UNIVERSITÀ DI PISA

Business Process Modeling

Online Movie Store - Project Report

Xinyi GU - 584731
Yian LI - 667279

Department of Informatics
Data Science and Business Informatics

Academic Year 2023/2024

Contents

1 Introduction

The project aims to design and analyze the business processes of an online movie store. The service allows customer to compile and purchase a playlist of movies over the Internet. The customer starts the process by logging in and browsing the catalogue. The customer can search for a movie by author or title, select to play a movie trailer in streaming, and add or remove items from the playlist. Once the playlist is built, the customer can either proceed to check out or to abandon the session (and the playlist will become empty). During checkout, the customer selects an expiration date for the playlist and provides credit card details for payment. After the order is processed successfully, a confirmation email is sent to the customer, the movies become available for download, and the order is saved. After checkout, the customer can log out or continue to compile another playlist.

The Business Process Model and Notation (BPMN) is selected to design the process for two main reasons. First, BPMN is well-suited for modeling choreography, which, in this case, involves the interactions between the customers, the movie store system, and the bank payment system. Second, BPMN allows for a straightforward translation into executable workflows, such as Petri nets, making it easier to implement and automate the process.

Three pools are modeled for this business process:

- **Customer**: The customer is the end-user who interacts with the online movie store. They are the protagonist of the business process.
- **Movie Store System**:The system is responsible for managing the backend processes of the online movie store, triggered by the customer. It handles customer requests, processes payments, manages the movie catalogue, and communicates with the bank for payment verification.
- **Bank payment system**: The bank processes the payment transactions, triggered by the movie store system. It verifies the payment details and manage the payment to the movie store system.

2 BPMN diagram

The complete process is shown in Figure 1. There are three pools: one for the customer, one for the movie store system, and one for the bank system.

The process begins when the customer opens the online store. The customer must log in to access the services offered by the website. After logging in, the default movie catalogue page is provided, but the customer can also choose to search movies (by author or title). They can play the movie trailer, add or remove movie to the playlist. This process can be repeated to add multiple movies into their playlist. When the playlist is complete, the customer can proceed to check out. Alternatively, they can also decide to abandon the session.

If the customer chooses to proceed to check out, they will be asked to select an expiration date for the playlist, enter payment card information, and then submit a payment request. Upon receiving the payment request, the store system will send an authorization request to the bank and wait for the response. When the bank receives the authorization request, the system will validate the payment information, then decide whether to authorize the payment and send the result to the store system.

Based on the bank's feedback, if the payment is rejected, the store system will send a failure notification to the user and end the current checkout process. If the bank approves the payment, the system will send a confirmation email to the user, save the order, and allow the user to download the movies. Accordingly, after receiving the confirmation email, the user will be allowed to download the movies and find the saved order in the order history. In both case, after the checkout process, the user can choose to return to the homepage to start editing a new playlist or log out, ending the entire process.

2.1 Customer

The pool of the website should be more or less symmetric to that of the customer, each step of the customer's operation is submitting a request to the web page and receiving a response from the web page. (The website has some additional tasks in the payment process, which will be detailed later.) According to the project requirements, we mainly arrange the process from the customer's perspective.

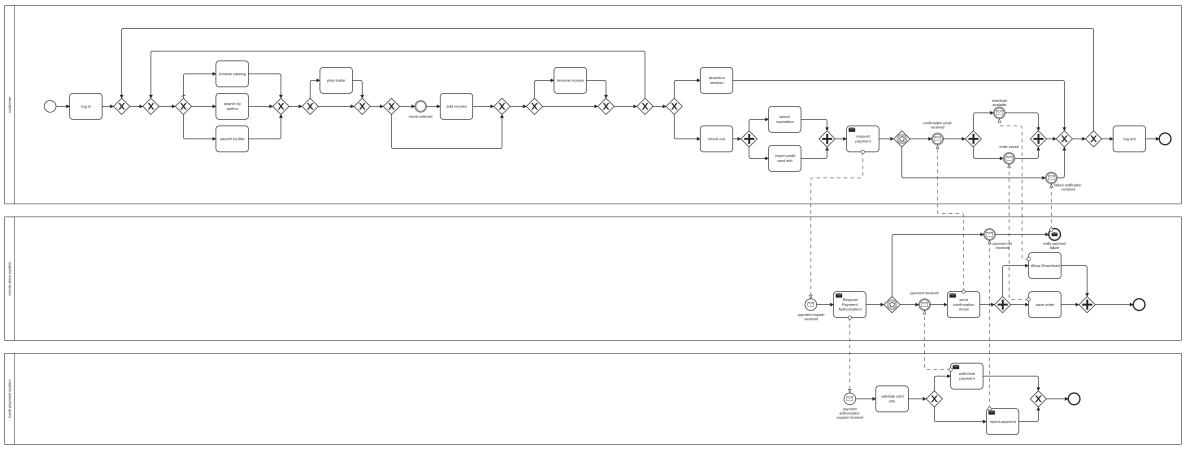

Figure 1: Complete BPMN diagram

2.1.1 Playlist compiling

As shown in the Figure 2, for the customer, compiling a playlist starts with browsing/searching for interesting movies (using an XOR gateway, with browsing the catalogue as the default event). The customer can freely choose to play or not play the trailer, add or not add movies, remove or not remove movies (all using XOR gateways), and can repeat the above steps if they want to add multiple movies or if the playlist is still empty, until a satisfactory playlist is obtained. Before adding a movie, we used an intermediate event to indicate the state of selecting a movie, making the process from 'browse catalogue' to 'add movie' clearer.

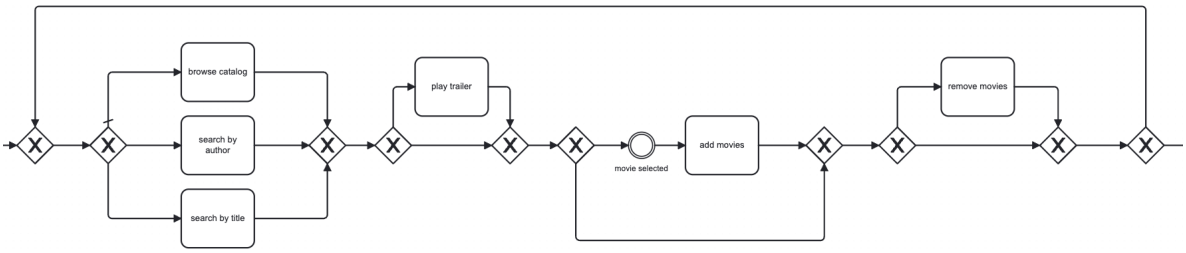

Figure 2: Playlist compilation

2.1.2 Payment

As shown in the Figure 3, after editing the playlist, the customer can choose to check out or abandon the current session (using an XOR gateway). If the customer chooses to abandon the current session, the playlist will be cleared, and they can either return to the initial interface to compile another playlist or log out (using an XOR gateway).

In the case of checking out, after submitting the payment request, the next activity is not up to the customer, they should wait for the payment feedback so the event-based gateway is applied here. Upon payment approval, the customer will receive a confirmation email, the order will be saved, and the customer will be allowed to download the movies. We consider these three events to happen simultaneously, but since an event-based gateway cannot directly connect to a parallel gateway, we chose to handle the event of receiving the confirmation email first. If the payment fails, the order will not be saved, and the playlist will be cleared automatically.

