



**UNIVERSITÀ DEGLI STUDI DI ROMA
TOR VERGATA**

MACROAREA DI SCIENZE MATEMATICHE, FISICHE E
NATURALI

CORSO DI LAUREA IN INFORMATICA

A.A. 2019/2020

Tesi di Laurea

Strumenti MSaaS per l'analisi simulativa di processi di
business

RELATORE

Prof. Andrea D'Ambrogio

CANDIDATO

Ionut Jaraveti

Indice

Introduzione	1
1 Business Process	3
1.1 Definizione di business process	3
1.2 Business Process Management	5
1.3 Business Process Modeling	7
1.4 Business Process Simulation	9
2 Business Process Modeling Notation	12
2.1 Definizione di BPMN	12
2.1.1 Elementi di BPMN	14
2.2 PyBPMN	18
3 L'applicazione web ViewSimulation per piattaforme MSaaS	19
3.1 eBPMN	19
3.2 MSaaS	23
3.3 ViewSimulation	23
3.3.1 Requisiti	23
3.3.2 Architettura	24
4 Caso di Studio	29
4.1 Descrizione Emergency Attendance	29
4.2 Mapping del processo	31
4.3 Annotazione del modello BPMN	33

4.4 Simulazione di Emergency Attendance	34
5 Conclusioni	37
6 Ringraziamenti	39
Elenco delle figure	40

Introduzione

Ogni organizzazione possiede dei processi di business. A causa della crescente complessità delle loro attività interne, la pianificazione e la gestione dei processi e delle risorse rappresentano degli elementi di massima importanza.

Negli ultimi decenni tutto questo ha portato ad un cambiamento di strategie da parte di molte aziende. Queste strategie riguardano la gestione dei processi e prevedono l'applicazione di nuovi metodi e strumenti in grado di ottenere una pianificazione e una gestione migliore del proprio business.

In questo contesto, il Business Process Management rappresenta l'insieme delle attività necessarie a definire, ottimizzare, monitorare ed integrare i processi aziendali al fine di rendere efficace l'operato dell'azienda.

Questa disciplina globale fa uso di metodi, tecniche e strumenti per supportare la progettazione, l'analisi, la promulgazione e la diagnosi dei business process.

Differenti aspetti innovativi hanno stimolato la ricerca scientifica verso tecniche di gestione dei processi aziendali sempre più avanzate, ovvero verso la creazione di applicazioni software in grado di gestire in maniera organizzata questi processi.

Attualmente sul mercato esistono tanti software progettati per aiutare i business analysts nella gestione dei processi aziendali, ma spesso questi software sono molto limitati e soprattutto non offrono un buon livello di flessibilità nella progettazione e modellazione dei processi.

Il presente documento ha come scopo la presentazione di tecniche e strumenti

di **MSaaS** ossia di **Modeling & Simulating as a Service**. Questo paradigma è caratterizzato da tanti aspetti positivi:

- maggiore flessibilità e libertà nella creazione dei modelli di processo;
- maggiore scalabilità e capacità di adattamento ai cambiamenti del mercato;
- la possibilità di analisi del processo sia nella fase di progettazione sia in quella operativa;

L'analisi di un **business process** utilizzando questo approccio si basa sullo standard internazionale **BPMN - Business Process Model and Notation** e su altri concetti che verranno introdotti successivamente.

Il SEL - Software Engineering Laboratory - gruppo di ricerca dell'Università di Tor Vergata, guidato dal Prof. Andrea D'Ambrogio, ha realizzato un progetto in cui sono state applicate le tecniche M&S (Modeling & Simulation) per l'analisi dei processi di business. D'ora in avanti chiameremo questo progetto "Simulatore".

Per raggiungere tali obiettivi si è utilizzato il Simulatore, appena introdotto.

Si è deciso quindi di sviluppare una Applicazione Web che, utilizzando le funzionalità del Simulatore, eseguisse la simulazione di alcuni modelli realizzati all'interno dello stesso progetto e che fosse in grado di visualizzare i risultati e le statistiche di notevole interesse per quanto concerne la valutazione dei processi aziendali. Inoltre, è stata predisposta la possibilità che l'utente (un business analyst, un progettista ovvero un professionista del settore) modifichi alcuni parametri fondamentali ai fini della simulazione in oggetto per ulteriori valutazioni delle statistiche.

Capitolo 1

Business Process

In questo Capitolo viene introdotto il concetto di processo di business e vengono presentate delle tecniche e degli strumenti utilizzati per la gestione dei processi di business.

1.1 Definizione di business process

Un **business process** oppure un processo aziendale può essere definito come un insieme di compiti che sono eseguiti in modo coordinato in un ambiente tecnico ed aziendale per raggiungere degli obiettivi aziendali.

Questi obiettivi possono essere la consegna di prodotti, di informazioni, di servizi oppure una combinazione di questi elementi.

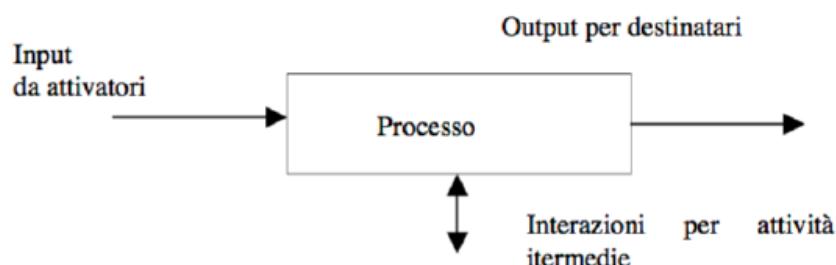


Figura 1.1: Descrizione di un processo

Un processo deve prevedere **input** chiaramente ben definiti, costituiti da tutti i fattori che contribuiscono direttamente o indirettamente al valore aggiunto del prodotto finale che rappresenta l'**output**. La trasformazione dell'input in output può essere eseguita con l'impiego di lavoro umano, di macchine o di entrambi.

L'output di un processo è destinato a clienti che possono essere esterni all'azienda oppure interni. Da questo punto di vista si distinguono:

- *processi primari* che hanno come clienti soggetti esterni all'azienda;
- *processi di supporto* che hanno come clienti soggetti interni all'azienda e quindi supportano i processi primari ovvero non creano di per sé un valore aggiunto all'output riconosciuto dal cliente.

I processi possono essere classificati in processi direzionali, processi di gestione e processi operativi.

- i processi direzionali sono caratterizzati da decisioni non strutturate finalizzate alla pianificazione di medio-lungo termine dell'organizzazione.
- i processi di gestione regolano il funzionamento del sistema operativo di una particolare azienda e controllano il raggiungimento degli obiettivi.
- i processi operativi costituiscono le attività principali di un'azienda che concorrono al raggiungimento degli obiettivi.

Nelle aziende dotate di un sistema di gestione della qualità, i processi aziendali devono essere misurabili e monitorabili nel tempo mediante l'utilizzo di indicatori di prestazione chiave. Le *KPI (Key Performance Indicator)* sono degli indicatori quantificabili di efficacia e di efficienza di un processo.

1.2 Business Process Management

Il **Business Process Management** è l'insieme delle attività necessarie a definire, ottimizzare, monitorare ed integrare i processi aziendali al fine di rendere efficace il business dell'azienda.

I principi fondamentali del Business Process Management sono la rappresentazione esplicita dei processi aziendali con le loro attività e i vincoli di esecuzione tra essi. Una volta che i processi sono definiti, essi possono essere sottoposti ad azioni mirate di analisi, di miglioramento e di promulgazione.

In questo contesto, viene fatto uso di metodi, tecniche e strumenti per supportare la progettazione, l'analisi, la promulgazione e la diagnosi dei business process operativi.

In figura 1.2 (estratta da [10]) si possono osservare le varie fasi che caratterizzano il **BPM**: Progettazione, Modellazione, Esecuzione, Monitoraggio e Ottimizzazione.

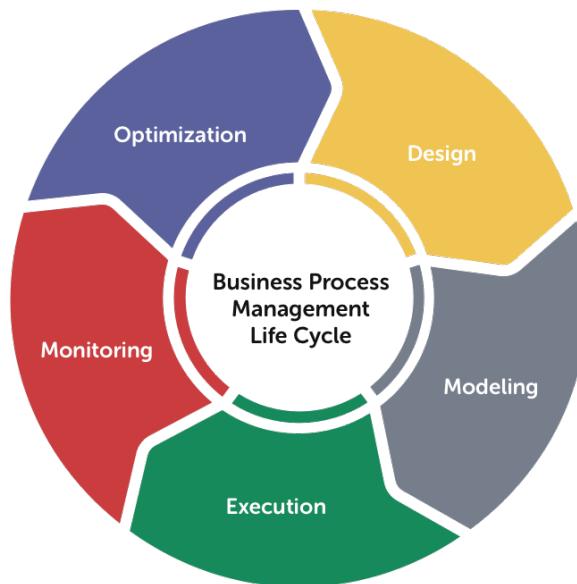


Figura 1.2: Business Process Management Lifecycle

Poichè i processi operativi interessano variabili quantitative e sono ripetuti su grandi volumi quotidianamente, essi sono adatti all'automazione, a differenza dei processi di carattere decisionali che utilizzano la tecnologia come un supporto che difficilmente può sostituire l'attività umana.

