



UNIVERSITÀ DI PISA

MSc IN DATA SCIENCE AND BUSINESS INFORMATICS

## **Business Process Modeling**

**P58: Scuola di Pittura**

Francesco Daquino - 646706

---

ANNO ACCADEMICO 2023 - 2024

# Indice

<b>1</b>	<b>Introduzione</b>	<b>2</b>
<b>2</b>	<b>Modellazione BPMN</b>	<b>2</b>
2.1	Pool Allievo . . . . .	2
2.2	Pool Scuola di pittura . . . . .	3
<b>3</b>	<b>Workflow nets</b>	<b>4</b>
3.1	Workflow net Allievo . . . . .	5
3.2	Workflow net Scuola di pittura . . . . .	6
3.3	Workflow net completa . . . . .	6
<b>4</b>	<b>Variante del processo</b>	<b>7</b>
<b>A</b>	<b>Appendice</b>	<b>9</b>

# 1 Introduzione

Questo elaborato si propone di analizzare i processi che descrivono uno scenario riguardante una scuola di pittura impegnata nella gestione delle richieste degli allievi. Nello specifico, in sezione 2 verranno mostrati i diagrammi di processo progettati seguendo la rappresentazione Business Process Modeling Notation (BPMN) e solo successivamente, nella sezione 3, verranno approfondite le analisi condotte sulle reti ottenute trasformando i suddetti diagrammi.

## 2 Modellazione BPMN

Lo scenario in oggetto è stato modellato utilizzando la notazione grafica BPMN per mezzo del software Signavio, in particolare usando il tool Signavio Process Manager. Questo standard notazionale è risultato più adeguato rispetto al linguaggio di modellazione EPC (Event-Driven Process Chain) dal momento che si tratta di un processo collaborativo. Gli attori individuati sono due, l'Allievo e la Scuola di pittura, per ciascuno dei quali si è deciso di modellare una **pool**. In particolare, all'interno della pool Scuola di pittura sono state individuate due **lane** che rappresentano i ruoli più specifici del Maestro e dell'Ufficio amministrativo, Come è possibile notare in figura 1. Il flusso informativo tra le due pool è stato modellato tramite **message flows**.