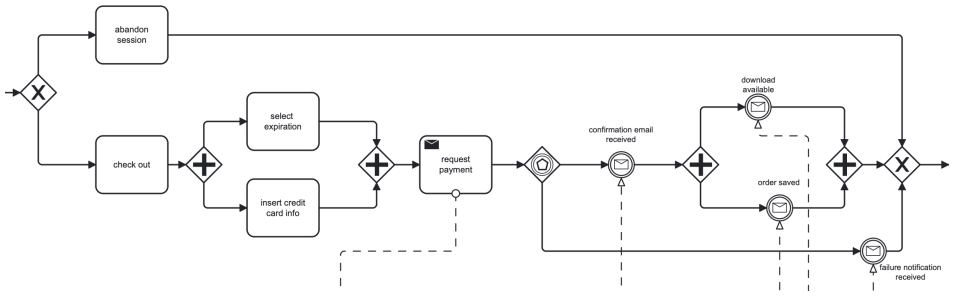

Figure 3: Payment

2.2 Movie store system

For the movie store system pool (Figure 4), we have simplified the online store's process and focus only on the payment part.

The process is triggered by the customer's payment request. The system first requests payment authorization from the bank and proceeds with subsequent processes based on the authorization result returned by the bank (event-based gateway). The order is terminated if the payment is rejected and a failure notification is sent to the customer. If the payment is approved, the system sends a confirmation email to the customer, saves the order, and allow the customer to download the movie(parallel gateway).

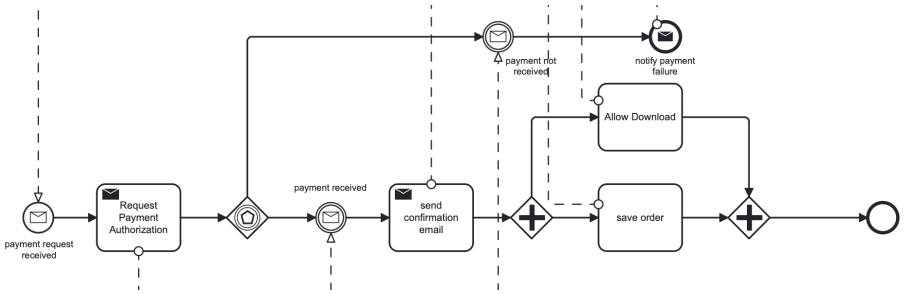

Figure 4: Movie store system

2.3 Bank payment system

The bank only participates in the payment part of the process (Figure 5). The process starts when the bank receives the payment authorization request. The bank then determines whether to approve the request based on the validation of credit card information as well as the sufficiency of the fund (XOR gateway) and send the feedback to the store system.

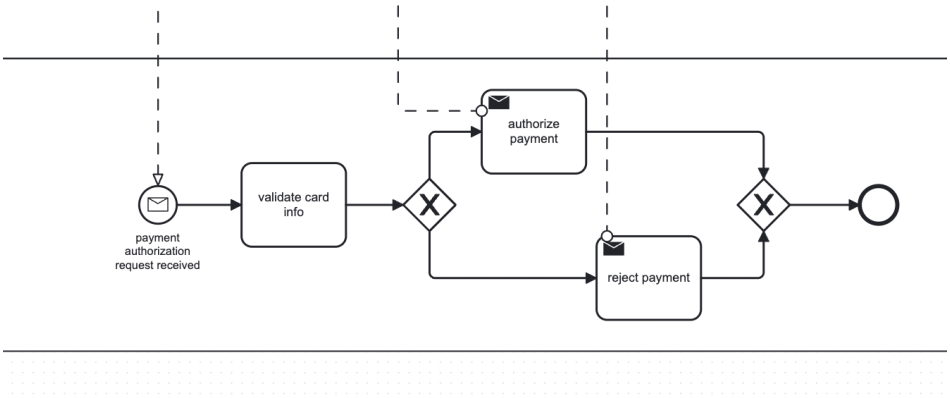

Figure 5: Bank payment system

3 Petri net

The BPMN diagrams have been transformed into Petri nets. The transformation was first applied to each pool individually. The process to achieve this translation involved several steps:

- converting each sequence flow into place.
- converting each activity/event, XOR/AND gateways into transition, and event-based gateways into places.
- desugarizing XOR splits/joins for clarity and simplicity.
- adding initial and final places.

All three Petri nets result in workflow nets with the presence of unique initial and final places, and all nodes are on a path from the initial place to the final place. Moreover, they are safe and sound, on the basis of which the workflow nets can be further connected based on the message flows between them and become workflow modules. Finally, the multiple initial and final places were merged to create a comprehensive workflow system.

The WoPeD software was used to draw the workflow nets and perform semantic analysis (focusing on the structural integrity and soundness of the networks), as well as the generation of coverability (reachability) graphs.

3.1 Semantic analysis

3.1.1 Customer

The Petri net for the Customer pool is illustrated in Figure 6. As shown in Figure 7, the customer Petri net consists of 48 places, 54 transitions, and 112 arcs. Here are some key properties of the system:

- **Sound**: The net is qualifies as a sound net as it is live and bounded.
- **Free-choice**: Most transitions have only one input place, so the pre-sets of each pair of them are either equal or disjoint. The exception is the transitions transformed from *AND-joins*, where the places in the pre-set of these transitions have only one corresponding post-set which is themselves.
- **Safe**: The net is safe because it is free-choice and sound.
- **Not S-/T-nets**: The transitions transformed from *AND-joins* have multiple outgoing arcs and the places transformed from the sequence flow before *XOR-splits* and after *XOR-joins* have multiple incoming/outgoing arcs.
- **S-coverable**: After adding the reset transition, the net consists of 4 S-components, which differs from each other only in the part of 2 parallel blocks. All places in the net are covered by S-components.

- **Positive S-invariant**: The semi-positive S-invariant can be easily obtained by assigning the weight 0 to all places not covered by a S-component. Then the positive S-invariant can be found by summing up the semi-positive S-invariants.

- **Well-structured**: The net with the added reset transition is *well-handled*, as it contains no PT-handles or TP-handles, thus, the workflow net is well-structured.

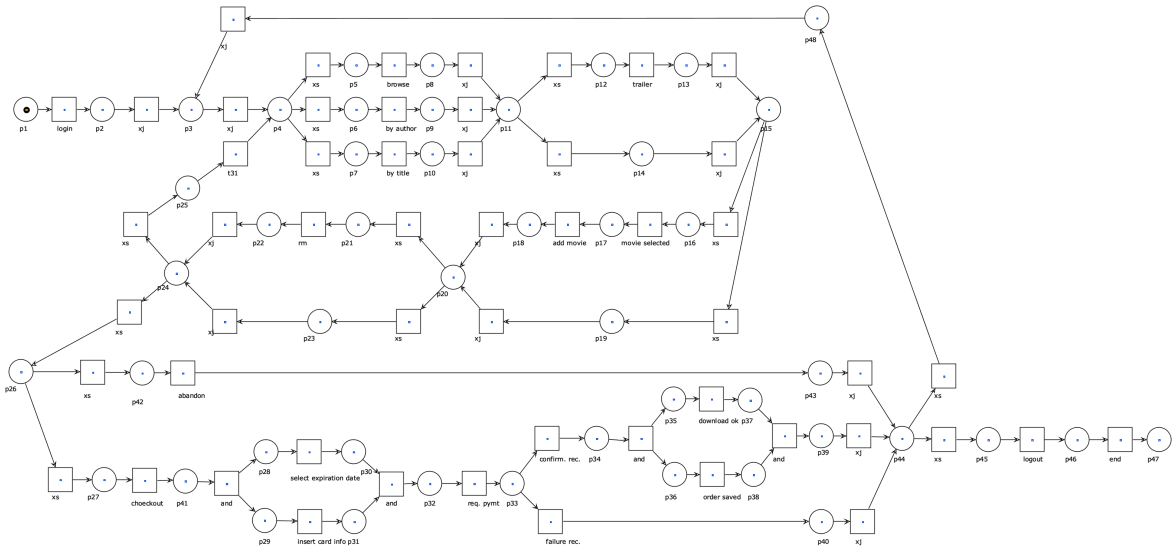

Figure 6: Petri net Customer

3.1.2 Movie store system

Figure 9 shows the Petri net of the movie store system. The properties of the net (Figure 8) are quite similar to those of the customer Petri net, so a detailed analysis is omitted. Two key differences are highlighted: the number of places, transitions, arcs, and the number of S-components.

