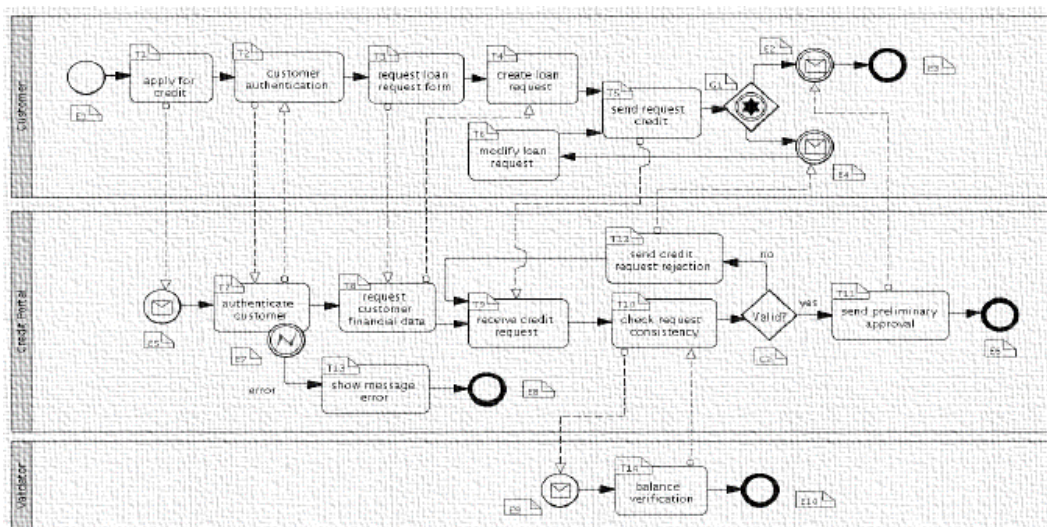


# طراحی مدل فرآیند کسب و کار یا Business Process Diagram به واسطه استاندارد BPMN



تهیه و تنظیم: پیمان مالکی



## فهرست مطالب

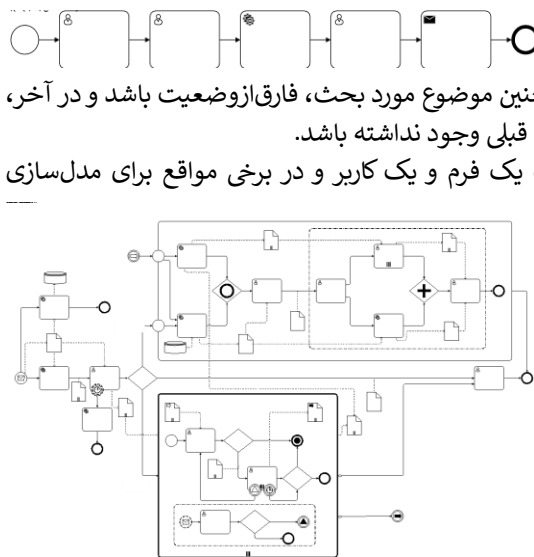
2	نمودار مدل فرآیند کسب و کار با نمادگذاری
4	چه زمانی و به چه دلایلی تهیه این نمودار اهمیت خواهد داشت؟
4	گذرگاه‌ها
5	گذرگاه مبتنی بر قاعده
8	گذرگاه مبتنی بر رویداد
9	فعالیت‌ها
9	فعالیت‌های مبتنی بر وظایف
13	فعالیت‌های مبتنی بر زیرفرآیندها
21	طبقه‌بندی ثانویه فعالیت‌ها
24	جریان‌ها
25	جریان‌های توالی
26	جریان‌های پیام
28	ارتباطات
30	داده‌ها
30	اشیاء داده
33	مخزن داده
34	اشیاء پیام
35	نمادهای افزوده شده
35	گروه‌ها
36	حاشیه‌نویسی‌های متنی
36	حوزه‌ها و مسیرهای فرآیند
37	حوزه‌های فرآیند
39	مسیرهای فرآیند
39	رویدادها
43	چند پیشنهاد مفید در هنگام طراحی فرآیندها
44	نمونه نمودار فرآیند
47	لغات پرکاربرد BPMN