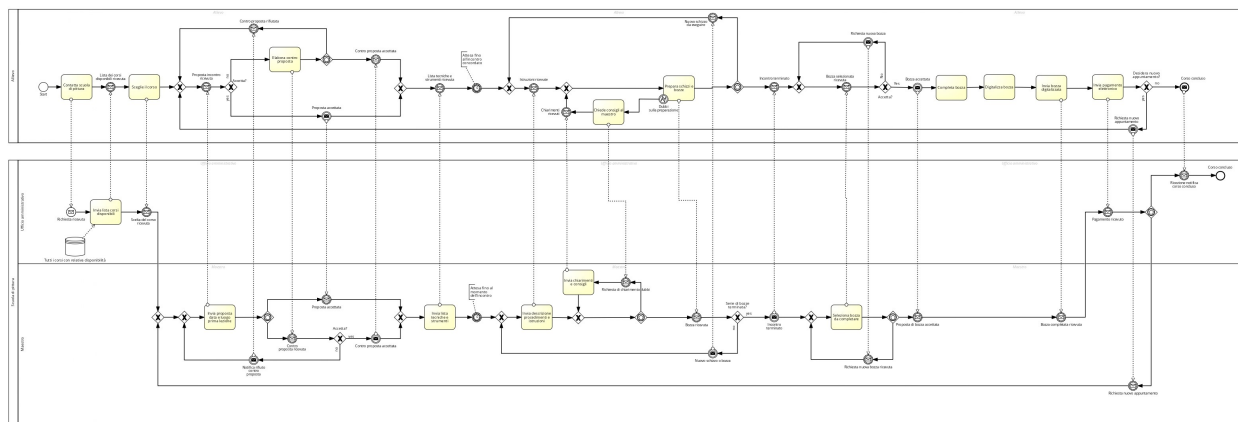


Figura 1: Diagramma BPMN rappresentante lo scenario in oggetto

### 2.1 Pool Allievo

Nella pool Allievo riportata in figura 2 il processo prevede un task iniziale che modella l'attività con cui il soggetto contatta la Scuola per ricevere la lista dei corsi disponibili. L'evento "Lista dei corsi disponibili ricevuta" è modellato come un **intermediate catching event**, dal momento che il processo può proseguire soltanto quando l'evento è stato intercettato. A seguito del task di scelta del corso, è presente uno **XOR gateway** di tipo join, utilizzato per unificare i diversi flussi. Una volta che l'Allievo ha ricevuto la proposta per l'appuntamento (**catching event**), la scelta che può compiere (accettare/non accettare) e che influenza le azioni successive, è stata modellata con una porta **XOR** in modalità split. Le alternative di scelta sono rappresentate sui rami e conducono a due diverse tracce: nel caso in cui l'Allievo accetti la proposta, allora il processo genererà un messaggio al Maestro (modellato come **intermediate throwing event**), altrimenti dovrà elaborare una controproposta e rimanere in attesa che arrivi il messaggio comunicante la risposta del Maestro. Tale situazione è stata modellata usando un **event-based gateway**, dal momento che la decisione è basata su un'informazione proveniente dall'esterno. Se la controproposta viene accettata il flusso si riunisce in una porta **XOR** di tipo join, altrimenti si torna indietro ed il processo rimane in attesa che arrivi una nuova proposta dal Maestro. Questa sequenza si ripete finché non viene trovato un accordo per l'appuntamento. Prima della data fissata, l'Allievo riceve una lista di tecniche e strumenti utili per le esercitazioni,

modellato come un **intermediate catching event**, quindi rimane in attesa del giorno dell'appuntamento come mostrato dall'**evento intermedio timer**. Durante l'incontro l'allievo prepara e mostra al maestro la bozza, ma solo dopo aver ricevuto prima una serie di istruzioni. Il task di preparazione schizzi e bozze è stato modellato con un **error event** annesso, per rappresentare la situazione in cui l'allievo, in presenza di dubbi, non riesca nel completamento della bozza. In tal caso il task viene completato solo se l'allievo chiede consiglio al maestro (**task**) e ottiene chiarimenti (**intermediate catching event**). Allora l'allievo può proseguire con le altre prove della serie di bozze fintanto che la serie non sarà esaurita, altrimenti riceve nuove istruzioni e procede con la bozza successiva. Si ritiene importante approfondire la scelta di usare l'event-based gateway per modellare la possibilità che l'incontro termini oppure debba essere eseguita una nuova prova: nello specifico, questa decisione si è rivelata necessaria per modellare nel modo più fedele possibile la possibilità che esista "una serie di prove" (come specificato nello scenario di input). Eseguita una singola bozza non è infatti detto che l'incontro termini, dal momento che il Maestro può decidere di fare altre prove (della serie di prove). Si suppone che la decisione di terminare l'incontro o eseguire una nuova bozza sia rimessa al Maestro e pertanto in questa pool Allievo il passaggio è stato modellato con un **event-based gateway**. Qualora l'incontro termini, l'Allievo riceve la comunicazione di fine incontro (**catching event**), riceve la bozza da completare selezionata dal maestro tra quelle prodotte durante l'incontro e decide se accettarla o meno. Tale decisione è stata modellata con una porta **XOR** in modalità split: se l'allievo non accetta può richiedere una nuova bozza (**intermediate throwing event**) e si ripete il ciclo di eventi tramite uno **XOR gateway** di tipo join, se invece accetta, allora procede alla comunicazione della decisione (**intermediate throwing event**) ed esegue una serie di task in sequenza relative alle operazioni di completamento, digitalizzazione ed invio della bozza selezionata. Infine, si procede al pagamento elettronico all'Ufficio amministrativo e, una volta inviato, l'allievo può decidere se fare richiesta per un nuovo appuntamento oppure concludere il corso. Questa eventualità è stata modellata con una porta **XOR** in modalità split: se l'Allievo non desidera un nuovo appuntamento allora si raggiunge l'**end event** che rappresenta il corso concluso, altrimenti viene richiesto un appuntamento e il flusso torna indietro per ricongiungersi nel primo **XOR join**, a significare che l'allievo rimane in attesa che il maestro comunichi la proposta di incontro, come descritto sopra.