The Petri net for the movies store system consists of 13 places, 12 transitions, and 26 arcs. It has 2 S-components, meaning it can be decomposed into 2 smaller sub S-nets, thus, the positive S-invariant can be obtained by adding 2 semi-positive uniform S-invariants.

3.1.3 Bank payment system

The Petri net of the bank payment system is shown in Figure 10. The properties of the net (Figure 11) is also similar to those of the customer and the movie store system nets. The number of places, transitions, arcs and the number of S-components is different.

As shown in , the Petri net for the bank payment system consists of 9 places, 9 transitions, and 18 arcs. It has 1 S-component, indicating that the workflow net is an S-system, The S-invariant for this net can be represented by a vector `[k,..., k]` of length 9, where k is a constant. The S-invariant assigns the same weight k to each place in the net.

3.1.4 Complete Workflow system

The Petri net for the entire scenario (Figure 13) is formed by connect the 3 individual petri net together according to the message flows. The message flows were converted into places. Upon this step, workflow modules are obtained. To construct a valid workflow system, different initial and final places of the workflow modules should be joined into unique initial and final place, which brings further adjustments with respect to the seperate nets.

- **Initial place**: The unique initial place of the full scenario is represented by the customer's starting point, as the customer initiates the entire process. This means the entire business

Figure 7: Semantic analysis Petri net Customer

Figure 8: Semantic analysis Petri net movie store system

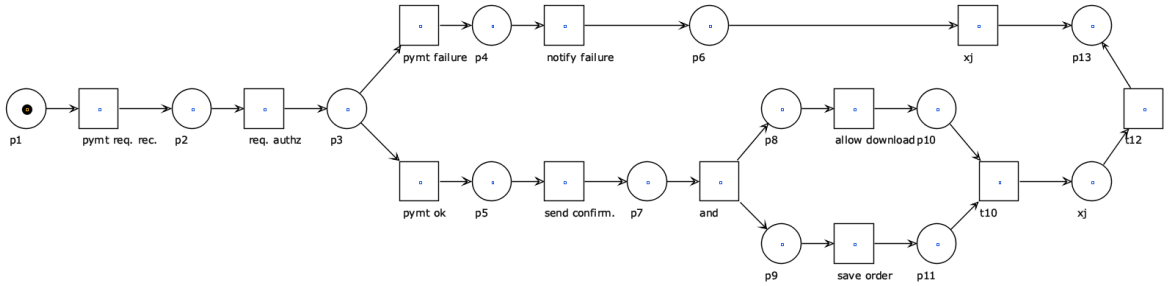

Figure 9: Petri net Movie store system

process begins with the customer's actions, which then trigger related processes in the movie store system and the bank system.

- **Final place**: The unique final place of the scenario is also represented by the customer's endpoint. The movie store system and bank payment system conclude earlier than the customer, and their termination points should align with the completion of the payment process, which requires an *AND-join* of the final places. Therefore, an *AND-join* transition is added after the XOR-join of payment process and before the XOR-split where the customer decides whether to *log out* or *compile a new playlist*. The final places of the store and bank systems, along with the end of the payment process, are thus merged.

The complete Petri net consists of 78 places, 77 transitions, and 176 arcs.

As shown in Figure 12, the complete system remains sound while certain properties are changed with respect to the separate nets:

- **Loss of Free-Choice**: Due to the introduction of event-based gateways, the merged net is no longer free-choice. This is because, in the complete scenario, event-based gateways introduce pairs of transitions whose pre-sets intersect, violating the definition of a free-choice net. Specifically, there are two violations of free-choice, each corresponding to one of the event-based gateways. This means that the selection of some transitions does not freely depend on the actor and is influenced by the result of payment sent by the store system.

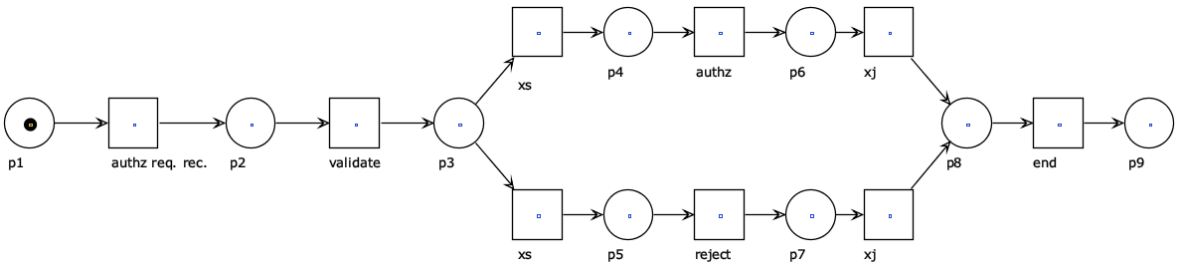

Figure 10: Petri net bank payment system

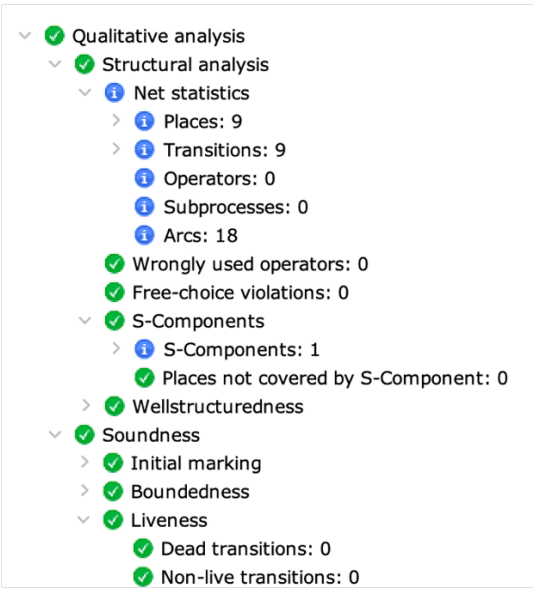

Figure 11: Semantic analysis Petri net bank payment system

Figure 12: Semantic analysis Petri net complete

- **Loss of Well-Structuredness**: The merge of the nets introduced also 21 PT-handles and 19 TP-handles, thus, it is no longer well-structured.
- **Increase in S-Components**: The number of S-components has significantly increased to 30.
- **S-Coverability**: Despite the increase in S-components, the merged net remains S-coverable, meaning that at least one positive S-invariant exists.

3.2 Coverability Graph

All 4 Petri nets are bounded, thus, the reachability graph is finite and coincide with the coverability graph. This means that all places and transitions in the network can be exhaustively listed and analyzed.

- *Customer*: 48 vertices and 58 edges, , start from p1 and end with p47.
- *Movie store system*: 13 vertices and 14 edges, start from p1 and end with p13.
- *Bank payment system*: 9 vertices and 9 edges, start from p1 and end with p9.
- *Full scenario*: 212 vertices and 437 edges, start from p1 and end with p47.