Il metodo per gestire un processo aziendale può essere sintetizzato nel seguente modo:

- identificazione del processo aziendale;
- definizione degli attori del processo;
- definizione degli input e degli output scambiati tra gli attori introdotti;
- definizione delle attività e delle procedure che regolano lo svolgimento del processo;
- analisi della durata delle attività definite;
- definizione delle prestazioni attese da quel processo aziendale;
- definizione delle responsabilità del processo aziendale.

Negli ultimi anni questa disciplina ha ricevuto una notevole attenzione sia dalla comunità di amministrazione aziendale sia da quella informatica. Ovviamente i membri di queste due comunità sono caratterizzati da diversi interessi ed aspetti educativi. Le persone coinvolte nell'amministrazione aziendale sono interessate a migliorare l'operatività dell'organizzazione, incrementare la soddisfazione dei propri clienti, riducendo costi del business e fornire nuovi prodotti e servizi ad un basso costo. Per quanto riguarda l'informatica, i ricercatori studiano le proprietà strutturali dei processi mentre gli ingegneri software sono interessati a fornire sistemi software scalabili e robusti per la realizzazione tecnica dei processi aziendali.

I software dedicati dovrebbero velocizzare e semplificare la gestione e il miglioramento dei processi di business. Per soddisfare queste condizioni, un tale software deve monitorare l'esecuzione dei processi, consentire ai manager di fare analisi e cambiare tecnologia e organizzazione sulla base di dati concreti e non in base ad opinioni

soggettive.[12]

Poichè esiste uno stretto legame tra processi e profitabilità aziendali, vi è una sempre maggiore esigenza di incrementare il valore delle attività riducendo allo stesso tempo le risorse impiegate per svolgerle, di ridurre i costi legati ai sistemi informativi aziendali e di sottoporre i processi aziendali a continui controlli di qualità.

Integrando tra loro i processi di business in modo da poterli monitorare ed ottimizzare secondo un orientamento univoco agli obiettivi aziendali, si sfruttano appieno le potenzialità di un sistema di **Business Process Management**, ottenendo una sostanziale riduzione dei costi e dei tempi di implementazione degli stessi processi.

Inoltre, il **BPM** può aiutare a migliorare le prestazioni aziendali grazie alla riduzione dei costi interni, legati alla propria attività, all'aumento di flessibilità, al raggiungimento di tempi di consegna più rapidi e soprattutto migliorando le relazioni con gli stakeholder interni ed esterni.

Alla luce di questi vantaggi si può affermare che il **BPM** rappresenta un ottimo strumento per la gestione dei processi di quelle organizzazioni che hanno come obiettivo l'aumento della soddisfazione della propria clientela, la misurazione delle performance aziendali e la maggiore attenzione e controllo per le risorse impiegate.

1.3 Business Process Modeling

La decisione di modellare deriva dalla necessità di simulare la realtà. L'utilità di un modello risiede nel fatto che esso rappresenti uno strumento utile per gestire la complessità.

Un modello rappresenta un certo soggetto, rispetto ad un particolare scopo, mediante l'astrazione dai dettagli che sono irrilevanti per quello scopo.

La modellazione di un processo è comunemente basata sulla rappresentazione di una singola istanza di processo, attivata da un evento. Ciascuna istanza di processo ha dati propri e richiede lo svolgimento delle attività che costituiscono il processo.

Si possono identificare due scopi principali per la modellazione:[5]

- *Organizational design* (la progettazione organizzativa) in cui il modello è utilizzato per una completa comprensione del business process durante il suo ciclo di vita, dalla comunicazione con le parti interessati fino alla valutazione e la perfezione del processo.
- *Application development* (sviluppo dell'applicazione) in cui il modello è più dettagliato e ciò include tutte le informazioni tecniche richieste per l'implementazione e l'automatizzazione del business process.

La progettazione è la fase iniziale nello sviluppo di un processo. Questa fase permette di descrivere un processo formalmente.

La progettazione formale di un modello di processo rende non ambigua la definizione del processo e questo significa che durante la sua implementazione non vi sia spazio per interpretazioni personali per quanto riguarda la sua parte operativa.

Esistono diverse sintassi che si possono utilizzare per descrivere un processo:

1. **Flowchart**: un formalismo costituito da un insieme di rettangoli e linee. Non ha una sintassi molto ricca.
2. **Event Process Chain Diagrams (EPC)**: un grafo ordinato di eventi e funzioni in cui ogni attività all'interno del processo è attivata da un evento.
3. **Business Process Execution Language (BPEL)**: un linguaggio basato su xml che permette di interagire con altri servizi.
4. **Petri nets (PN)**: un linguaggio matematico di modellazione per sistemi distribuiti, la versione grafica può essere utilizzata per rappresentare i modelli di processo.[11]

5. **Business Process Model and Notation (BPMN)**: lo standard internazionale per la rappresentazione dei business process, sintassi che verrà descritta nel prossimo capitolo.

La mappatura o la rappresentazione dei processi può avvenire in due ottiche diverse:

- la situazione attuale, detta **as-is**;
- la situazione futura desiderata, detta **to-be**.

Queste sono due attività di analisi nettamente distinte, che portano a definire i miglioramenti necessari per passare dai processi rilevati nell' **as-is** a quelli formalizzati nel **to-be**.

Alcuni possibili vantaggi associati alla modellazione dei processi di business:

- i modelli vengono utilizzati come strumento visuale per il miglioramento e la reingegnerizzazione dei processi;
- è possibile avere delle misure di prestazioni dei processi;
- i processi di business possono essere automatizzati;
- anche le misure possono essere calcolate, in modo automatizzato, dagli strumenti per l'esecuzione dei processi.

1.4 Business Process Simulation

Business Process Simulation (BPS) è considerata la tecnica più popolare e più utilizzata per l'analisi dei modelli di processo.[8]

Valutare l'effetto di un cambiamento nei processi aziendali con precisione e rapidità, avere la comprensione e la stima del tempo e del costo necessari al completamento di un processo, sono diventati fattori molto importanti per la crescita del business di

un'organizzazione e vengono impiegati dal management per pianificare al meglio l'attività e gli obiettivi aziendali.

Quest'analisi risulta molto complicata soprattutto a causa della complessità dei compiti e delle numerose relazioni che esistono o che si vanno a creare fra le varie risorse disponibili durante l'esecuzione del processo. Tutto questo comporta che l'utilizzo di strumenti analitici risulta insufficiente e inadeguato per affrontare la la complessità e la dinamicità dei sistemi nei quali un'azienda è costretta ad operare.

Differentemente, effettuare una simulazione tramite l'utilizzo di sistemi e software informatici abbinata alla tipica semplicità dei flowchart rappresenta un metodo molto efficace e uno strumento molto potente.

Questo strumento offre la possibilità di valutare soluzioni alternative prima di effettuare un grande impiego di risorse e tempo.

La capacità di visualizzare come un processo possa svolgersi misurandone le performance e valutandone le eventuali variazioni rende la simulazione lo strumento perfetto, strettamente necessario nel processo decisionale e gestionale di un'azienda.

Il motore di simulazione può essere realizzato generalmente utilizzando due approcci diversi:

- un linguaggio che fornisce primitive specifiche per la simulazione. Questo approccio fornisce più flessibilità richiede più skills ed un maggiore effort;
- un package di simulazione, o un tool, che fornisce diversi blocchi predefiniti specifici del dominio che vengono utilizzati per produrre ed eseguire il modello di simulazione. Questo approccio è meno flessibile ma più user-friendly e generalmente fornisce un'interfaccia grafica per la definizione del modello di simulazione. [5]

La **BPS** ha la capacità di cogliere i limiti delle risorse che si hanno a disposizione, il processo decisionale e logico e l'imprevedibilità della realtà nella quale l'azienda

svolge il proprio business. Durante la simulazione il modello di un processo effettua le operazioni dell'attività modellata. Questo viene realizzato eseguendo passo dopo passo gli eventi definiti dal modello rispettando le tempistiche e i vincoli che questi eventi richiedono nel mondo reale.

La simulazione fornisce statistiche riguardanti gli elementi appartenenti al modello. In questo modo è possibile valutare le performance di un processo analizzando i dati prodotti come output dalla simulazione.

L'utilità della simulazione risiede nel fatto che essa costituisce una sorta di laboratorio dove poter eseguire esperimenti per comprendere sino a fondo il comportamento di un processo all'interno di un sistema.

Questi esperimenti sono fondamentali per anticipare possibili risposte del sistema e sviluppare così nuovi interventi per governare meglio le realtà complesse.

Alcuni aspetti di **Business Process Simulation** sono anche meno positivi:

- la creazione di un modello di simulazione potrebbe sollecitare alcuni skills di programmazione;
- l'esattezza e la completezza del modello di processo di business determinano l'affidabilità e la solidità dei risultati della simulazione;
- il numero delle simulazioni effettuate può influire sulla precisione dei risultati, arrivando in questo modo ad elaborare simulazioni difficili ed impegnative.

Capitolo 2

Business Process Modeling Notation

In questo Capitolo viene descritto lo standard de facto per la modellazione dei processi di business, ossia il linguaggio BPMN - Business Process Modeling Notation e l'estensione PyBPMN che arricchisce un modello BPMN con annotazioni molto importanti ai fini della simulazione .