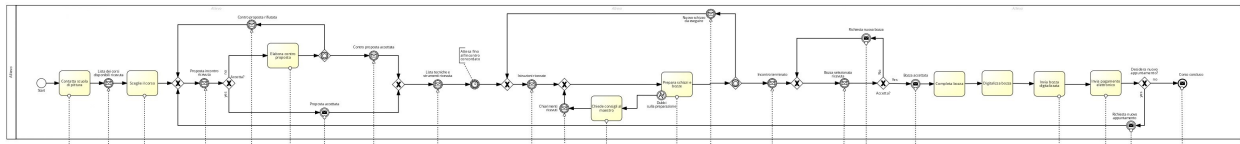


Figura 2: Diagramma BPMN della pool Allievo

## 2.2 Pool Scuola di pittura

La pool Scuola di pittura è articolata in una lane Ufficio amministrativo e una Maestro, ed il processo vede verificarsi il suo evento di **start** nella prima lane, quando l'Ufficio riceve la richiesta di un allievo e invia la lista di corsi disponibili. Quando la Scuola riceve la scelta del corso, modellato come un **intermediate event** di tipo catching, allora il processo si sposterà nella lane del Maestro che si occuperà di inviare la proposta della data e luogo della prima lezione, rappresentato da un **task**. Si noti che la comunicazione tra Ufficio e Maestro non è stata modellata dal momento che appartengono alla stessa pool. L'**event-based gateway** si propone di modellare le due possibili tracce del processo: se si verifica l'evento "Proposta accettata" allora il Maestro invia la lista di tecniche e strumenti di preparazione all'allievo, altrimenti riceverà la controproposta dell'Allievo e deciderà se accettare o no (**XOR gateway** tipo split). Se la controproposta è accettata allora il flusso si riunisce in una porta **join** e invierà la lista di tecniche da utilizzare, altrimenti il Maestro dovrà tornare sul task di invio proposta data e luogo, e questa sequenza continuerà finché i due attori non arriveranno ad un accordo. Una volta inviata la lista, l'attesa del giorno dell'incontro è stata ancora modellata con un **intermediate timer event**. Il giorno dell'incontro, il Maestro invia all'allievo una descrizione di procedimenti ed istruzioni per la preparazione della bozza (**task**), ed un **event-based**

**gateway** modella l'eventualità esterna che l'allievo abbia dubbi di preparazione. In tal caso il Maestro riceve la richiesta di chiarimento dubbi (**intermediate catching event**) ed invia consigli fino a che l'allievo non ha più dubbi e riesce nella preparazione della bozza, processo iterato tramite utilizzo di **XOR join** per convergere i flussi. Nel caso di preparazione bozza riuscita e ricevuta dal maestro, si incontra uno **XOR split** per decidere se la serie di bozze è terminata. Due **intermediate throwing event** modellano gli eventi successivi alla scelta del maestro: se la serie è terminata termina anche l'incontro, se il Maestro vuole far eseguire un'altra bozza si ripete il processo di invio descrizione procedimenti ed istruzioni, ed eventuali chiarimenti e consigli nel caso di dubbi dell'allievo. Una volta che il Maestro dichiara terminato l'incontro (**intermediate throwing event**), procede nella selezione della bozza da completare per l'allievo, modellato come task e seguito da un **event-based gateway** relativo alla scelta dell'allievo di accettare o meno la bozza. Nel primo caso il maestro riceve, tramite due **intermediate catching event** successivi, la comunicazione di proposta accettata e la stessa bozza completata da parte dell'allievo. In caso di rifiuto si ripete la sequenza di processo a partire dalla selezione di una nuova bozza. Al termine del processo l'Ufficio amministrativo riceve il pagamento, ma è possibile che arrivi la richiesta di un nuovo appuntamento (**event-based gateway** perchè la decisione viene dall'esterno) e in tal caso il flusso si riunisce in uno **XOR join** che permette al Maestro di inviare di nuovo la sua proposta, altrimenti il corso è da considerarsi concluso (**end event**).