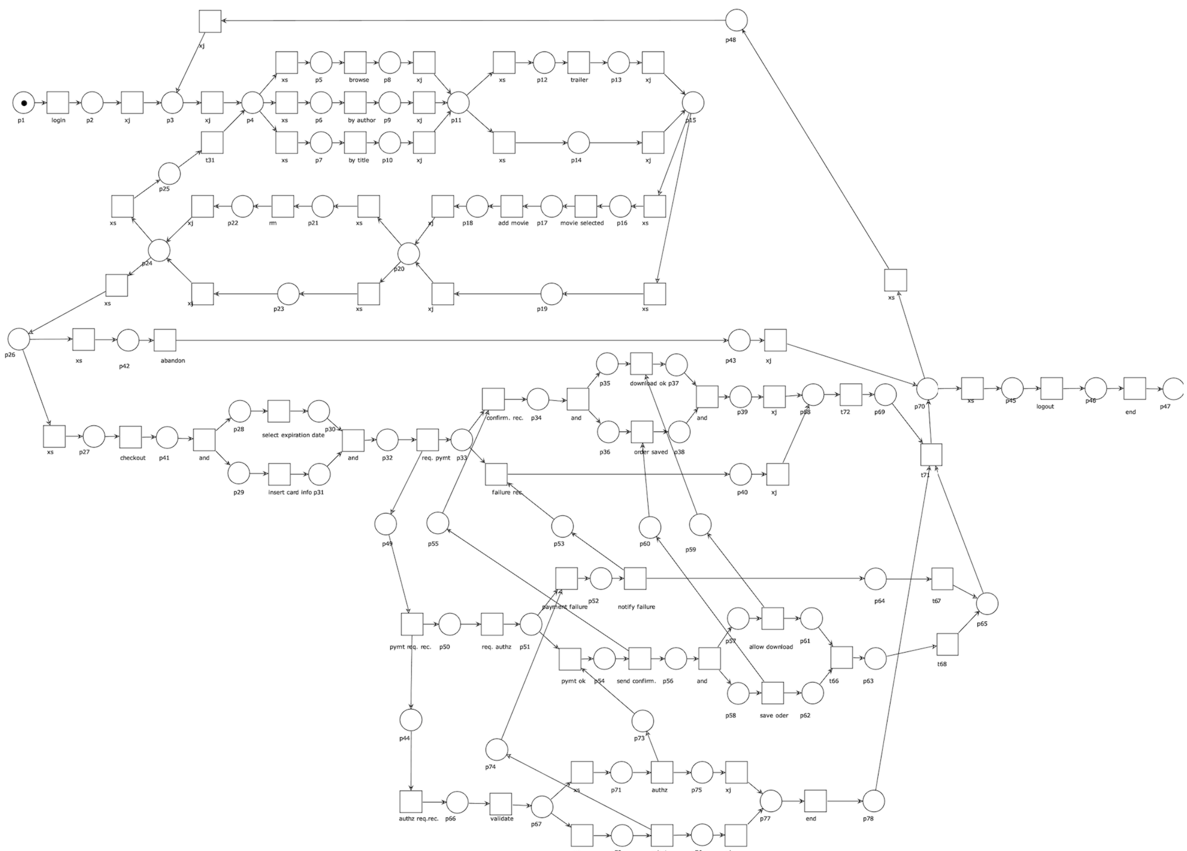

Figure 13: Petri net complete

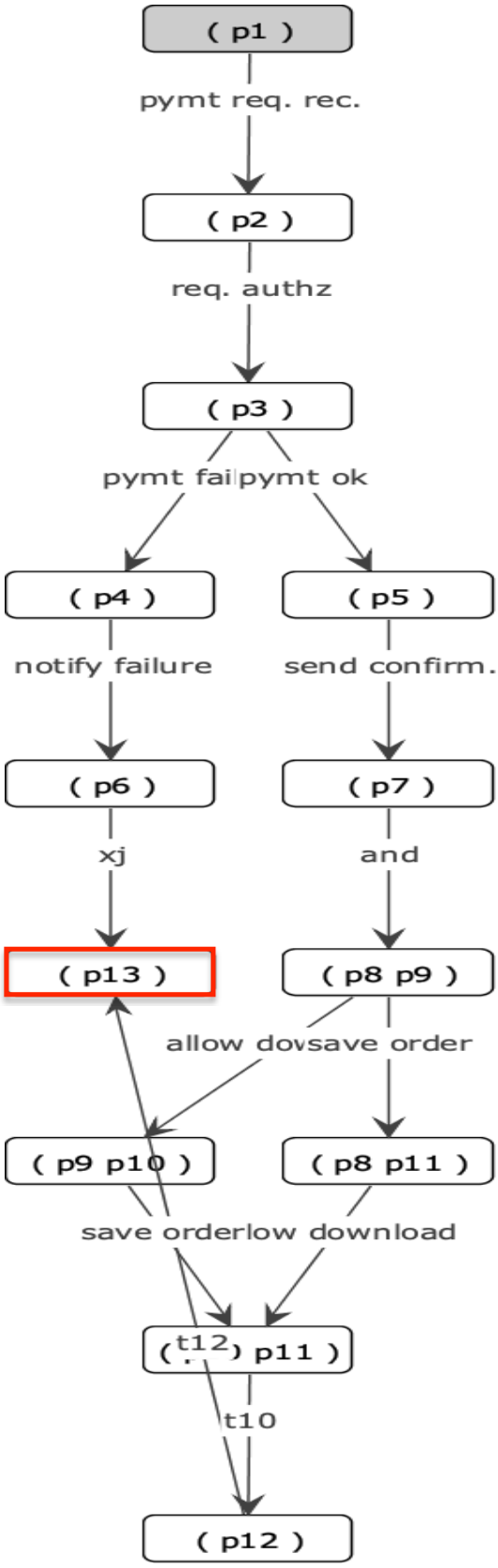

Figure 14: CG movie store system

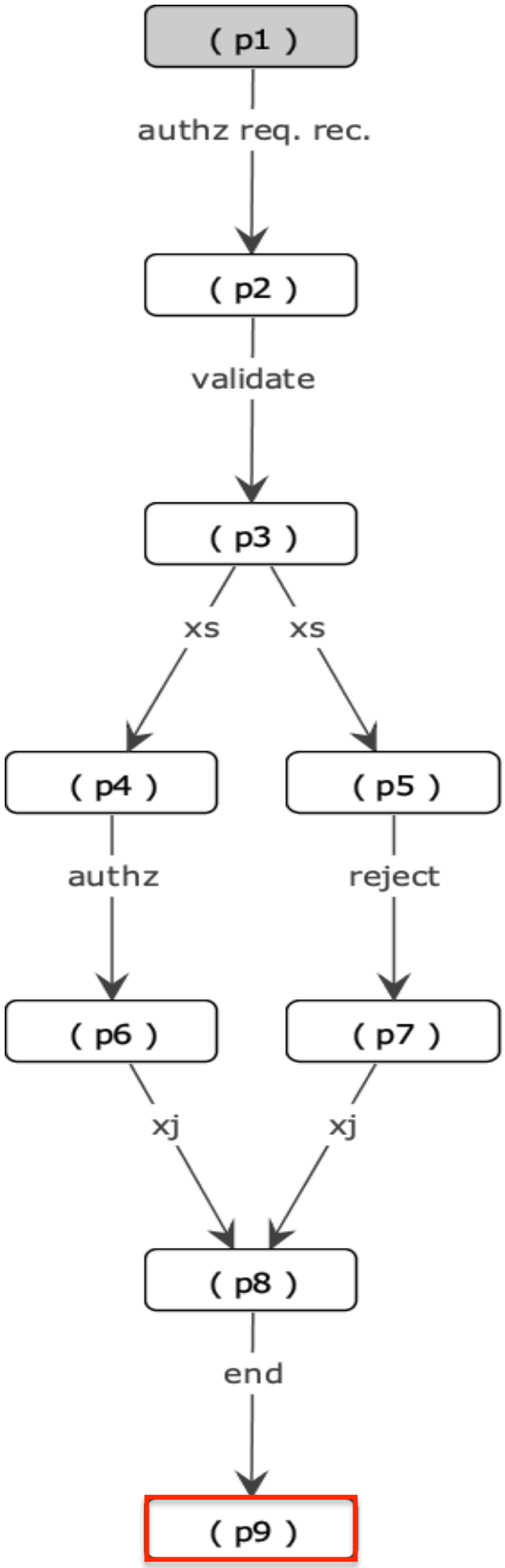

Figure 15: CG bank payment system

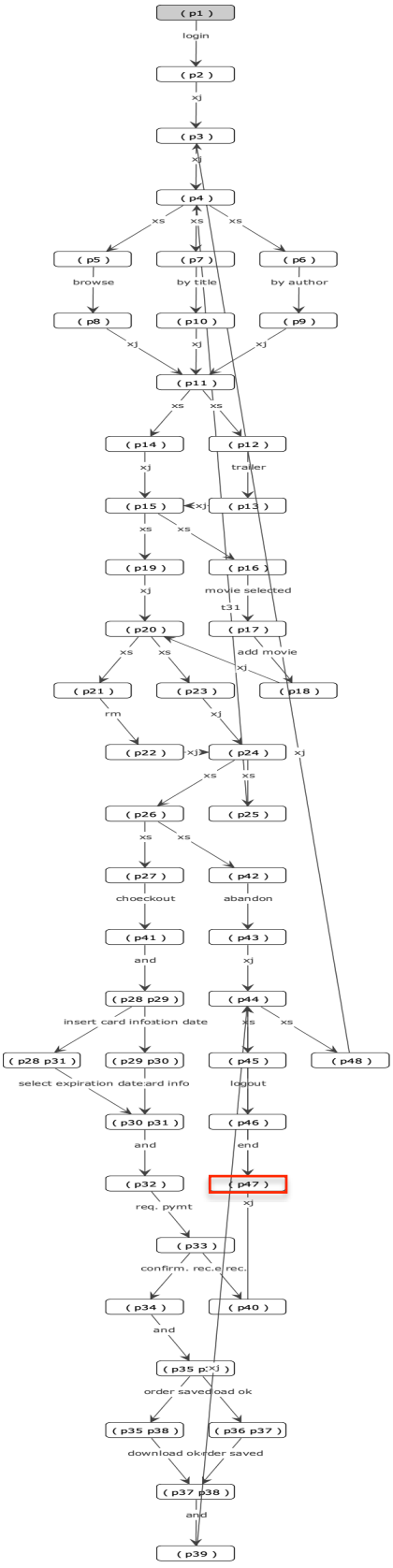

Figure 16: CG customer

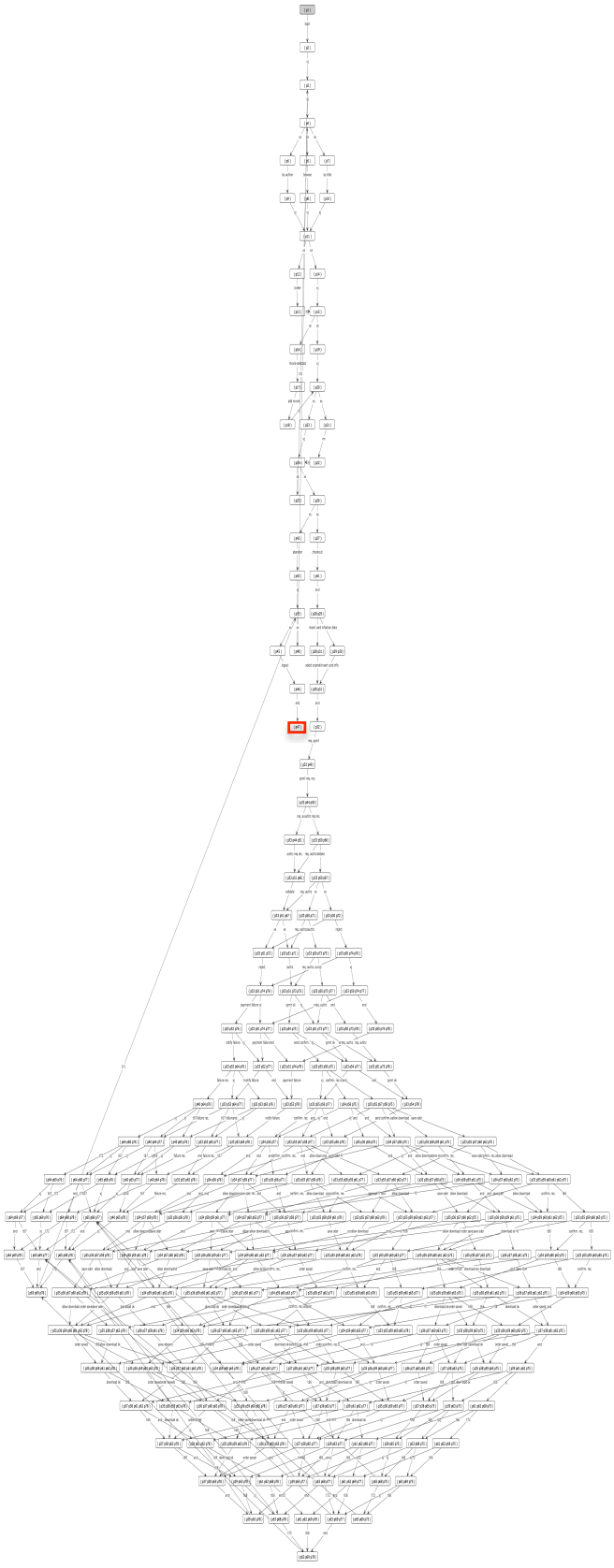

Figure 17: CG complete

4 Project updates and extensions

In the project updates and extensions, we have refined the system pool to fully capture the interactions between the user and the system (Figure 18). As mentioned previously, the operations between the customer and the system are largely symmetrical since every action taken by the customer essentially involves submitting a request to the system and receiving a response. To ensure that the customer proceeds only after receiving the system's response, intermediate events have been added.

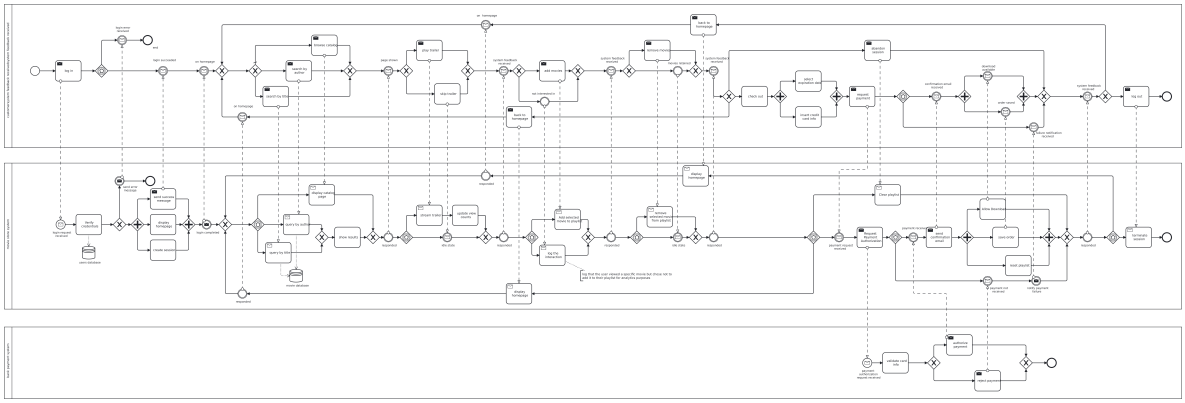

Figure 18: Revised BPMN diagram

4.1 BPMN

4.1.1 Login process enhancements

First, we have improved the login process Figure 19. The customer initiates the process by submitting a login request to the website. The process of the movie store system is triggered by receiving this request, the website verifies the user's credentials against its user database. Using an XOR gateway, if the verification fails, the website returns an error message and ends the process. The customer then receives the error message and the process is also terminated. If the verification is successful, the website (using a parallel gateway) sends a success message, creates a session, and displays the homepage to the customer. At this point, the message flow is sent from the system to the customer, indicating that the login is complete and the customer is on the homepage. Since the customer's next action depends on receiving the login result, an event-based gateway is employed.