در هر پروژه‌ای که با ذینفعان فراوانی درگیر باشید، حتماً با چالش بزرگی به نام درک مفاهیم از منظرهای متفاوت، برخورد خواهید داشت. این عدم تطابق درک به دلیل تفاوت سطح انتزاع و یا دیدگاه افراد از مطالب مورد بحث است. همواره، بزرگ‌ترین چالش در این پروژه‌ها، استفاده از یک زبان ساده، گویا و فراگیر است. همه افراد یک پروژه باید بتوانند با زبان و با یکدیگر به راحتی ارتباط برقرار نمایند. کارهایی مثل، برگزاری جلسات مجتمع با حضور کلیه ذینفعان، مستندسازی، استفاده از استانداردهای مورد تأیید کلیه افراد پروژه و ده‌ها پیشنهاد دیگر، راه‌حلهایی برای کاهش این تنش‌ها و چالش‌ها است. این موارد، سرعت پیشرفت پروژه را افزایش می‌دهد. در برخی مواقع سعی می‌شود با استفاده از این موارد، مشکل عدم درک را به حداقل رسانند. ولی یک پیشنهاد مفید این است که با زبانی صحبت کنید که همه افراد درک مناسب و کافی از عناصر تشکیل دهنده آن داشته باشند تا برای درک آن نیاز به محاسبات و یا تحلیل خاصی وجود نداشته باشند!

بهترین زبان برای بیان مطالب، زبان تصویر است و بهترین روش برای به تصویر کشیدن مفاهیم ذهنی، ترسیم نمودار است. مرتبطین هر کسب و کار با تجسم فرآیندهای کسب و کاری در قالب نمودار، بسیار راحت خواهند بود. اگر راهکاری بتواند از ترسیم نمودارهای اولیه تا برنامه کاربردی مورد نظر را پشتیبانی نماید، ارزش قابل توجهی را به ذینفعان پروژه ارائه کرده است. در همین راستا، استاندارد BPMN نمادهایی را در اختیار قرار می‌دهد که الگوریتم و گفتگوی عناصر یک فرآیند با یکدیگر را ترسیم می‌نماید. تعامل و تقابل فرآیندهای کسب و کاری با سطح انسانی، در مقابل تعامل و تقابل فرآیند با سطح سیستمی یا نرم‌افزاری، می‌تواند با مدل‌سازی فرآیند کسب و کار به کمک نمادگذاری (BPMN) حل شود. ولیکن برای اینکه این استاندارد، شما را در رسیدن به این هدف والا یاری کند باید کلیه ذینفعان پروژه با این استاندارد آشنا شوند تا بتوانند کلیه نیازها، درخواست‌ها و موضوعات و طراحی‌ها را به واسطه همین استاندارد، شناسایی، مبادله، بررسی و تأیید نمایند.

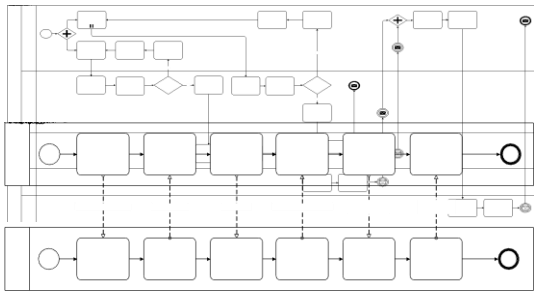
استاندارد BPMN نمودارهای مختلفی را ارائه می‌دهد، که برای طراحان و مسئولین فرآیندهای کسب و کاری، طراحی شده‌اند. با استفاده از استاندارد BPMN در سیستم‌های مدیریت فرآیند کسب و کار (BPMS)، سازوکارهایی برای تطابق زبان اجرایی سیستم‌های BPM ارائه شده‌است. یک نمودار فرآیند کسب و کاری مبتنی بر استاندارد BPMN را BPD می‌گویند. در یک نگاه انتزاعی، مدل‌ها و کاربردهای مختلفی می‌توان از این مدل‌سازی انتظار داشت و از این نظر می‌توان انتظار داشت که این مدل‌سازی دارای انواع نمونه زیر باشد:

- مدل‌های اختصاصی یا حفاظت شده: این مدل‌ها بیشتر برای مدل‌سازی سرویس و یا الگوریتم‌های درون-ماشینی استفاده می‌شوند. این مدل را زمانی باید استفاده نمود که موضوع مورد بحث از بیش از یک کنترل‌کننده سرویس<sup>1</sup> استفاده ننماید و به گونه‌ای باشد که با پایین‌ترین سطح انتزاع با الگوی، دریافت پیام درخواست، بررسی و در نهایت، عملیات ارسال پیام پاسخ، قابل پیاده‌سازی باشد و همچنین موضوع مورد بحث، فارق از وضعیت باشد و در آخر، عامل انسانی مثل عملیات فرم مکانیزه یا عملیات دستی کاربر در طی سه مرحله ذکر شده قبلی وجود نداشته باشد.
- مدل‌های عمومی مبتنی بر الگوریتم یا فرم: این مدل‌ها بیشتر برای مدل‌سازی عملیات یک فرم و یک کاربر و در برخی مواقع برای مدل‌سازی الگوریتم‌های تصمیم‌گیری استفاده می‌شوند. این مدل را زمانی باید استفاده نمود که موضوع مورد بحث دارای الگوریتم چند مرحله‌ای است که می‌توان هر مرحله از آن را در یک زمان، ولیکن با مفاهیم یا موضوعات متفاوت و به هم پیوسته دنبال کرد و به گونه‌ای باشد که با پایین‌ترین سطح انتزاع با الگوی، گفتگوی کاربر و سیستم، قابل پیاده‌سازی باشد و همچنین در الزامات موضوع مورد بحث، فارق از وضعیت یا وابسته به وضعیت بودن؛ اولویت کسب و کاری نباشد و در آخر، چنانچه در الزامات موضوع مورد بحث، وابسته به وضعیت، انتخاب شود، برای پیاده‌سازی این الزام، نیازی به ایجاد و یا استفاده از مفهوم کارتابل وجود نداشته باشد. این حالت را شبه وضعیت نیز می‌گویند.
- مدل‌های عمومی مبتنی بر یک کارتابل: این مدل‌ها بیشتر برای مدل‌سازی عملیات یک دستورالعمل یا فرآیند سازمانی چند کاربری استفاده می‌شوند. این مدل را زمانی باید استفاده نمود که موضوع مورد بحث دارای الگوریتم چند مرحله‌ای است که نمی‌توان هر مرحله از آن را در یک زمان دنبال کرد و به گونه‌ای باشد که با پایین‌ترین سطح انتزاع با الگوی، گفتگوی کاربران با یکدیگر و سیستم، قابل پیاده‌سازی باشد و همچنین موضوع مورد بحث، وابسته به وضعیت باشد و برای آن نیاز به ایجاد و یا استفاده از مفهوم



<sup>1</sup> کنترل‌کننده سرویس یا باس کنترلر که به صورت خلاصه به آن کنترلر گفته می‌شود، مثل: سرور ایمیل، سرور بانک اطلاعاتی و سرور برنامه

یک کارتابل داشته باشد تا وضعیت فرآیند در آن ذخیره گردد و در آخر، کاربران این فرآیند باید در یک سازمان، گروه و یا واحد باشند (بسته به



تقسیم‌بندی دامنه‌های عملیاتی) و کلیه عملیات اجرایی مورد نظر در مدل مورد نظر را فقط برای ایجاد یک نتیجه یا محصول مشخص انجام دهند.