2.1 Definizione di BPMN

Il **Business Process Modeling Notation (BPMN)** è una notazione di modellazione per processi di business, definita dall'Object Management Group (OMG).[9]

La notazione nasce dall'esigenza di creare un linguaggio di modellazione che sia in grado di eliminare il gap tecnico esistente tra le descrizioni dei processi per mezzo di diagrammi di flusso e le descrizioni di queste ultime in un linguaggio di esecuzione. Grazie alla sua notazione ricca e facile da comprendere, il BPMN è diventato uno standard nel settore del *Business Process Management*.

La prima versione nasce nel 2006 con l'obiettivo di fornire una sintassi comprensibile facilmente da tutti gli attori coinvolti nella gestione dei processi: dagli analisti che definiscono i processi, dagli sviluppatori che devono implementare i processi e dagli amministratori del business che devono gestire e controllare i processi.

Infatti questo standard permette ai business analysts di specificare modelli astratti di processi di business che sono sucessivamente mappati nei linguaggi di esecuzione dei sistemi del *BPM*, come il *Business Process Execution Language (BPEL)*.

Nonostante il linguaggio permetta di rappresentare efficientemente un *business process* a diversi livelli di astrazione, viene più utilizzato nelle prime tappe (analisi e progettazione) del ciclo di vita del processo.[5]

Un modello BPMN fornisce 3 tipologie diverse di diagrammi per la descrizione grafica dei processi di business:

- Processi (Orchestrazioni) che si dividono in:
 - *Private Business Processes* sono i processi interni ad una specifica organizzazione, chiamati anche workflow, il cui flusso di esecuzione è ben definito per ciascun partecipante.
 - *Public Business Processes* rappresentano l'interazione fra un processo privato e un altro processo o partecipante e specificano gli oggetti o le informazioni scambiate tra i vari partecipanti. In questo modo i dettagli delle attività del processo privato non vengono mostrati e quindi i processi privati vengono utilizzati come delle "black boxes".
- Coreografie
 - Una coreografia è la definizione del comportamento previsto tra due o più partecipanti di business il cui obiettivo è quello di mostrare l'interazione concentrandosi sul flusso dei messaggi.
- Collaborazioni che possono includere processi e coreografie
 - Una collaborazione descrive le interazioni tra due o più entità aziendali contenute in Pools. L'interazione avviene attraverso il Message Flow che

permette la comunicazione tra due o più Pools. Solitamente una collaborazione rappresenta la connessione tra due Public Processes in cui le Activities sono i punti di contatto.

I diagrammi presentati sono definiti **token based**. Il token è un concetto teorico utilizzato per descrivere il comportamento di un business process durante la sua esecuzione.^[5] Tale token attraversa i Sequence Flows e passa tramite gli elementi del diagramma. In questo modo viene descritto ciò che avviene in ogni elemento attraversato dal token. Uno Start Event genera un token che deve essere consumato da un End Event dopo aver percorso la rete di Sequence Flows, Gateways e Activities durante l'esecuzione del processo.

2.1.1 Elementi di BPMN

Gli elementi di un diagramma BPMN si dividono in 4 categorie:

1. **Flow Objects** - sono gli elementi di flusso del processo (Figura 2.1)
2. **Artifacts** - vengono utilizzati per specificare informazioni e documentazione addizionale sui processi di business (Figura 2.2)
3. **Connecting Objects** - vengono utilizzati per collegare tra loro i Flow Objects, Artifacts e le Swimlanes (Figura 2.3)
4. **Swimlanes** - vengono utilizzati per organizzare i partecipanti nei processi e le loro attività (Figura 2.4)

Nome	Descrizione	Notazione Grafica
Event	Viene rappresentato con un cerchio e denota qualcosa che accade durante un processo. Le icone all'interno del cerchio indicano il tipo di evento. Ci sono 3 tipi di eventi, a seconda di quando interagiscono nel flusso del processo: Start, Intermediate e End.	<p>Start Intermediate End</p>
Activity	Viene rappresentata con un rettangolo con angoli arrotondati e descrive il tipo di lavoro che deve essere svolto. Un Activity può essere atomica o non-atomica. I Tipi di Activities che sono parte di un modello di processo sono: Sub-Process e Task.	
Gateway	Viene rappresentato con un diamante ed è utilizzato per controllare la divergenza e la convergenza dei Sequence Flows in un processo. Questo permette di determinare diverse tipologie di flusso. Exclusive Gateway crea flussi alternativi in un processo in cui solo uno dei percorsi può essere scelto. Inclusive Gateway crea flussi alternativi in cui vengono valutati tutti i percorsi. Parallel Gateway crea percorsi paralleli senza valutare tutte le condizioni.	<p>Exclusive Inclusive Parallel</p>

Figura 2.1: Tabella Flow Objects

Nome	Descrizione	Notazione Grafica
Data Object	Non hanno alcun effetto sul flusso del processo. Forniscono informazioni su cosa richiedono le Activities per essere eseguite oppure cosa esse producono. I Data Objects possono rappresentare un oggetto singolo o una collezione di oggetti.	
Message	Questo elemento viene utilizzato per rappresentare i contenuti di una comunicazione fra due partecipanti del diagramma.	
Group	Un gruppo rappresenta un insieme di elementi appartenenti ad una singola categoria. Le categorie non hanno nessun effetto sul flusso del processo, esse sono utilizzate soltanto per scopi di documentazione o di analisi quindi i gruppi rappresentano l'unico modo per visualizzarle all'interno del processo.	
Text Annotation	Le annotazioni testuali sono un meccanismo per il modellista per fornire ulteriori informazioni a chi legge il diagramma BPMN.	

Figura 2.2: Tabella Artifacts

Nome	Descrizione	Notazione Grafica
Sequence Flow	Un flusso di sequenza viene utilizzato per mostrare l'ordine con cui vengono eseguite le attività all'interno del processo.	A solid horizontal arrow pointing to the right.
Message Flow	Un flusso di messaggio viene utilizzato per mostrare il flusso di messaggi tra due partecipanti che sono preparati a spedirli e riceverli.	A dashed arrow starting from a small circle on the left and ending with a triangle on the right.
Association	Viene utilizzata per connettere informazioni e artefatti con gli elementi grafici di BPMN. Le frecce nelle associazioni indicano la direzione nella quale l'informazione si propaga.	A dotted horizontal line with a triangle at the end.

Figura 2.3: Tabella Connecting Objects

Nome	Descrizione	Notazione Grafica
Pool	Un pool viene utilizzato per rappresentare graficamente un partecipante in un processo. Inoltre serve anche da contenitore grafico per partizionare un insieme di attività da altri pool.	A rectangle divided vertically into two sections. The left section is labeled "Name".
Lane	Un lane viene utilizzato come una sotto-partizione all'interno del pool che si estende per l'intera lunghezza di quest'ultimo sia in orizzontale che in verticale. Inoltre viene utilizzato per organizzare e categorizzare le attività.	A rectangle divided vertically into three sections, each labeled "Name".

Figura 2.4: Tabella Swimlanes

Il **BPMN** serve a modellare solo i processi. Non è utile per la rappresentazione di altri aspetti come le strutture organizzative, le scomposizioni funzionali, i modelli di dati, le regole di business e le strategie dell'organizzazione.

2.2 PyBPMN

Il linguaggio **BPMN** ha delle limitazioni per quanto riguarda la specificazione delle proprietà non funzionali dei *business process*. Per rimediare a questo problema, alcuni autori hanno esteso la definizione standard del BPMN.

PyBPMN (Performability-enabled BPMN) è una estensione BPMN che si rivolge alla specificazione delle proprietà di performance e affidabilità dei processi di business.[1, 2, 6, 7]

L'estensione si basa su un approccio che sfrutta i principi e gli standard introdotti dalla Model Driven Architecture (MDA). In particolare essa viene realizzata inizialmente specificando il metamodello BPMN e poi il metamodello PyBPMN si ottiene aggiungendo le metaclassi che definiscono le caratteristiche della performance e della reliability.

PyBPMN ricopre 4 aree principali delle proprietà non funzionali:

- Workload, per modellare il carico di lavoro dei compiti nei processi (la classe *GaWorkloadEvent*);
- Performance, per specificare le proprietà di performance e le proprietà di rendimento come service time, associate ai singoli compiti (la classe *PaQualification*);
- Reliability, per esprimere le proprietà di affidabilità delle risorse utilizzate dalle attività per portare a termine le richieste (la classe *DaQualification*);
- Resource management per specificare le risorse coinvolte nei processi.

PyBPMN permette agli utenti di specificare risorse atomiche (PyPerformer), gruppi di risorse (PySubsystem) o insiemi di risorse alternative.

Capitolo 3

L'applicazione web ViewSimulation per piattaforme MSaaS

In questo Capitolo viene presentato il linguaggio di simulazione eBPMN sul quale si basa l'implementazione del Simulatore di processi di business progettato dal gruppo di ricerca SEL - Software Engineering dell'Università di Tor Vergata, guidato dal Prof. Andrea D'Ambrogio. In seguito viene descritto il paradigma MSaaS e l'applicazione web ViewSimulation, realizzata come interfaccia user-friendly per l'utilizzo del Simulatore.

3.1 eBPMN

L'utilizzo della simulazione per analizzare i business process richiede prima una derivazione di un modello di simulazione dal modello BPMN per fornire tutti i dettagli necessari per rendere il modello eseguibile.

Nel precedente capitolo è stato introdotto l'estensione PyBPMN, la quale non altera il contenuto e la validità del modello originale, ma ha come obiettivo primario quello di completare i modelli BPMN, annotandoli con parametri di performance, risorse di esecuzione, carico di lavoro previsto.