4.1.2 Playlist editing process

We introduced additional events in the customer pool (Figure 20) when editing the playlist: skip a trailer, not interested in certain movies, and keep all movies. The customer will then return to the homepage if they choose to continue the editing.

For the system pool (Figure 20), the events mainly based on the choice made by the customer (event-based gateways). The system's responses are as follows:

- **Browse/search**: The system displays the catalog page or if it is a search request, the system needs to query the movie database and display the search results to the customer.
- **Play/not play the trailer**: the system streams the trailer and updates the view counts or simply remains idle.
- **Add/not add the movie**: The system either adds the movie to the playlist or logs the interaction that the customer viewed a specific movie but chose not to add it (for analysis purposes).
- **Remove/not remove the movie**: The system removes the movie from the playlist or remains idle.

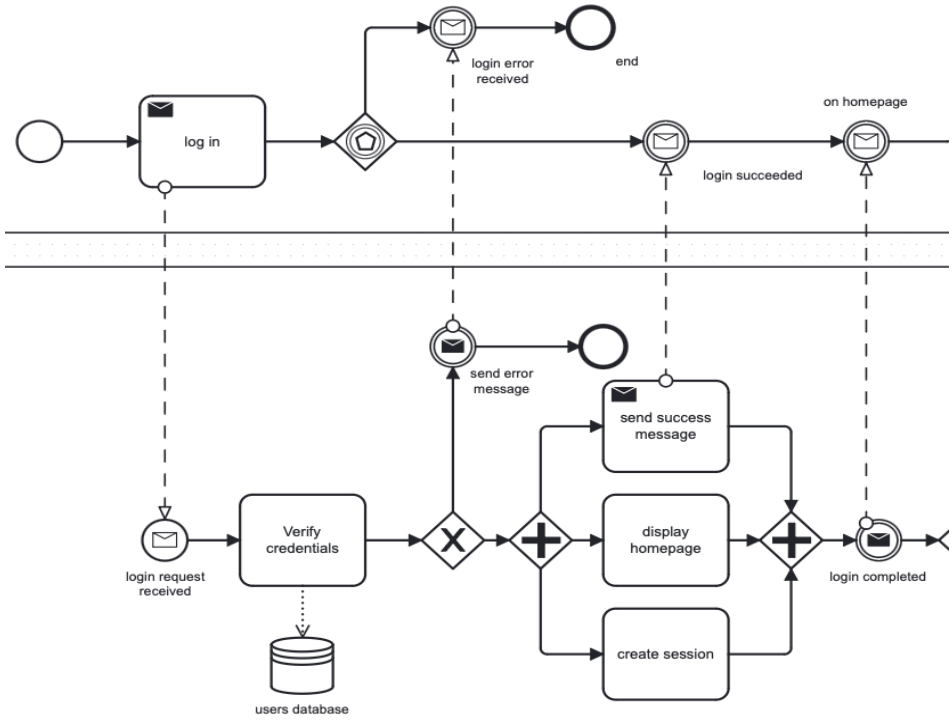

Figure 19: Login process

If the customer chooses to go back to the homepage after previous processes, the system displays the homepage.

After each block of the event-based gateway, a message flow is sent from the system (intermediate event: responded) to the customer (message receipt: system feedback received).

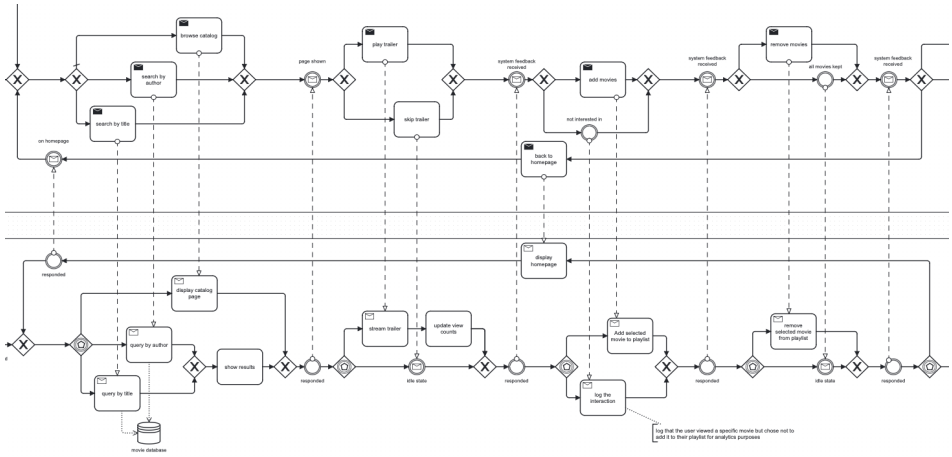

Figure 20: Playlist editing process

4.1.3 Payment Process Optimization

The payment process is essentially the same as described earlier. However, we have made some optimizations (Figure 21):

- After receiving payment, the system will also reset the playlist.

- If the customer chooses to abandon the session, the system will clear the session.
- An intermediate event in the system pool is added, where the message flow is sent from the system (intermediate event: responded) to the customer (message receipt: system feedback received).
- After the payment process, if the customer logs out, the system terminates the session, or if they choose to compile another playlist, they are redirected to the homepage, which the system then displays.

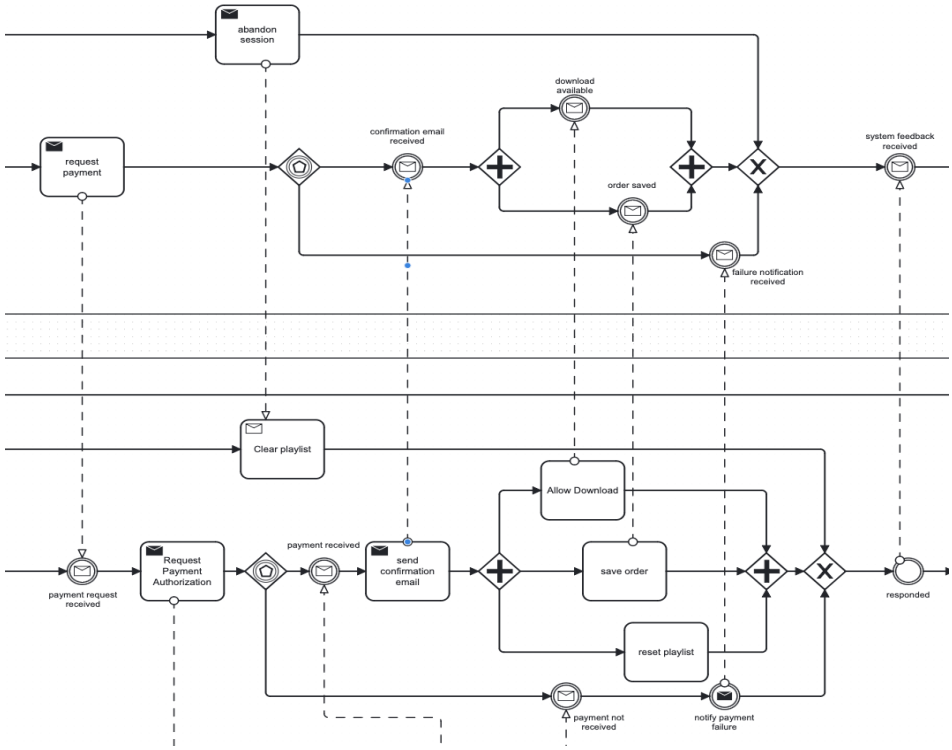

Figure 21: Payment process

4.2 Petri net

The explanation of the transition from BPMN to Petri Nets is omitted, as the same principles outlined in Chapter 3 have been applied.