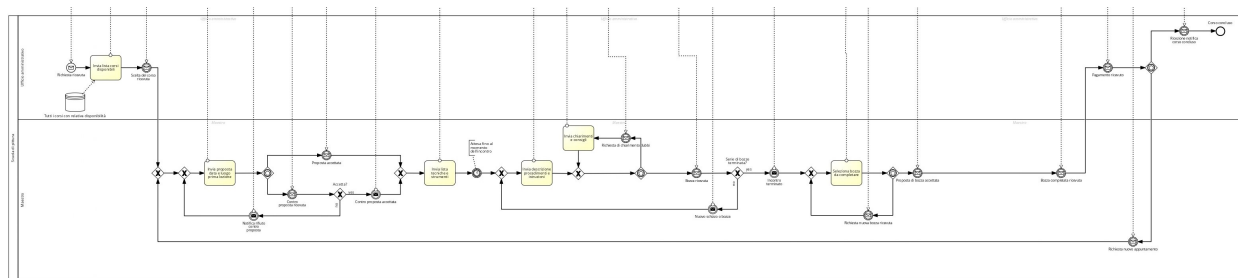


Figura 3: Diagramma BPMN della Scuola di pittura

### 3 Workflow nets

Una volta modellati i processi in BPMN, è stato possibile trasformarli in una particolare tipologia di reti di Petri, le workflow nets. Tale trasformazione consente di lavorare con una semantica più formale e abilita una serie di possibili tecniche di analisi. Nello specifico, la traduzione da BPMN a workflow nets prevede:

- sostituzione di ogni sequence flow arrow con una piazza;
- sostituzione di ogni task e ogni evento con una transizione;
- traduzione del gateway XOR con una piazza, seguita da tante transizioni quante sono le possibili alternative di scelta, collegate a loro volta a un'ulteriore piazza che rappresenta la condizione di input per il task da svolgere;
- sostituzione dell'Event-based gateway con una piazza collegata a tante transizioni quanti sono gli eventi che possono verificarsi e che abilitano una certa traccia di processo;
- traduzione dell'error intermediate event come una transizione alternativa al task a cui è annesso, seguito da una piazza e ulteriore transizione che si attiva al verificarsi dell'errore.

Nelle sezioni successive verranno prima analizzate separatamente le reti Allievo e Scuola di pittura, per poi andare a modellare ed analizzare per intero il processo collaborativo. Le immagini dei Coverability graphs sono riportati in Appendice A.

### 3.1 Workflow net Allievo

La rete progettata per Allievo con WoPeD, composta da 39 piazze, 44 transizioni e 88 archi, è riportata in figura 4.

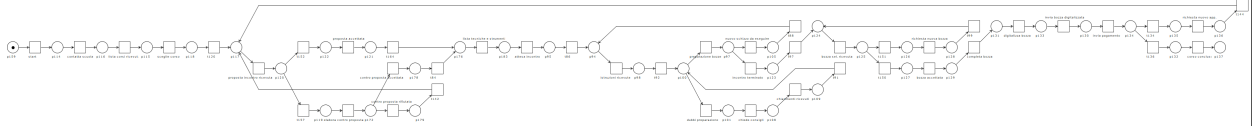


Figura 4: Workflow net Allievo

I risultati dell'analisi sono riportati in figura 5. In primo luogo occorre notare che è una workflow net dal momento che è possibile distinguere una piazza di input  $i$  e una di output  $o$ , con  $\bullet i = \emptyset$  e  $o \bullet = \emptyset$ , e in più ogni piazza/transizione appartiene ad un path che va da  $i$  fino  $o$ . Inoltre, i pre-set e post-set di ogni transizione sono costituiti da un'unica piazza: questo ci permette di dire che è una **S-net**, da cui consegue che è necessariamente **safe**, **sound** ed anche **bounded**. La rete risulta **strongly-connected** e  $M_0(P) = 1$ , quindi la workflow net è **live**; è inoltre una **free-choice** net dal momento che per ogni coppia di transizioni i loro pre-sets sono disgiunti o uguali. Essendo free-choice, live e bounded, allora è **S-coverable** e come si nota in figura esiste un unico **S-component**. Non esistono **PT-handles** e **TP-handles** pertanto la rete è da definirsi **well-structured**.