L'approccio *M&S* basato sul BPMN e PyBPMN utilizzato per la realizzazione del Simulatore sfrutta anche un linguaggio di simulazione di business process. Questo linguaggio è l'**eBPMN**, basato su *JAVA*, e viene utilizzato per implementare i modelli

di simulazione creati grazie al *PyBPMN*.

Il linguaggio eBPMN è un linguaggio specifico del dominio di simulazione che è stato definito per ottenere dei modelli di simulazione in grado di rispettare le specifiche dei modelli di business process realizzati tramite il BPMN.

Infatti il linguaggio simula l'esecuzione del processo di business implementando il concetto del token definito nella specifica originale del BPMN.

Inoltre, le caratteristiche del linguaggio permettono di simulare il comportamento delle risorse e delle proprietà non funzionali mancanti nei modelli originali BPMN ma definite grazie all'estensione PyBPMN. L'eBPMN consente agli utenti di simulare sia un singolo processo sia una collaboration di processi che interagiscono tra di loro, come è stato descritto nel precedente capitolo, utilizzando gli elementi di MessageFlow.

Ogni elemento di un processo oppure di una collaboration viene specificato con le seguenti misure di performance:

- service time (mean e variance) - ossia il tempo di esecuzione, la sua varianza e il tempo medio di esecuzione di ogni resource e task;
- waiting time (mean e variance) - ossia il tempo di attesa, la sua varianza e il tempo di attesa media di ogni resource e task;
- resource utilization - ovvero la percentuale di tempo trascorsa da una resource per l'esecuzione di una richiesta;
- number of tokens processed - ovvero il numero di tokens processati da ciascun element e resource;
- cycle time (mean e variance) - ossia il tempo trascorso da un token per completare la collaboration da quando è stato generato dallo Start Event fino alla sua terminazione nell'End Event;

È necessario precisare che attualmente, l'eBPMN è in grado di implementare soltanto alcuni elementi dello standard BPMN e questo avviene in base ai seguenti vincoli:

- ogni Pool deve includere un elemento di Start Event il quale ha il compito di generare i tokens;
- ogni FlowNode deve avere un SequenceFlow in ingresso e un SequenceFlow in uscita;
- non sono supportate le Coreografie e i Complex Gateways.

Uno dei principali vantaggi di eBPMN è che il codice eseguibile può essere ottenuto direttamente da un modello BPMN tramite l'utilizzo di un approccio basato sulle tecniche model-driven.

Questo significa che anche i business analysts che non hanno nessuna conoscenza di programmazione sono in grado di utilizzare questo strumento per realizzare dei modelli di simulazione a partire dai modelli di business process creati. Ovviamente questo metodo è molto più flessibile rispetto ai vari software e tools di gestione e modellazione di business process presenti sul mercato.

Il metodo model-driven che si basa sull'eBPMN è stato realizzato seguendo le specifiche dell'OMG MDA (Model Driven Architecture) e creato con Eclipse Modeling Framework e Eclipse Modeling Project grazie all'utilizzo di alcuni plugin in grado di implementare il metamodello PyBPMN, la trasformazione dal modello BPMN a quello PyBPMN, la trasformazione dal modello PyBPMN al codice eseguibile eBPMN. Questo metodo è stato descritto in maniera approfondita in [5, 3, 4].

Le sue fasi (vedi 3.2) possono essere riassunte in questo modo:

- **Business Process Specification:** in cui il business analyst progetta un modello per il processo di business inserendo i requisiti funzionali e il risultato è un modello BPMN;
- **Business Process Annotation:** dove il business analyst completa il modello realizzato con delle annotazioni specificando le proprietà non funzionali del processo.

- **BPMN to PyBPMN Transformation:** in questa fase di trasformazione *model-to-model* il modello BPMN annotato viene trasformato automaticamente in un modello PyBPMN mediante un plugin.
- **PyBPMN to eBPMN Transformation:** sempre con l'utilizzo di un plugin il modello PyBPMN ottenuto in seguito ad una trasformazione *model-to-text* viene trasformato in codice nel linguaggio eBPMN.
- **Business Process Simulation:** viene eseguita la simulazione del codice eseguibile ottenuto precedentemente, il cui output rappresenta il comportamento del processo e delle risorse dei suoi elementi;
- **Business Process Evaluation:** in questa fase il business analyst valuta i risultati della simulazione e se questi non soddisfano le richieste definite il metodo può essere reiterato con la possibilità di effettuare delle modifiche al modello del processo.

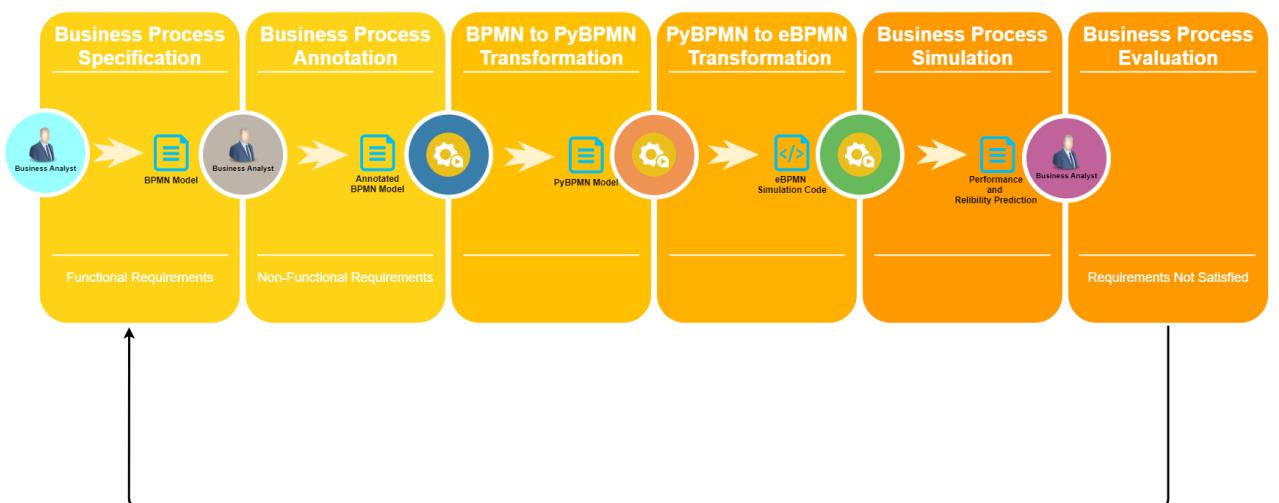


Figura 3.1: Tecniche MDA per M&S di processi di business

3.2 MSaaS

Lo sviluppo di sistemi complessi può sicuramente trarre dei vantaggi dall'utilizzo delle tecniche di analisi che si basano sul **Modeling & Simulation**. I prodotti di **M&S** sono delle risorse di altissimo valore e risulta essenziale che essi siano accessibili su larga scala. Infatti come è stato possibile osservare, gli approcci **M&S** hanno dimostrato di essere una soluzione efficace anche per l'analisi dei business process.

Per superare i problemi riguardante la costruzione, l'implementazione dei modelli di simulazione e il costo per instaurare, mantenere le infrastrutture adatte per le simulazioni distribuite sono state introdotte le tecniche di trasformazione dei modelli che si basano sull'automazione e l'utilizzo di un nuovo paradigma, il **MSaaS**.

Modeling and Simulation as a service è un concetto emergente che sfrutta l'approccio *service-oriented* per offrire le applicazioni **M&S** come dei servizi. Questo paradigma comporta una maggiore flessibilità, una migliore accessibilità e scalabilità e una maggiore affidabilità nella fornitura dei servizi offerti.

Inizialmente il paradigma utilizzava le tecnologie *web services* per raggiungere tali obiettivi, in cui la modellazione e soprattutto le funzioni di simulazioni venivano messe a disposizione come dei servizi web. I recenti sviluppi delle tecnologie *cloud* portano ulteriori miglioramenti nell'applicazione di questo approccio e nella creazione di **piattaforme MSaaS** che forniscono i servizi di modellazione e simulazione su richiesta e mantengono nascosti agli utenti i requisiti e i dettagli dell'infrastruttura, della piattaforma e del software sottostante.

3.3 ViewSimulation

3.3.1 Requisiti

Il Simulatore realizzato dal SEL prevede all'interno del progetto anche dei modelli di processi di business già definiti seguendo lo standard **BPMN** e le tecniche di *model-driven* per il **M&S** appena presentate.

Una volta avviata la simulazione di un modello tutti i risultati e le statistiche vengono stampate nella console di Eclipse, cosa che potrebbe risultare scomoda per la loro lettura, valutazione o per il loro utilizzo. Da qui nasce l'idea e la necessità di progettare una applicazione web che sia in grado di visualizzare i risultati della simulazione di un modello di business process.

L'applicazione prende il nome di **ViewSimulation** e deve dare la possibilità all'utente di scegliere un modello e modificare alcuni parametri di grande rilievo ai fini della simulazione come il numero di *replications* da effettuare e il numero di *working units* da assegnare a ciascuna *resource* utilizzata dagli elementi che compongono il modello di processo. Le statistiche della simulazione vengono presentate all'utente sotto forma di tabelle e grafici.

L'applicazione web è stata ideata per utenti che abbiano una minima conoscenza di **Business Process Management**.