- مدل‌های عمومی مبتنی بر چند کارتابل: این مدل‌ها بیشتر برای مدل‌سازی عملیات بین سازمانی یا بین واحدی یا بین فرآیندی استفاده می‌شود که چند کاربر از گروه‌ها یا طبقه‌بندی و یا سازمان‌های مختلف در هر یک از آن فرآیندها دخیل هستند. این مدل را زمانی باید استفاده نمود که موضوع مورد بحث دارای الگوریتم چند مرحله‌ای است که نمی‌توان مراحل آن را در یک زمان دنبال کرد و به گونه‌ای باشد که با پایین‌ترین سطح انتزاع با الگوی، خودکارسازی فرآیندها در تعامل با یکدیگر، قابل پیاده‌سازی

باشد و موضوع مورد بحث، وابسته به وضعیت باشد و برای این کار نیاز به ایجاد و یا استفاده از مفهوم چند کارتابل داشته باشد تا وضعیت هر فرآیند در کارتابل وابسته به آن ذخیره گردد. از طرف دیگر وجود یک سیستم پیام‌رسان بین فرآیندی قطعی شده باشد و رویدادهای آغاز مبتنی بر پیام برای هر یک از فرآیندها، تعریف شده باشد. یا به عبارت دیگر هر یک از فرآیندها با دریافت پیام مشخص شده‌ای، بتوانند آغاز شوند و به چرخه بیفتند و همچنین فرآیندها، دارای رویدادهای میانی مبتنی بر پیام برای ارسال پیام درخواست و دریافت پیام پاسخ باشند.

در کل، مدل‌سازی استاندارد فرآیند کسب‌وکار با نمادگذاری (BPMN) به کسب‌وکارها توانایی درک رویه‌های کسب‌وکاری خود را در یک نمای نمادگذاری شده، می‌دهد و به سازمان‌ها این توانایی را می‌دهد که این رویه‌ها را با شیوه‌ای استاندارد به اشتراک بگذارند. علاوه بر این، استفاده از این نمادهای گرافیکی، درک عملکردهای هر همکاری و تقابل کسب‌وکاری بین سازمان‌ها را تسهیل می‌کند. این استاندارد، به طور صددرصد و تضمین شده کلیه کسب‌وکارها و شرکت‌کنندگان در کسب‌وکارها را پوشش می‌دهد و به سازمان‌ها این امکان را می‌دهد که به سرعت با شرایط پیش‌بینی نشده، سازگار شوند و یا در تعاملات تجاری B2B سریع‌تر توسعه یابند.

نکته: این استاندارد فاقد هر گونه قاعده نامگذاری عناصر است ولیکن برخی قواعد زیر بهترین پیشنهاد برای روش‌های نامگذاری عناصر نمودارها می‌باشند:

- همیشه از کلمات کلیدی که برای کسب‌وکار معنادار هستند استفاده کنید
- از اختصارات غیر معمول استفاده نکنید
- از نوع عنصر در نام آن استفاده نکنید
- از حروف تعریف و ضمائر خودداری کنید
- همه فعالیت‌ها را نامگذاری کنید
- نام فعالیت‌ها از یک عبارت Verb-Noun تشکیل شده باشد
- همواره از زمان حال یک فعل، استفاده کنید
- از اسامی شناخته‌شده برای کسب‌وکار مورد نظر استفاده کنید
- هرگز چندین فعالیت را با یک نام نامگذاری نکنید (به جز فعالیت‌های فراخوانی)
- گذرگاه‌ها هیچ کاری انجام نمی‌دهند یا تصمیم نمی‌گیرند. فقط تجسمی از واگرایی یا همگرایی جریان هستند؛ پس هرگز گذرگاه‌های همگرا را نامگذاری نکنید
- اگر عدم نامگذاری یک گذرگاه همگرا، درک نمودار را مشکل کند از یک حاشیه‌نویسی متنی برای آن استفاده کنید
- گذرگاه‌های انتخاب‌انحصاری واگرا را با یک عبارت پرسشی نامگذاری کنید
- جریان‌های توالی شرطی را با استفاده از شرایط مرتبط آنها که به عنوان نتایج بیان شده است نامگذاری کنید
- جریان‌های توالی پیش‌فرض را نامگذاری نکنید
- همه رویدادها باید نامگذاری شوند
- رویدادهای پیام، سیگنال، سطح‌بندی و خطا را با یک فعل فعال نامگذاری کنید
- رویداد پیوند را با یک Noun نامگذاری کنید
- رویدادهای زمان را با استفاده از برنامه زمانی آنها نامگذاری کنید
- رویدادهای شرطی را با استفاده از شرایط فعال‌سازی آنها نامگذاری کنید
- رویدادهای پایانی را با استفاده از عنوان وضعیت پایانی نامگذاری کنید
- همه اشیاء داده باید نامگذاری شوند
- نام اشیاء داده را با استفاده از یک اسم واجد شرایط که نام یک شیء کسب‌وکاری یا شیء اطلاعاتی است که برای کسب‌وکار معنی دارد، نامگذاری کنید
- چندین نمونه از یک شیء داده یکسان (که واقعاً به یک شیء داده اشاره دارند) را با استفاده از یک نام یکسان که به دنبال آن وضعیت شیء مورد نظر در برکت یا پرانتز مشخص شده است نامگذاری کنید