4.2.1 Semantic analysis

As in the previous cases, the single nets satisfy all the conditions imposed by WoPeD regarding both structure and soundness. To note, the complexity and size of our complete workflow system prevented us from successfully conducting a full semantic analysis using Woped. Despite this, we were able to carry out a diagnosis with Woflan.

- **Customer**: The revised Petri net for the Customer pool is illustrated in Figure 22. As shown in Figure 25, the revised customer Petri net differs from the original one only in its net statistics (60 places, 67 transitions, and 138 arcs).

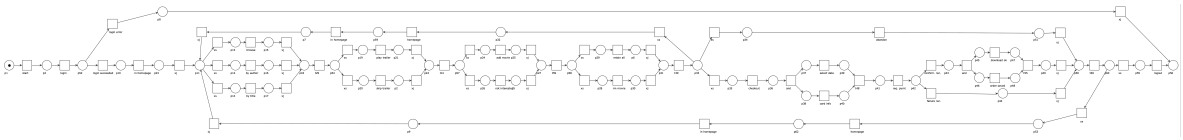

Figure 22: Revised Petri net customer

- **Movie store system**: Figure 23 shows the revised Petri net for the store system pool. The properties of the net (Figure 26) are also quite similar to the original one, two key differences are highlighted w.r.t. the original one: the net statistics (53 places, 56 transitions, and 120 arcs), and the number of S-components (9) as the net introduced more parallel blocks.

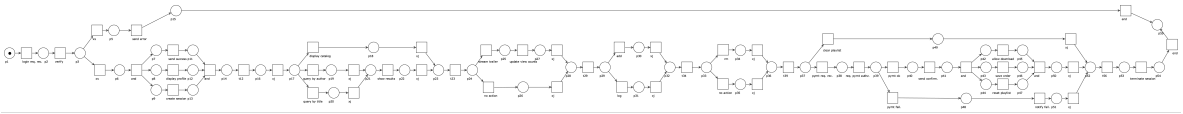

Figure 23: Revised Petri net system

- **Complete**: The revised Petri net for the whole scenario is reported in Figure 24. As Figure 27 shown, it satisfies the workflow net property and the soundness. Despite the lack of semantical analysis, we can confirm the loss of free-choice due to the introduction of event-based gateways.

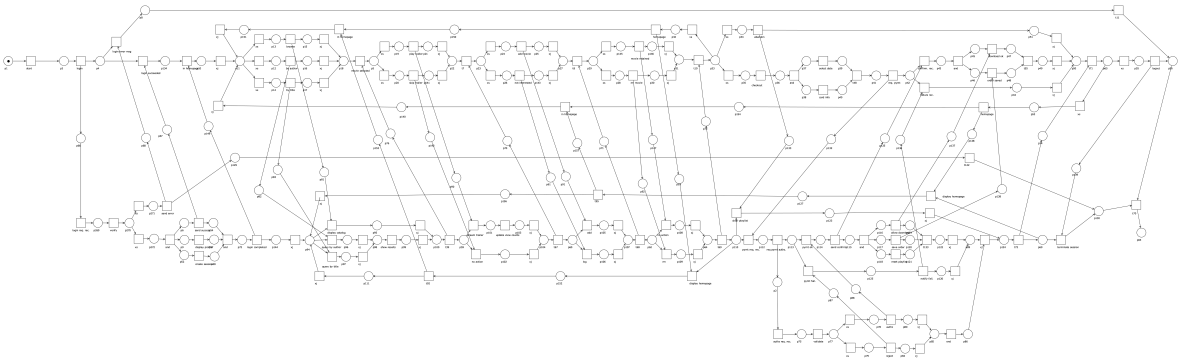

Figure 24: Revised Petri net complete

4.2.2 Coverability graph

The 3 new Petri nets are also bounded, thus, the reachability graph is finite and coincide with the coverability graph.

- *Customer* (Figure 28): 60 vertices and 71 edges, , start from p1 and end with p56.
- *Movie store system* (Figure 29): 57 vertices and 74 edges, start from p1 and end with p55.
- *Full scenario* (Figure 30): 405 vertices and 833 edges, start from p1 and end with p63.

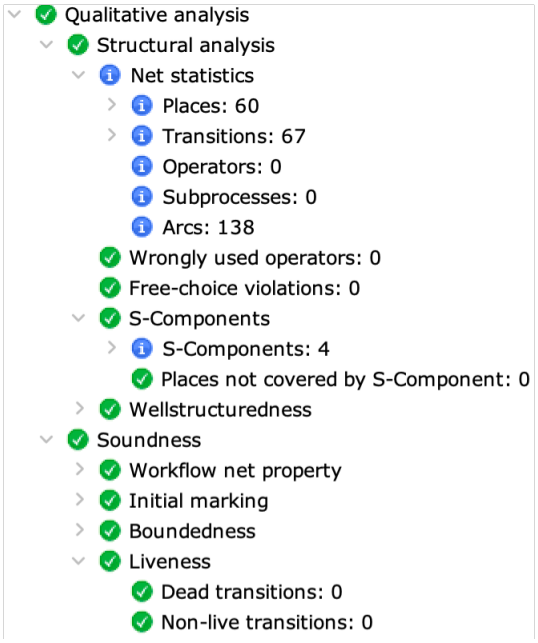

Figure 25: Semantic analysis Petri net Customer (revised)

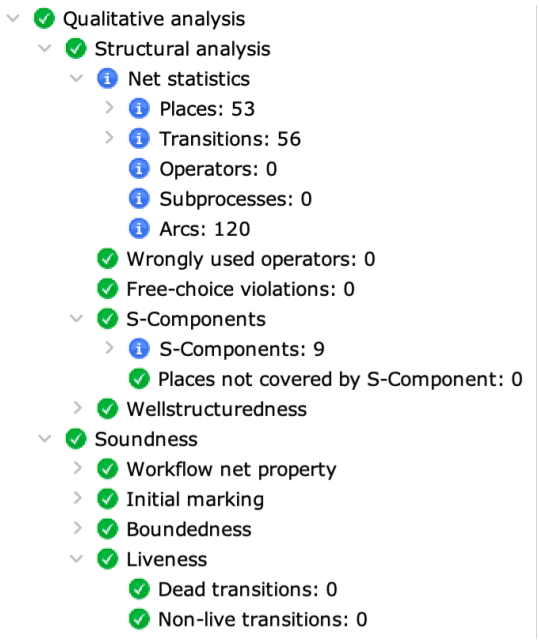

Figure 26: Semantic analysis Petri net movie store system (revised)

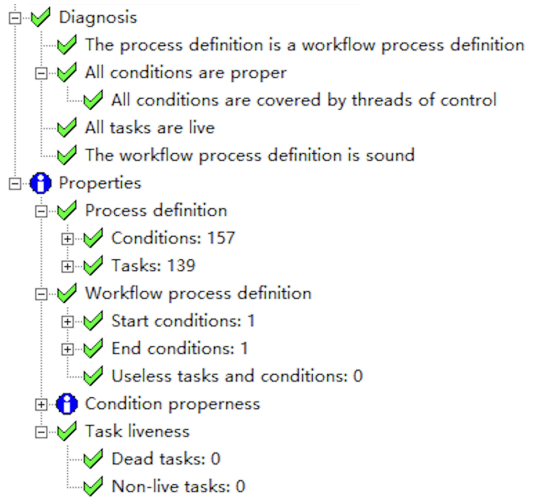

Figure 27: Diagnosis Petri net complete (revised)