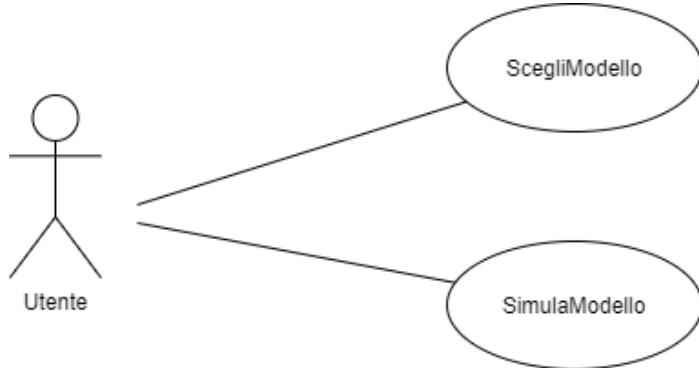


Figura 3.2: Use case Utente

3.3.2 Architettura

Si è deciso di sviluppare il **backend** dell'applicazione utilizzando il framework *Spring Boot* sulla piattaforma *Java* mentre per il **frontend** si è scelto *Angular* (vedi Figura

3.3).

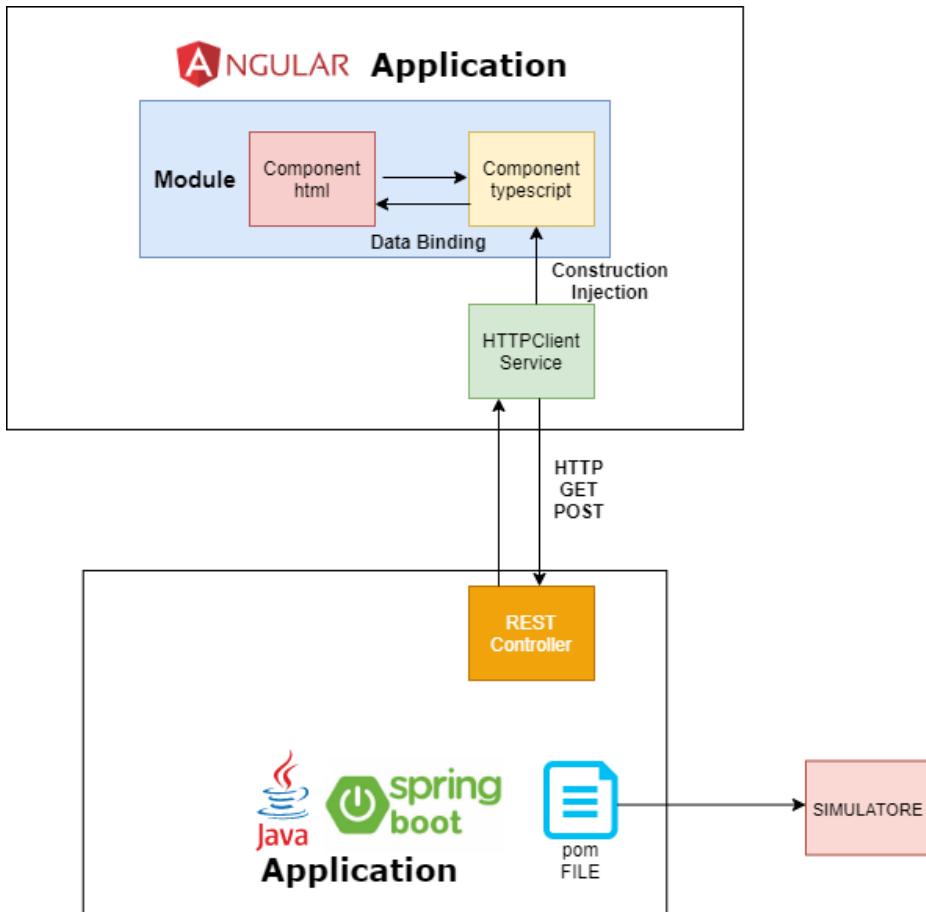


Figura 3.3: Architettura ViewSimulation

Nel backend è stata aggiunta una reference del *jar* del **Simulatore** alle dipendenze del progetto. In questo modo è stato possibile realizzare le classi di ogni modello di simulazione con il codice eBPMN autogenerato dai modelli di processo definiti precedentemente.

La classe principale del backend **AppSimulaBpApplication** contiene soltanto il metodo `main()` per far avviare l'applicazione in Spring Boot. Quindi tutto il funzio-

namento logico dell'applicazione si trova nel package dei controllers.

Utilizzando l'annotazione `@RestController` di Spring Boot sono stati creati due RestControllers che, secondo i principi dello stile architetturale REST, generano richieste HTTP tramite operazioni di tipo CRUD (Create/Read/Update/Delete):

1. **ExampleController** il quale ha come compito quello di restituire in formato `json` una lista contenente il nome, l'id, la descrizione e il diagramma di ognuno dei modelli di processo definiti.
2. **SimulazioneController** il quale deve avviare la simulazione di un modello di processo, eventualmente inserendo dei parametri ricevuti in input, e restituire un oggetto Simulazione in cui vengono memorizzati tutti i risultati che riguardano la simulazione del modello di processo.

Il Simulatore, come detto precedentemente, stampa in console tutte le statistiche inerenti alla simulazione del processo. Allora per poter utilizzare i risultati di tale simulazione e quindi avere la possibilità di trasmetterli al frontend, è stato necessario effettuare delle modifiche e la creazione di nuove classi:

Evento

- questa classe deve memorizzare le informazioni che riguardano gli elementi eBPMN dichiarati come End Event.

CollaborationStatistics

- la classe viene utilizzata per tenere traccia del comportamento degli elementi della collaboration e non solo. Infatti contiene un campo per il tempo totale della collaboration, vengono memorizzati gli End Event della collaboration e una map con una lista di misure per ciascun elemento della collaboration.

ConfigurationStatistics

- la classe serve per mantenere un riferimento a tutte le resources utilizzate dagli elementi eBPMN durante la simulazione del processo.

Simulazione

- la classe che raccoglie i dati della simulazione del processo;
- infatti contiene il numero di repliche, il numero totali dei tokens, un oggetto di tipo CollaborationStatistics e un altro oggetto di tipo ConfigurationStatistics;

È stato necessario introdurre i seguenti metodi:

- **getElementMeasures()**, metodo astratto nella classe *EPMNElement* e implementato in tutte le sue sottoclassi per restituire le misure di ciascun elemento eBPMN;
- **getResourceMeasures()**, metodo astratto nella classe *EPMNResource* e implementato in tutte le sue sottoclassi per restituire le misure di ciascuna resource del processo;
- **getStatistics()**, due metodi implementati nelle classi *Collaboration* e *ExecPlatform* che restituiscono un oggetto di tipo *CollaborationStatistic* e rispettivamente *ConfigurationStatistics*;
- **getSimulazione()** implementato nella classe *Scenario* che restituisce un oggetto di tipo *Simulazione*;
- **setCollabStats()** e **setConfigStats()** due metodi implementati nella classe *Scenario* per impostare ad un oggetto di tipo *Simulazione* un oggetto di *CollaborationStatistics* e rispettivamente di *ConfigurationStatistics*.

Per quanto riguarda il frontend i due **component** principali sono **Simulatore** e **Simulazione**. Entrambi utilizzano il **service Simulazione** per comunicare tramite il protocollo HTTP con il backend.

In Angular il service non è altro che una classe indipendente che viene definita per svolgere delle operazioni, come reperire delle informazioni da un server remoto, sollecitando i componenti da questi incarichi. Il componente invece è una classe che contiene la logica di interazione tra i dati e l'utente e definisce l'aspetto e il comportamento della View (il file html).

SimulazioneService

- il metodo **getExamples()** definito per effettuare una chiamata HTTP GET al backend, gestita da **ExampleController**.
- il metodo **startSimulazione()** definito per effettuare una chiamata HTTP POST al backend, gestita da **SimulazioneController**. Questo metodo invia un oggetto di tipo *Parametri*.

Simulatore

- l'obiettivo di questo componente è quello di ottenere dal backend tutti i modelli definiti e metterli a disposizione dell'utente;
- il metodo **getExamples()** utilizza il metodo **getExamples()** di SimulazioneService per ottenere un oggetto di tipo *Esempi* attraverso il quale viene effettuato il parsing dei dati ricevuti.

Simulazione

- l'obiettivo di questo componente è quello di avviare la simulazione del modello di processo scelto dall'utente e visualizzarne i risultati;
- il metodo **getSimulazione()** prende in input un oggetto di tipo *Parametri* e utilizza il metodo **startSimulazione()** di SimulazioneService per ottenere i risultati della simulazione del processo di business richiesta dall'utente.

Capitolo 4

Caso di Studio

In questo Capitolo viene descritto il caso studio analizzato in cui sono stati utilizzati alcuni degli strumenti presentati nei capitoli precedenti, tra i quali anche l'applicazione web ViewSimulation. Questo caso studio - **Emergency Attendance** è stato già discusso in [5] e si ispira ad un processo utilizzato come esempio dal Bizagi Process Modeler.

4.1 Descrizione Emergency Attendance

Una volta avviata l'applicazione **ViewSimulation**, l'utente procede con la scelta del modello.

