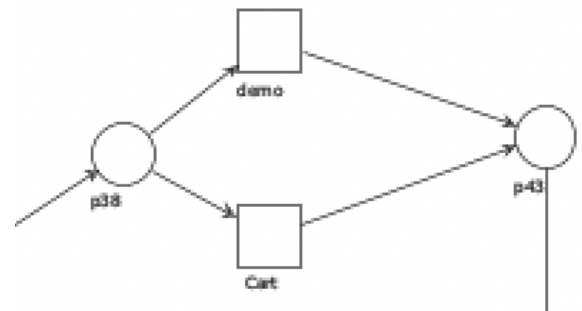
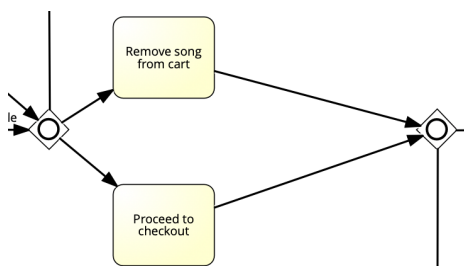


Figure 9 - Implicit OR-Gate Conversion

End Event

Representing the conclusion of the process and making sure the process instance is finished and no tokens are left behind.

3.2 Petri Nets

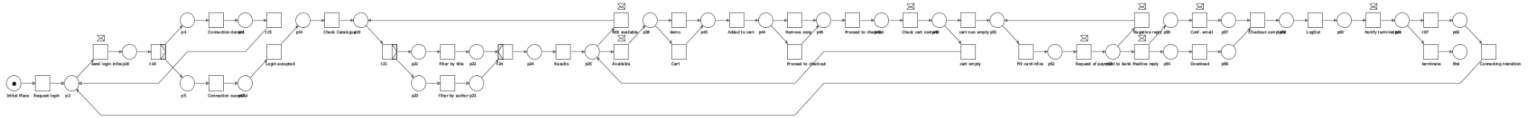


Figure 10 - Customer Petri Net built in Woped

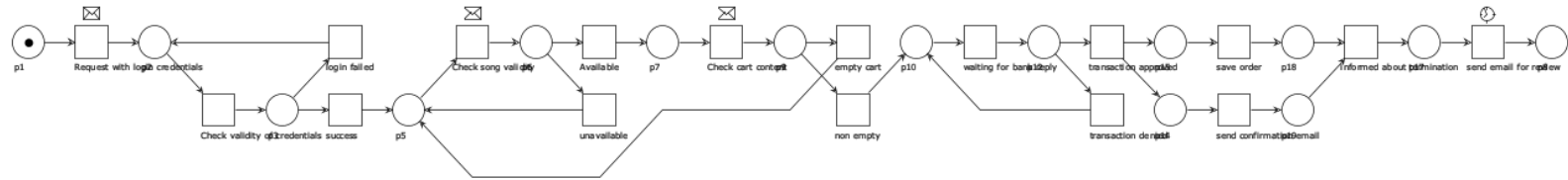


Figure 11 - Online Music Store Petri Net built in Woped

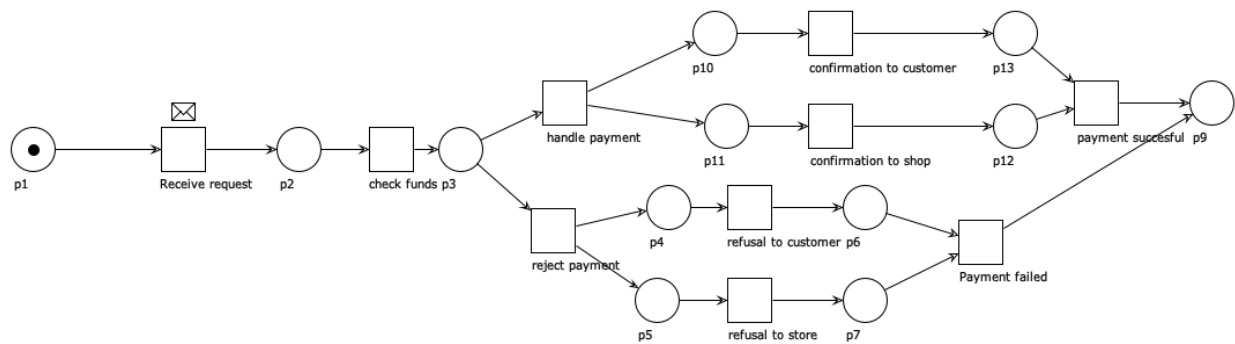


Figure 12 - Bank Petri Net built in Woped

4 Workflow Analysis

Once each actor's Pools are translated into Workflow Net, the analysis is carried out first on each individual network and then on the entire Workflow System. The analysis involves firstly using a brute-force approach through simulation with the token game (not always possible to execute all possible scenarios), secondly reading the reachability graph (unreadable due to the many states especially in the Customer's case) and the semantic analysis of WoPeD.