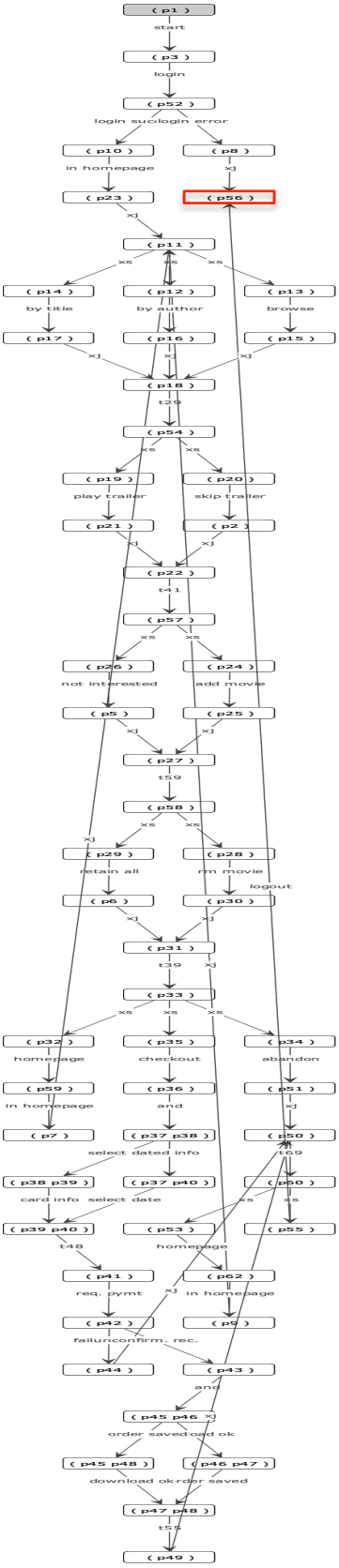

Figure 28: CG customer (revised)

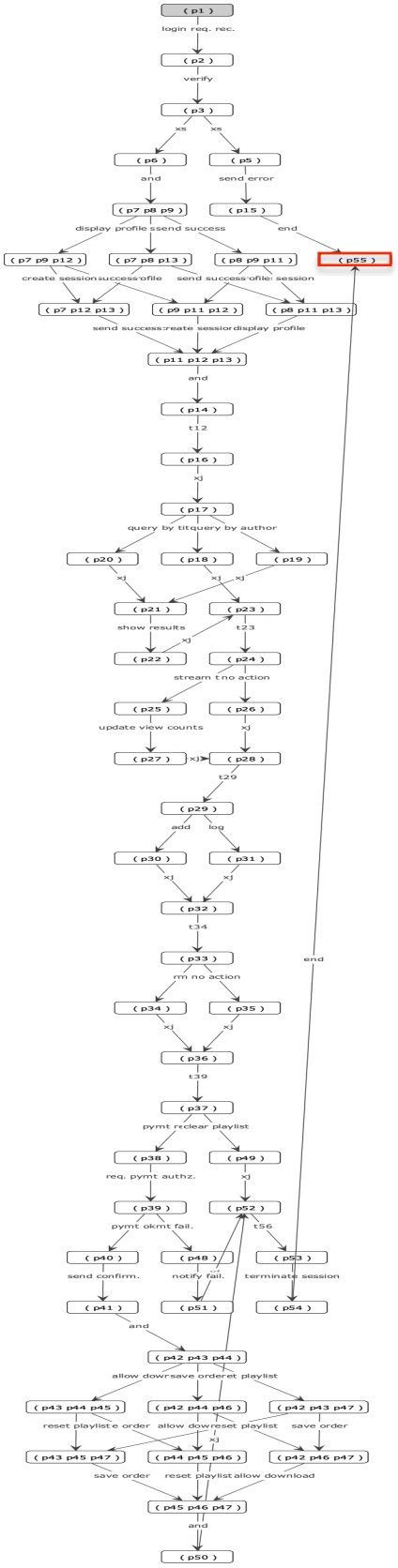

Figure 29: CG movie store system (revised)

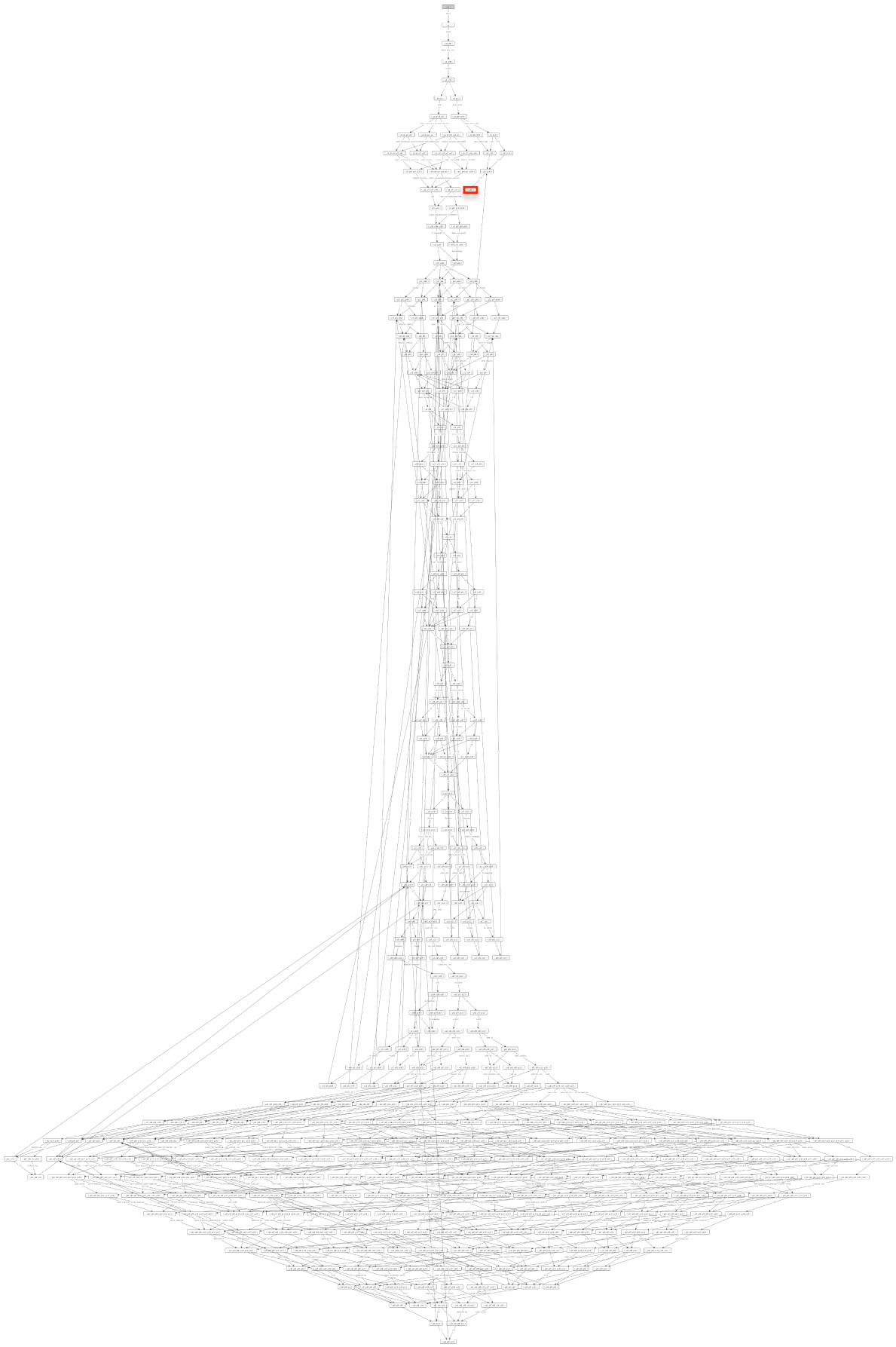

Figure 30: CG complete (revised)