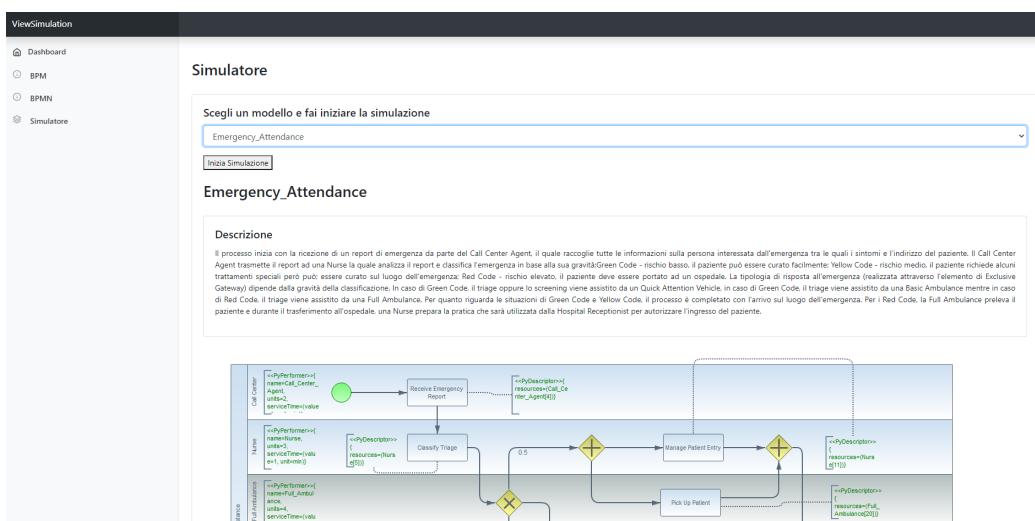


Figura 4.1: ViewSimulation-Scelta del modello di processo

In figura 4.1 è possibile osservare anche una breve descrizione del processo Emergency Attendance.

Il processo inizia con la ricezione di un report di emergenza da parte del **Call Center Agent**, il quale raccoglie tutte le informazioni sulla persona che ha fatto scattare l'emergenza. Ovviamente tra queste informazioni anche i sintomi e l'indirizzo del paziente.

Il **Call Center Agent** trasmette il report appena fatto ad una **Nurse**, la quale lo analizza e in base alla gravità della situazione classifica l'emergenza in:

- **Green Code** - il rischio è basso quindi il paziente può essere curato facilmente.
- **Yellow Code** - il rischio è medio in cui il paziente richiede alcuni trattamenti speciali però può essere curato sul luogo dell'emergenza.
- **Red Code** - il rischio è elevato quindi il paziente deve essere portato ad un ospedale.

Attraverso l'elemento di *Exclusive Gateway* si realizza la risposta all'emergenza. Questa risposta dipende certamente dalla gravità della situazione in base alla classificazione fatta precedentemente.

In caso di **Green Code** il triage oppure lo screening viene effettuato da una **Quick Attention Vehicle**, in caso di **Yellow Code** viene effettuato da una **Basic Ambulance** mentre nel caso più grave viene effettuato da una **Full Ambulance**. In quest'ultimo caso, a differenza delle situazioni di **Green Code** e **Yellow Code** in cui il processo termina con l'arrivo sul luogo dell'emergenza, in caso di **Red Code** la **Full Ambulance** preleva il paziente e durante il trasferimento all'ospedale, una **Nurse** prepara la pratica che sarà utilizzata dalla **Hospital Receptionist** per autorizzare l'ingresso del paziente.

4.2 Mapping del processo

In figura 4.2 è possibile osservare il modello BPMN del processo di Emergency Assistance.

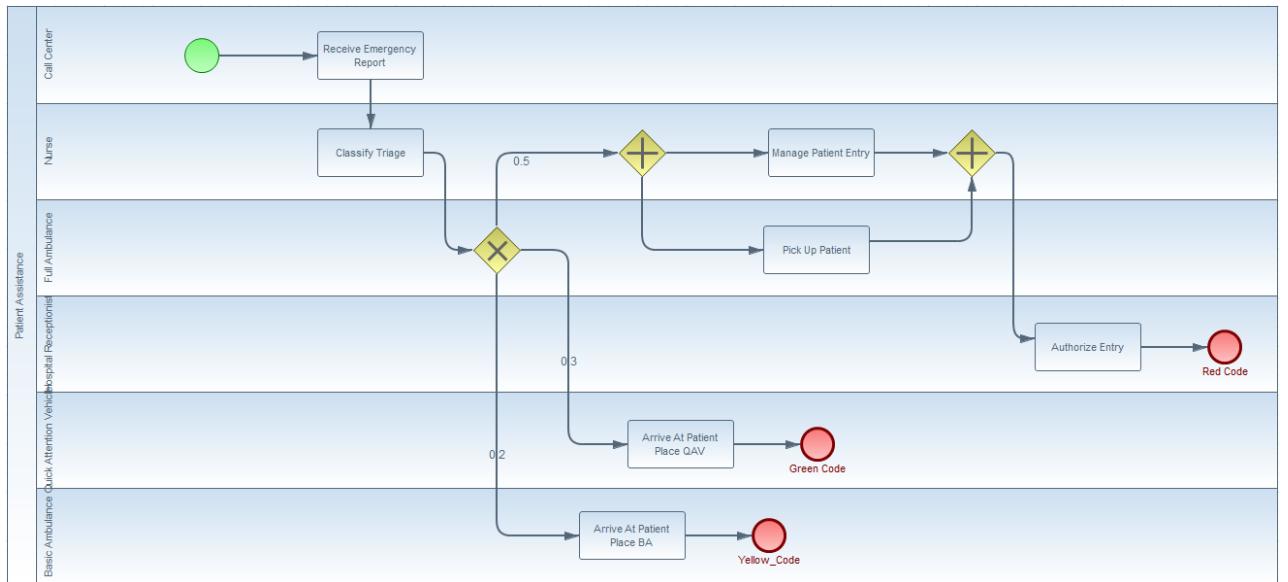


Figura 4.2: Modello BPMN di Emergency Attendance

Gli elementi di tipo **Event** si dividono in: uno **Start Event** che rappresenta il momento in cui il Call Center Agent riceve il report di emergenza e tre **End Event** che sono rappresentati dai momenti in cui il processo termina nei casi di Green Code, Yellow Code e Red Code.

In base a delle statistiche effettuate sull'analisi dei report di emergenza, questo modello di processo presenta la seguente distribuzione di probabilità di ciascuna tipologia di emergenza o flusso:

- Red Code: 50%.
- Yellow Code: 30%
- Green Code: 20%

Sono state identificate le seguenti **Activities** (**Task**), ciascuna con il suo tempo di elaborazione in minuti e le **Resources** da esse utilizzate per l'esecuzione del processo:

Activity	Processing Time(min)	Resource
Receive Emergency Report	4	Call Center Agent
Classify Triage	5	Nurse
Manage Patient Entry	11	Nurse
Pick Up Patient	20	Full Ambulance
Arrive At Patient Place QAV	7	Quick Attention Vehicle
Arrive At Patient Place BA	10	Basic Ambulance
Authorize Entry	4	Hospital Receptionist

Tabella 4.1: Tempi di elaborazione e Resources utilizzate dalle Activities

Per l'esecuzione di questo processo si è deciso la seguente assegnazione delle quantità disponibili per le Resources identificate:

Resource	Quantity
Call Center Agent	2
Nurse	3
Full Ambulance	4
Quick Attention Vehicle	2
Basic Ambulance BA	2
Hospital Receptionist	2

Tabella 4.2: Quantità disponibili delle Resources

4.3 Annotazione del modello BPMN

In figura 4.3 è possibile osservare il modello BPMN annotato del processo di Emergency Assistance.

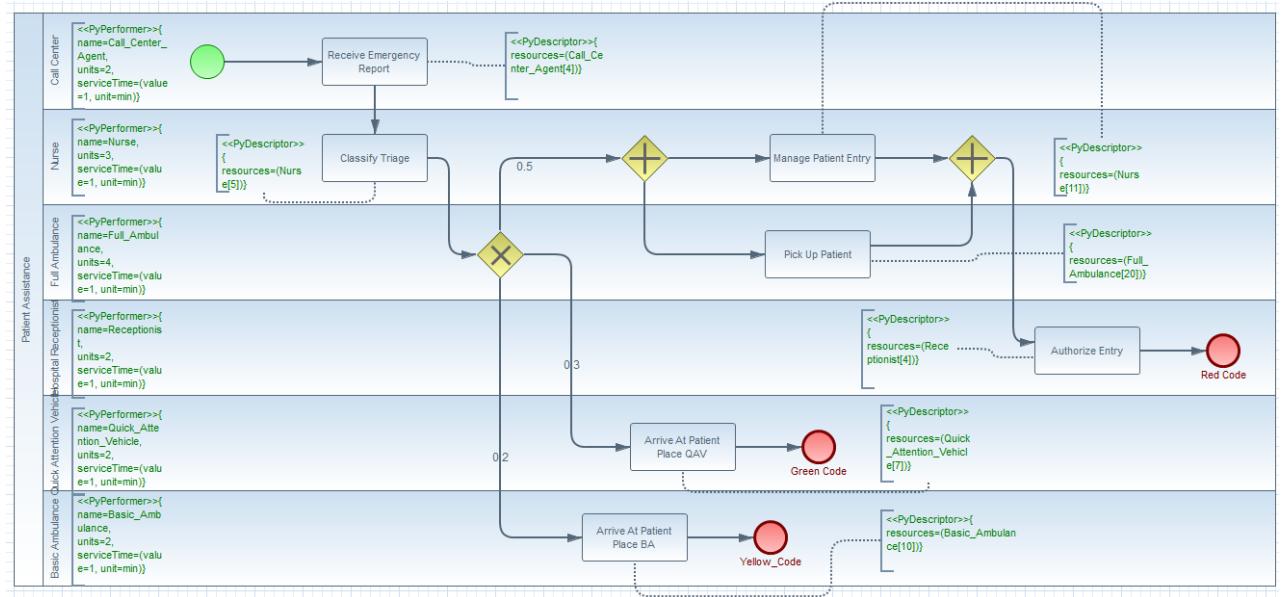


Figura 4.3: Modello BPMN Annotato di Emergency Attendance

Ciascuna **Resource** è stata specificata tramite l'annotazione `<<PyPerformer>>` con la quale sono state aggiunte le quantità disponibili della tabella 4.2.