4.1 Customer

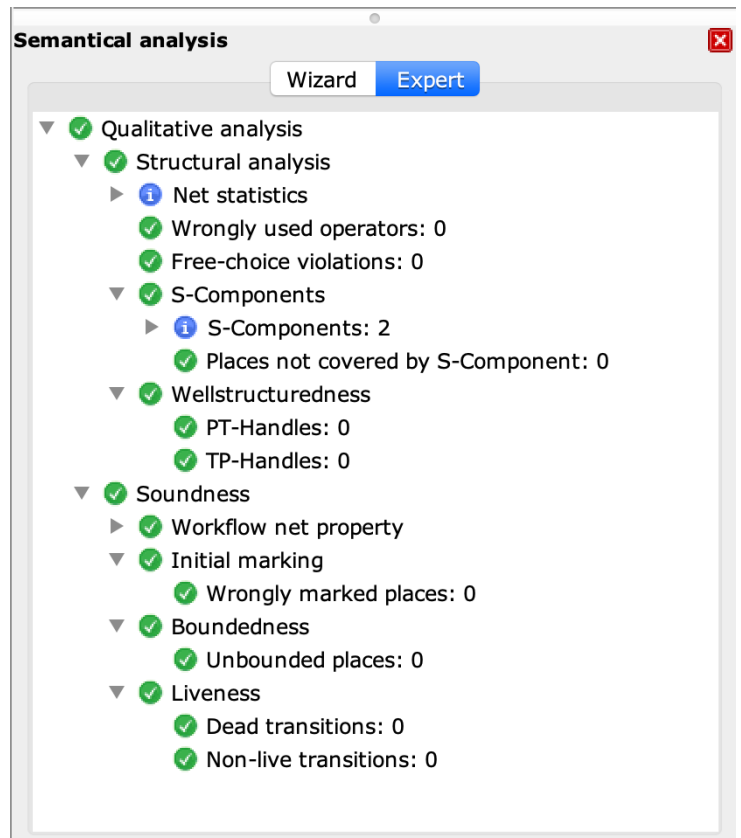
The net is:

- A Workflow Net
- It is free choice
- Not an S-system or T-system
- Deadlock free and bounded
- All soundness requirements are satisfied
- Well handle and well structured (in fact there are no PT- or TP- handles)
- S-covered by **2 S-components**

The net has:

- **33** places, **36** transitions, **3** operators, **77** arcs

The Coverability Graph has **33** vertices and **39** edges.



4.2 Online Music Store

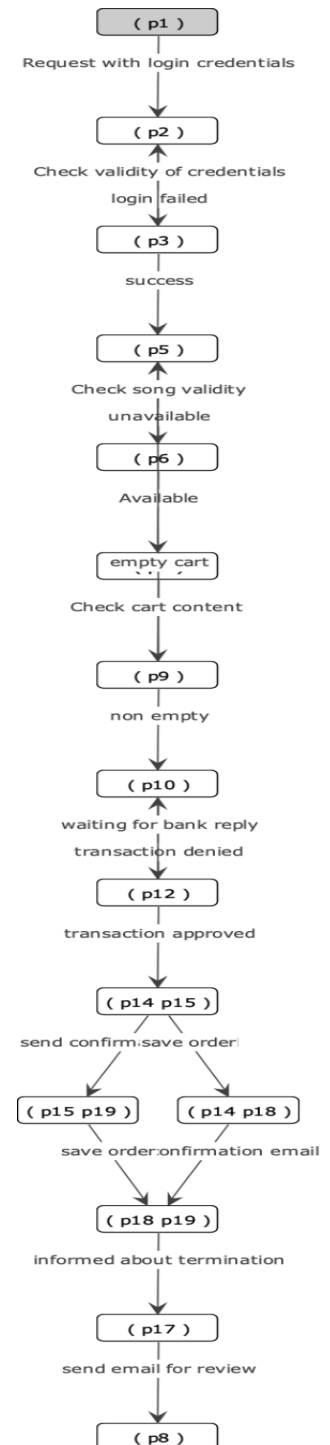
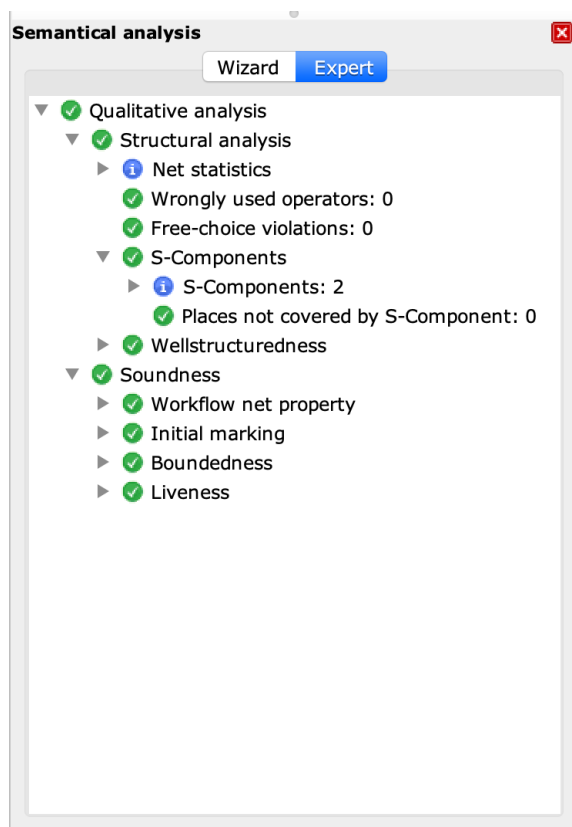
The net is:

- A Workflow Net
- It is free choice
- Not an S-system or T-system
- Deadlock free and bounded
- All soundness requirements are satisfied
- Well handle and well structured (in fact there are no PT- or TP-handles)
- S-covered by 2 S-components

The net has:

- 15 places, 17 transitions, 36 arcs

The Coverability Graph has 15 vertices and 19 edges.



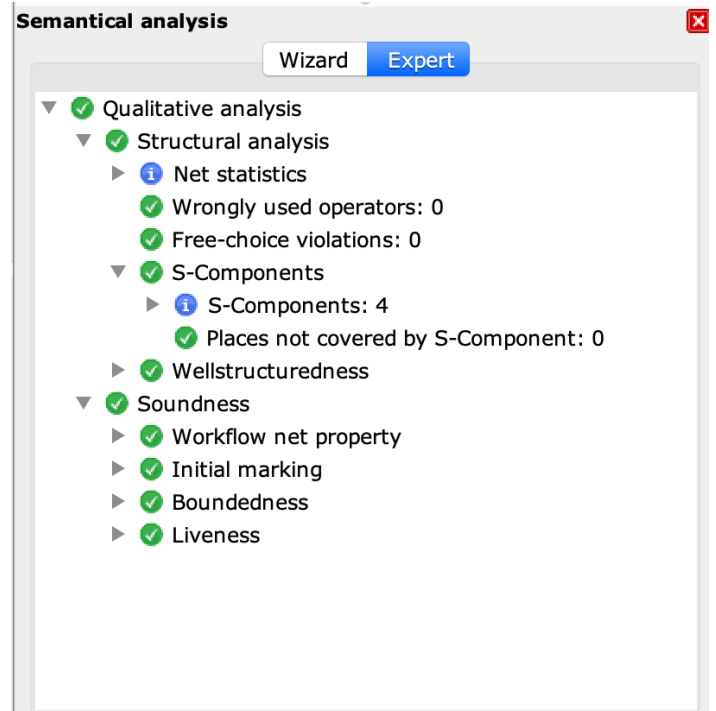
4.3 Bank

The net is:

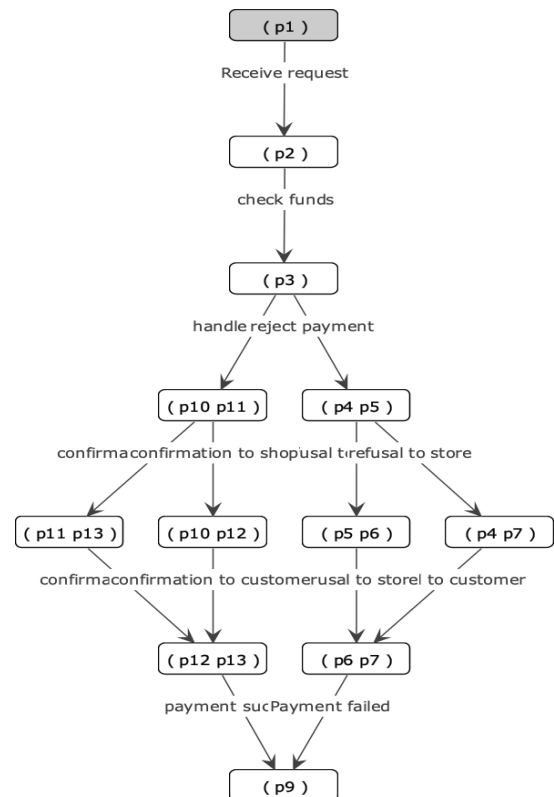
- A Workflow Net
- It is free choice
- Not an S-system or T-system
- Deadlock free and bounded
- All soundness requirements are satisfied
- Well handle and well structured (in fact there are no PT- or TP- handles)
- S-covered by 4 S-components

The net has:

- **12** places, **10** transitions, **24** arcs



The Coverability Graph has **12** vertices and **14** edges.



5 Conclusions

In the final part of our work we were able to connect the flows of all the aforementioned actors resulting in the *bpmn collaboration diagram* that can be seen below. Initially the diagram was unbounded in *p2* thus creating issues in *Woped* at the moment of computing the semantic analysis. After a troubleshooting phase we have been able to alter the configuration thus resulting in a net with the following characteristics:

- Workflow Net
- Free choice
- Deadlock free
- Soundness requirements satisfied
- Well handle and well structured
- S-covered by 6 S-Components
- 40 places, 43 transitions and 92 arcs
- Regarding the *Coverability Graph*, it consists of **39** vertices and **44** edges.