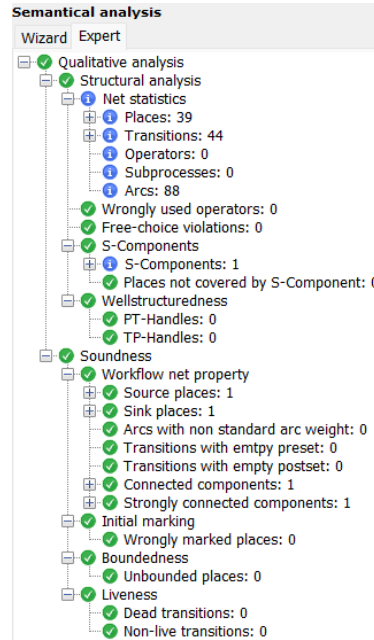


Figura 5: Analisi semantica della workflow net Allievo

Infine, è stato analizzato il rispettivo coverability graph (Fig. 13, in Appendice) ottenuto anch'esso tramite *WoPeD*. Esso coincide con il reachability graph, dal momento che quest'ultimo è finito (la workflow net è bounded) e comprende 44 archi e 39 vertici. Il vantaggio nell'realizzare anche questo grafo deriva dal fatto che, quando risulta finito, è in grado di rappresentare esattamente tutti i possibili comportamenti della wf net progettata.

### 3.2 Workflow net Scuola di pittura

Come nel caso di Allievo, anche la rete Scuola di pittura è stata realizzata e analizzata con *WoPeD* (Fig. 6) e i risultati dell'analisi sono riportati in figura 7.

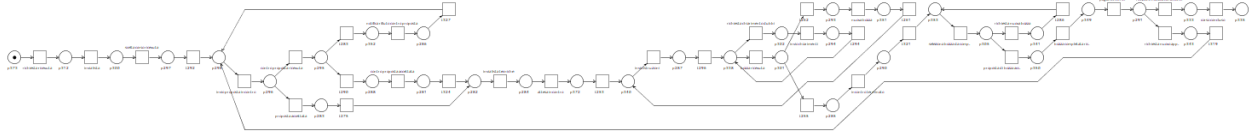


Figura 6: Workflow net Scuola di pittura

La rete è una workflow net costituita da 34 piazze, 39 transizioni e 78 archi, e valgono tutte le considerazioni fatte nella sezione precedente per Allievo: è una **S-net** quindi **safe**, **sound** e **bounded**, è **strongly-connected** con  $M_0(P) = 1$ , quindi **live**. È una **Free-choice net**, **S-coverable** con un unico **S-component** e **well-structured**. Anche in questo caso è interessante realizzare il coverability graph, coincidente con il reachability graph, che comprende 39 archi e 34 vertici (Fig. 14).

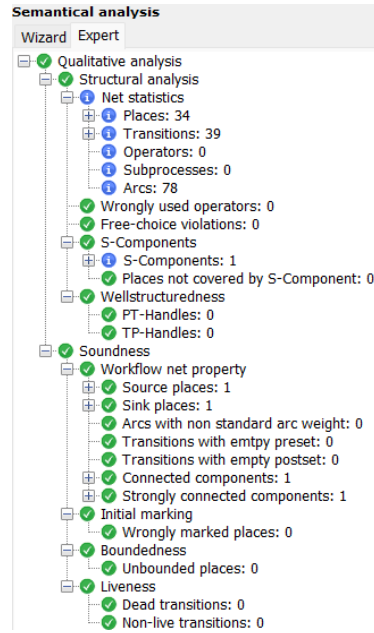


Figura 7: Analisi semantica della workflow net Scuola di pittura

### 3.3 Workflow net completa

I due workflow modules sono stati combinati per avere una rappresentazione completa dell'intero processo in oggetto. In particolare, sono state aggiunte delle piazze di interfaccia per permettere lo scambio di flussi informativi tra gli attori, come i message flows nella modellazione BPMN. Anche in questo caso la workflow net è stata realizzata e analizzata con *WoPeD*. L'unica piazza iniziale è quella che era presente in Allievo, che decide di contattare la scuola, e che rappresenta l'evento che genera le nuove istanze dell'intero processo.