Invece la specifica delle **Resources** utilizzate e i tempi di elaborazione di ogni **Activity** è stata realizzata tramite l'annotazione `<<PyDescriptor>>` con i valori della tabella 4.1.

A partire da questo modello BPMN annotato, utilizzando i tools di Eclipse *BPMN_to_PyBPMN* e *PyBPMN_to_eBPMN* è stato possibile ottenere automaticamente il codice *eBPMN* del processo con il quale effettuare la **Business Process Simulation**.

4.4 Simulazione di Emergency Attendance

Una volta selezionato il modello di Emergency Attendance e premuto il pulsante *Inizia Simulazione* viene avviata la simulazione del processo.

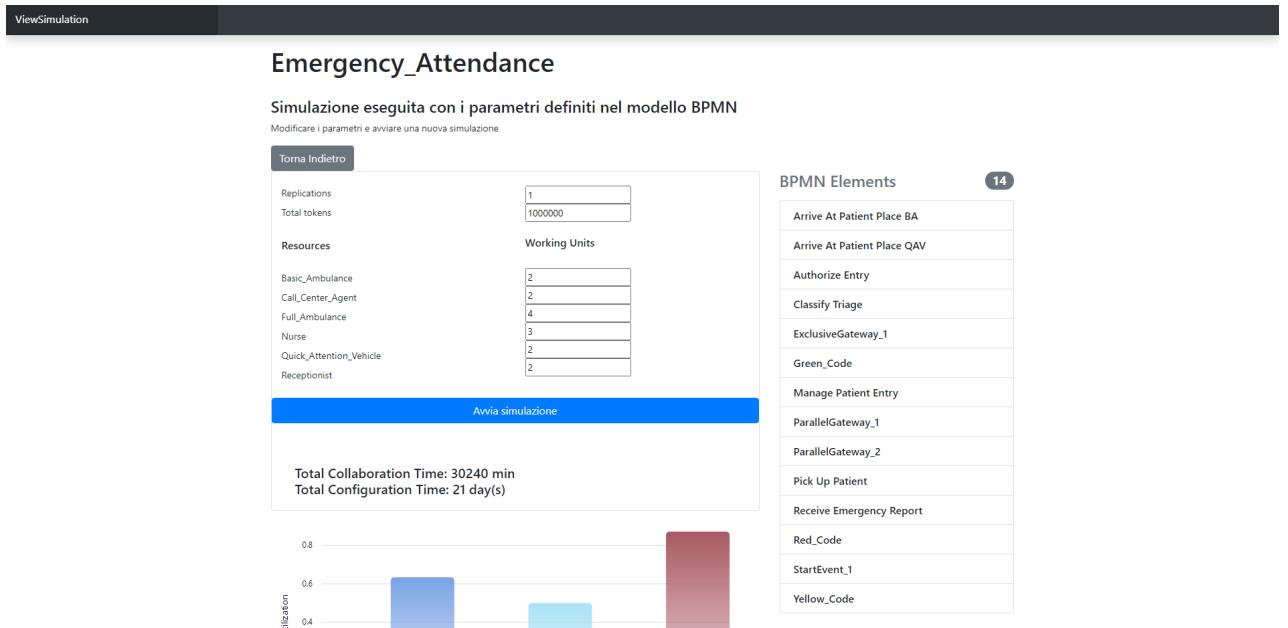


Figura 4.4: Simulazione Emergency Attendance

In figura 4.4 è possibile notare che la simulazione è stata effettuata con i parametri definiti precedentemente e l'utente ha la possibilità di modificare per esempio il numero di **Working Units** di ciascuna **Resource** e avviare una nuova simulazione e valutare le modifiche.

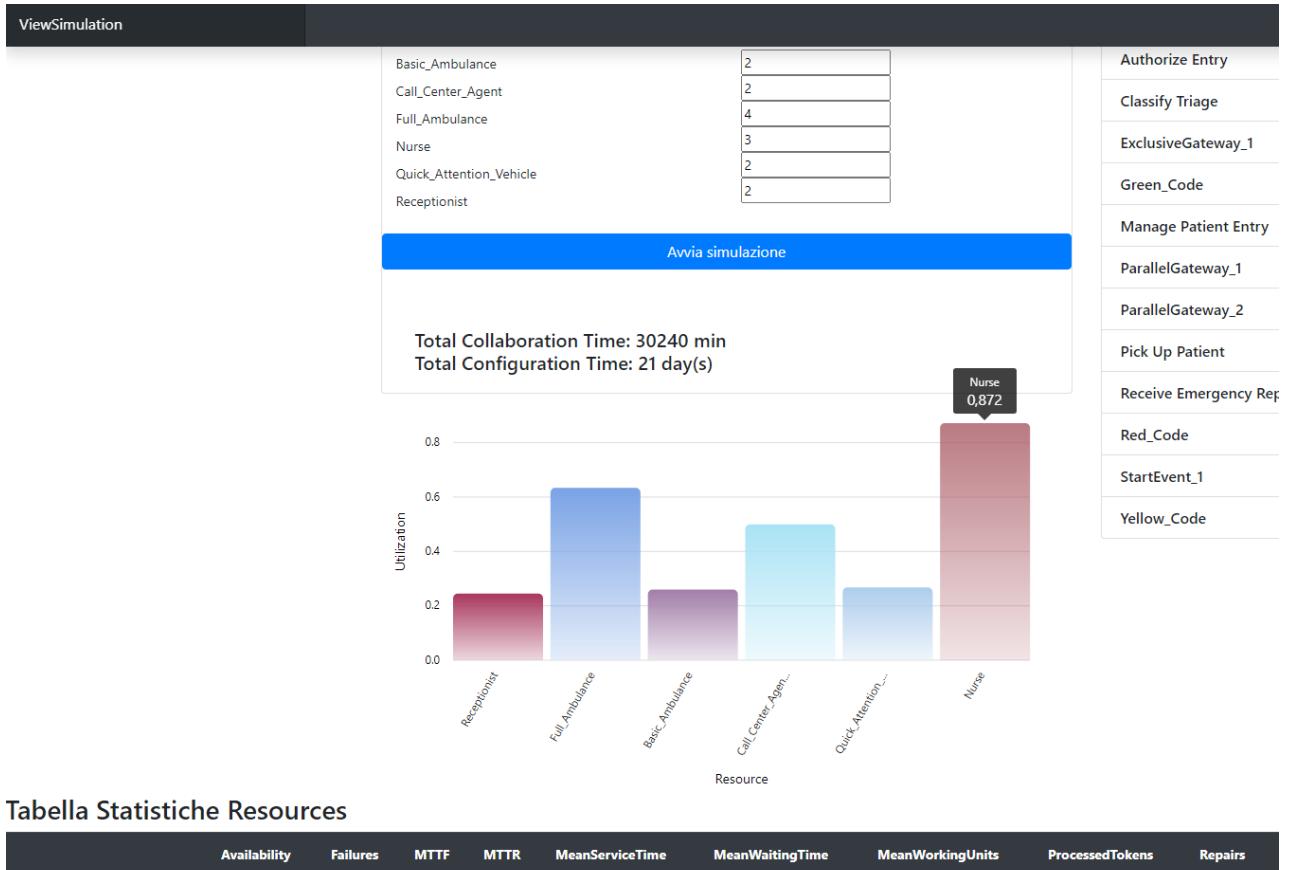


Figura 4.5: Percentuale di utilizzo di Nurse

In figura 4.5 è possibile osservare che il valore di *Resource Utilization* di **Nurse** ossia la percentuale di tempo in cui la resource Nurse è stata occupata nell'elaborazione di un task, è di 0.872 con l'utilizzo di 3 Working Units.

Aumentando le Working Units di Nurse per un totale di 6 unità e avviando una nuova simulazione, è possibile osservare in figura 4.6 come il valore di *Resource Utilization* di Nurse sia passato da 0.872 a 0.432. In questo modo si può valutare se è necessario utilizzare una risorsa in più in modo che il tempo di elaborazione di un certo task non sia così alto.

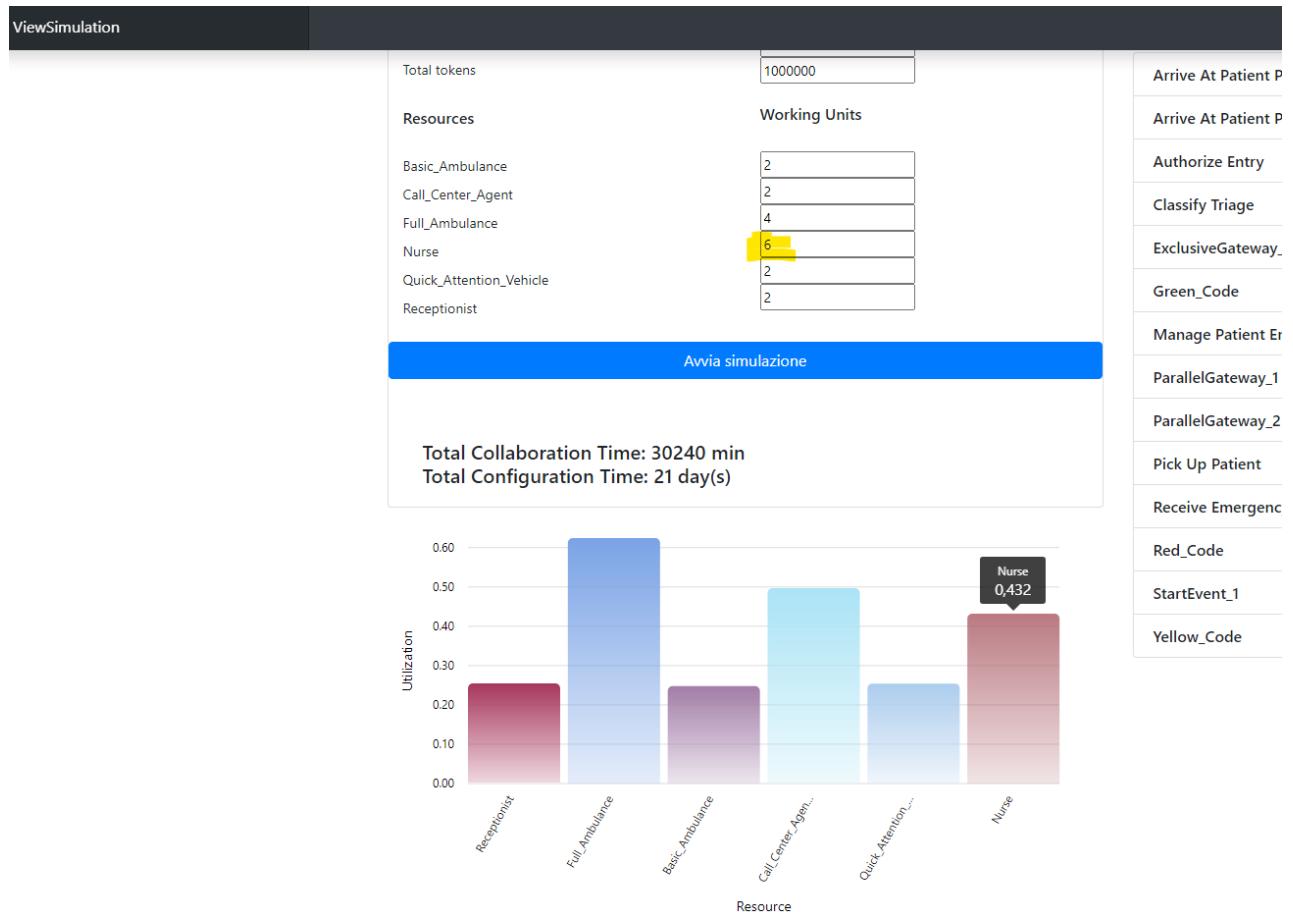


Figura 4.6: Percentuale di utilizzo di Nurse dopo l'aumento delle Working Units

Capitolo 5

Conclusioni

In questo lavoro di tesi è stato analizzato il tema della modellazione e della simulazione dei processi di business come fase di grande rilievo nella gestione dei processi aziendali. Quindi è stata descritta la notazione standard *BPMN* con la quale sono stati specificati i processi.

In particolare sono stati introdotti degli strumenti come il *PyBPMN* e l'*eBPMN* in grado di dimostrare l'efficienza e l'importanza di un approccio di tipo *Modeling & Simulation* per l'analisi simulativa dei processi.

In seguito è stata descritta la progettazione e l'implementazione di un'applicazione web, la cui realizzazione è stata possibile utilizzando il motore di simulazione creato dal SEL dell'Università di Tor Vergata. Inoltre è stato presentato il paradigma del *model-driven* applicato all'approccio di tipo *Modeling & Simulation* utilizzato per la creazione del motore. Di tale paradigma sono stati evidenziati sia gli aspetti positivi sia quelli che potrebbero risultare inconvenienti nella sua applicazione come strumento nelle varie attività del *Business Process Management*.

Successivamente si è scelto un modello di processo come caso studio ed è stato esaminato mediante l'utilizzo dell'applicazione web. In questo modo è stato possibile evincere la validità di un simile applicativo in cui l'utente ha modo di valutare i risultati della simulazione del processo e soprattutto analizzare gli effetti di eventuali modifiche eseguite ai parametri in input.

L'applicazione propone all'utente dei modelli di processo predefiniti e ovviamente

questo potrebbe rappresentare un limite importante dal punto di vista dell’usabilità. Quindi un futuro sviluppo di grande rilievo è quello dell’implementazione di un *parser* che sia in grado di elaborare un modello *BPMN*. In questo modo l’utente ha l’opportunità di inserire un modello di processo e avviare la simulazione.

Capitolo 6

Ringraziamenti

In primis, un ringraziamento speciale al mio Relatore, il prof. Andrea D'Ambrogio per la sua disponibilità, per i suoi consigli e per tutte le conoscenze che mi ha trasmesso durante questo percorso di tesi e durante il suo corso di insegnamento.

Ringrazio infinitamente i miei genitori e mia sorella per tutto il sostegno ricevuto durante questi anni, siete sempre stati i primi a credere in me ed è grazie a voi se, nonostante le mille difficoltà, non abbia mai pensato a mollare e sia riuscito a superare anche questo traguardo.

Continuo con ringraziare i miei amici più stretti, che a modo loro ovviamente, mi hanno sempre incoraggiato durante il mio percorso di studi: Claudia, Giancarlo, Sandro, Peppe, Mamiano, Matteo F. e Matteo G.

Un grazie di cuore ad Alex, Lorenzo e Fede ovvero il gruppo JDMD. Con voi ho condiviso l'ultima parte del mio percorso di studi, sicuramente la più importante, in cui ho imparato cosa significa il lavoro di squadra e dove ho incontrato tre bellissime persone.

Elenco delle figure

1.1	Descrizione di un processo	3
1.2	Business Process Management Lifecycle	5
2.1	Tabella Flow Objects	15
2.2	Tabella Artifacts	16
2.3	Tabella Connecting Objects	17
2.4	Tabella Swimlanes	17
3.1	Tecniche MDA per M&S di processi di business	22
3.2	Use case Utente	24
3.3	Architettura ViewSimulation	25
4.1	ViewSimulation-Scelta del modello di processo	29
4.2	Modello BPMN di Emergency Attendance	31
4.3	Modello BPMN Annotato di Emergency Attendance	33
4.4	Simulazione Emergency Attendance	34
4.5	Percentuale di utilizzo di Nurse	35
4.6	Percentuale di utilizzo di Nurse dopo l'aumento delle Working Units .	36

Bibliografia

- [1] Paolo Bocciarelli and Andrea D'Ambrogio. A bpmn extension for modeling non functional properties of business processes. In *Proceedings of the 2011 Spring Simulation Multi-Conference*.
- [2] Paolo Bocciarelli and Andrea D'Ambrogio. Performability-oriented description and analysis of business processes. In *Business Process Modeling: Software Engineering, Analysis and Applications*.
- [3] Paolo Bocciarelli, Andrea D'Ambrogio, Andrea Giglio, and Emiliano Paglia. Model trasformation services for msaas platforms. In *Proceedings of the 2018 Spring Simulation Conference (SpringSim'18)*.
- [4] Paolo Bocciarelli, Andrea D'Ambrogio, Andrea Giglio, and Emiliano Paglia. On the performance prediction capabilities of the ebpmn-based model-driven method for business process simulation.
- [5] Paolo Bocciarelli, Andrea D'Ambrogio, Andrea Giglio, and Emiliano Paglia. Bpmn-based business process modeling and simulation. In *Proceedings of the Winter Simulation Conference*, WSC '19, pages 1439–1453. IEEE Press, 2019.
- [6] Paolo Bocciarelli, Andrea D'Ambrogio, Andrea Giglio, Emiliano Paglia, and D. Gianni. A model-driven method for enacting the design-time qos analysis of business processes. In *Software Systems Modeling*.
- [7] Paolo Bocciarelli, Andrea D'Ambrogio, and Emiliano Paglia. A language for enabling model-driven analysis of business processes. In *Proceedings of*

the 2nd International Conference on Model-Driven Engineering and Software Development.

- [8] J. Mendling H. Reijers Dumas M., M. La Rosa. *Fundamentals of Business Process Management*. Springer-Verlag, 2018.
- [9] Object Management Group. *Business Process Model And Notation*, 2011.
- [10] Happyfox.com. What-is-bpmn. <https://www.happyfox.com/what-is-business-process-management/>, 2020.
- [11] Akhil Kumar. *Business Process Management*. Routledge, 2018.
- [12] Mathias Weske. *Business Process Management*. Springer, 2019.