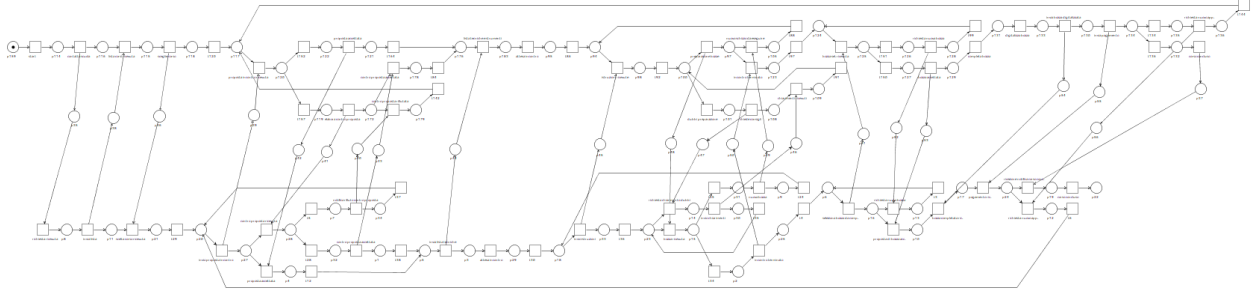


Figura 8: Workflow net completa

L'analisi della rete mostra che ha 93 piazze, 83 transizioni e 208 archi e risulta essere **sound**. Dal momento che l'aggiunta delle piazze di interfaccia per la modellazione del flusso informativo comporta la presenza di transizioni con pre-set non disgiunti o non uguali, la rete non risulta una **Free-choice net**. Non è neanche una **S-net**, dal momento che le transizioni non avranno più un post-set costituito da un'unica piazza. La wf net è **S-coverable**. Si nota inoltre che, a differenza delle due reti singole, in questo caso esistono **PT-handles** e **TP-handles** e quindi la rete non è **well-structured**. Proseguendo l'analisi, si può notare che la rete risulta essere **strongly-connected**, **bounded** e **live**.

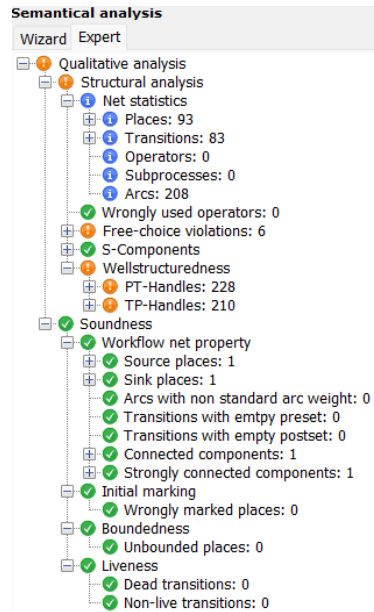


Figura 9: Analisi semantica della workflow net completa

Anche in questo caso il Coverability graph coincide con il Reachability graph dal momento che si tratta di una rete bounded (Fig. 15).

## 4 Variante del processo

Lo scenario in oggetto prevede la seguente variante del processo finora descritto ed analizzato: *al termine del corso l'Allievo può scegliere se iniziare un nuovo percorso di apprendimento*. Lo sviluppo della variante è stato realizzato mediante una modifica nei diagrammi BPMN e di conseguenza nelle reti di Petri, come si può osservare dalle figure 10 ed 11. Nello specifico, nella Pool Allievo, dopo l'evento throwing con cui l'attore notifica alla scuola che il corso è concluso, è stata aggiunta una porta **XOR** in modalità split, per modellare



la possibilità di intraprendere un nuovo percorso. Qualora l'Allievo desideri intraprendere il nuovo percorso tale volontà viene comunicata alla scuola e il sequence flow rientrerà nello **XOR join** iniziale, da cui poi l'Allievo potrà scegliere un nuovo corso (la Scuola non invierà di nuovo la lista dei corsi disponibili). Nel caso in cui invece l'Allievo non desideri iniziare un nuovo percorso, il processo giungerà alla fine (**end event**). Anche la pool Scuola di pittura è stata modificata: dopo la ricezione della notifica del corso concluso, è stato inserito un **event-based gateway**, che si propone appunto di modellare la possibilità che l'Allievo (ora soggetto esterno) desideri iniziare un nuovo percorso. Se l'evento che si verifica è "Notifica nuovo percorso", allora il sequence flow si riunisce nello **XOR** iniziale in modalità join e il processo attende che arrivi la comunicazione dall'Allievo circa la scelta del nuovo corso, altrimenti il processo è concluso.

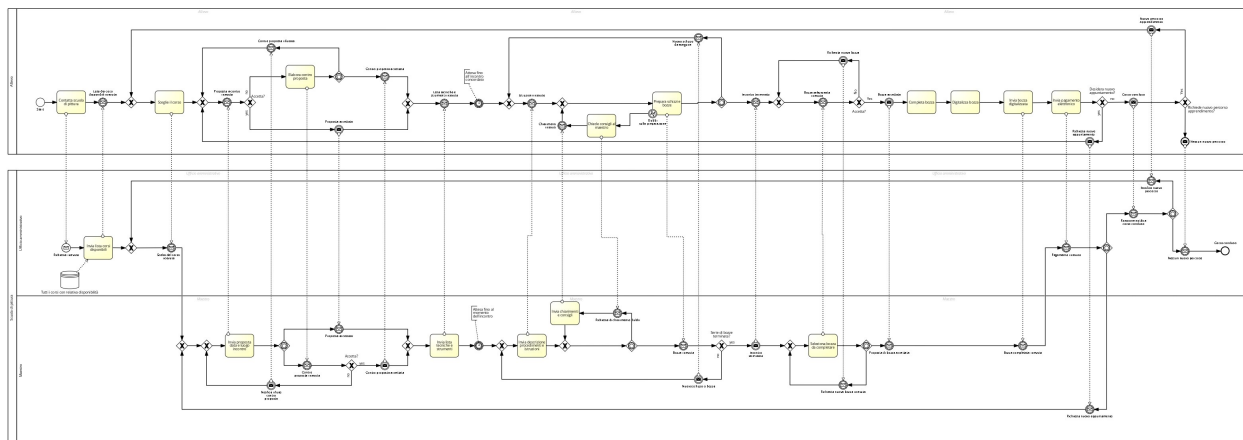


Figura 10: Diagramma BPMN del processo con variante descritto nello scenario in oggetto

Per quanto riguarda l'analisi del processo, dapprima si è tradotto in una workflow net utilizzando le stesse tecniche di cui sopra (Fig. 11), e poi analizzato per mezzo del tool *WoPeD* (Fig.12). Per questa analisi valgono tutte le considerazioni fatte nella sezione precedente per la wf net complessiva senza variante.

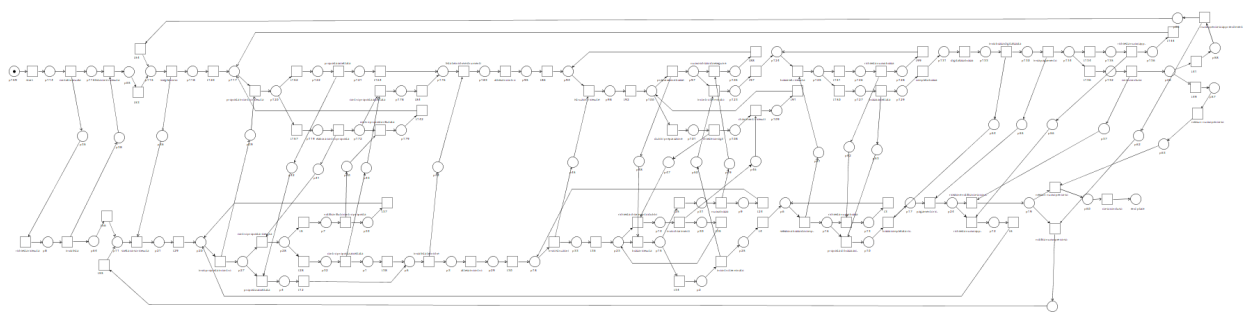


Figura 11: Workflow net variante

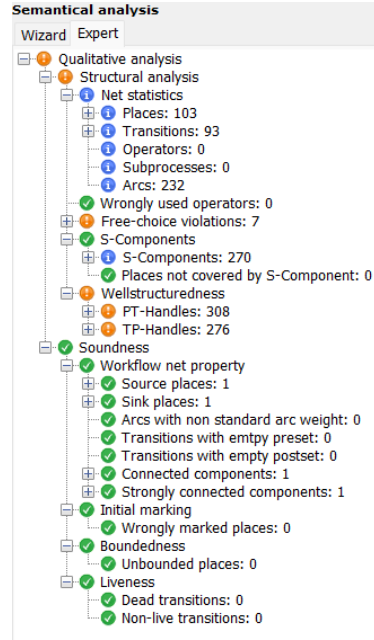


Figura 12: Analisi semantica della workflow net variante

## A Appendice

In questa Appendice vengono riportate le figure dei **Coverability graphs** a pagina intera che per motivi di leggibilità dell'elaborato si è deciso di non inserire nelle relative sezioni. Nello specifico, sono presenti, in ordine, i grafi della rete Allievo, Scuola di pittura, delle rete complessiva e della rete con variante.

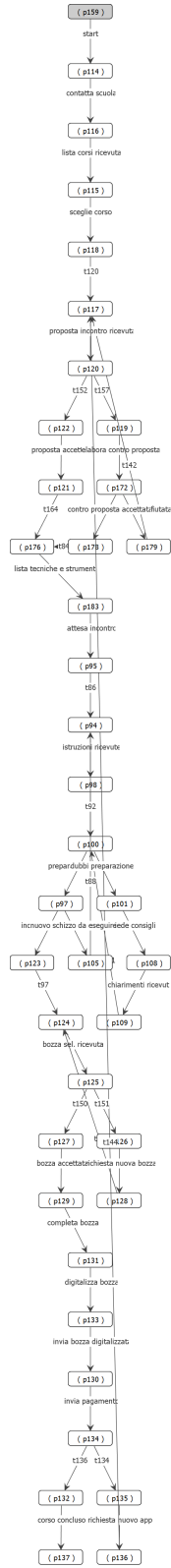


Figura 13: Coverability graph della workflow net Allievo

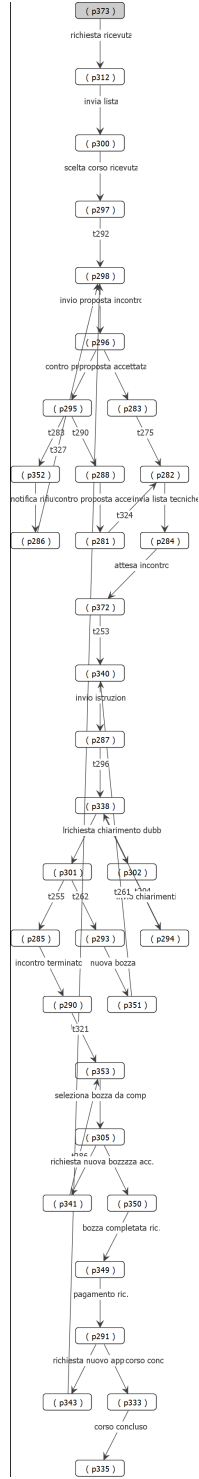


Figura 14: Coverability graph della workflow net Scuola di pittura

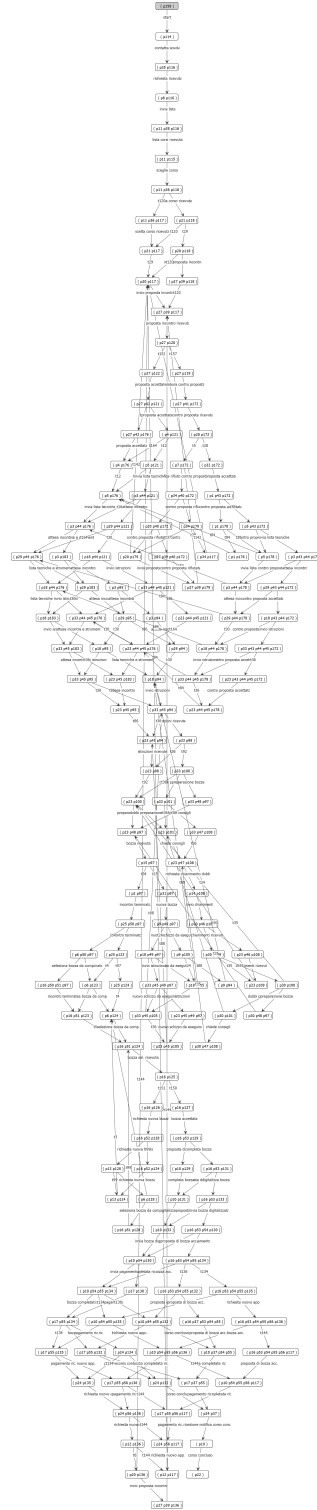


Figura 15: Coverability graph della workflow net complessiva



Figura 16: Coverability graph della workflow net variante