- شرکت‌کنندگان یا نقش‌ها را با استفاده از یک اسم واجد شرایط یا یک عبارت اسمی نام‌گذاری کنید
- نام حوزه‌های فرآیند را با استفاده از نام مشارکت‌کننده‌ها و یا نام فرآیندهای کسب‌وکاری نامگذاری نکنید
- یک مسیرفرآیندی را با استفاده از نقش‌کاری، نامگذاری کنید

#### چه زمانی و به چه دلایلی تهیه این نمودار اهمیت خواهد داشت؟

همانطور که در توضیح قبل نیز بیان شد؛ سه مدل از چهار مدلی که می‌توان به کمک این استاندارد، به تصویر کشید دارای یک مورد کاربری هستند و یا به گونه‌ای با آن ارتباط دارند. پس اگر موضوع مورد شناسایی و یا تحلیل، به کاربران، نقش‌های کاربری، کنشگران بیرون از سیستم و یا نقش‌های سیستمی ارتباطی داشت و یا در آن مطرح باشد؛ باید با جدیت بیشتری به دنبال طراحی نمودارهای فرآیندی آنها با این استاندارد باشید.

در واقع استاندارد BPMN جایگزین سخت گیرانه جدیدی برای نمودارهای فلوچارت متعلق به استاندارد سال 1920 میلادی می‌باشد و با وجود اینکه یادگیری آن، سخت‌تر است ولیکن با ارائه جزئیات بیشتر و ترسیم‌های دقیق‌تر، می‌تواند راه توسعه سیستم در حال بررسی را آسانتر کند.

با توجه به اینکه این استاندارد قالباً مبتنی بر نمادهای خود، تعریف می‌شود؛ به نظر می‌رسد که برای درک بهتر این نمودارها و استفاده مناسب‌تر از آنها در توسعه سیستم بهتر است این نمادها را به صورت کامل و دقیق بشناسیم.

نماد بسیار مهمی که در یک BPD ممکن است وجود داشته باشد، نماد گذرگاه (نماد هدایت پس از تصمیم‌سازی) است. در هر الگوریتمی که مورد بررسی قرار می‌گیرد، به موارد یا مراحل خواهد رسید که باید بتوانید بین چند جریان قابل ادامه در مسیر الگوریتم، یکی را انتخاب کنید. شاید به نظر برسد که هر انتخاب، می‌تواند نماینده یک یا چند شرط باشد که در محل مورد نظر، بررسی می‌شود و عملکرد مشخص شده برای هر نتیجه بررسی، اجرا می‌شود ولیکن استاندارد BPMN نوع دیگری از گذرگاه‌ها را معرفی کرده‌است که مبتنی بر بررسی شرایط نیستند. به همین منظور دو طبقه‌بندی زیر برای گذرگاه‌ها در این استاندارد وجود دارد:

- گذرگاه‌های مبتنی بر قواعد (که قبلاً در نمودارهای فلوجارت نیز دیده می‌شد)
- گذرگاه‌های مبتنی بر رویداد (در این استاندارد افزوده شده است)

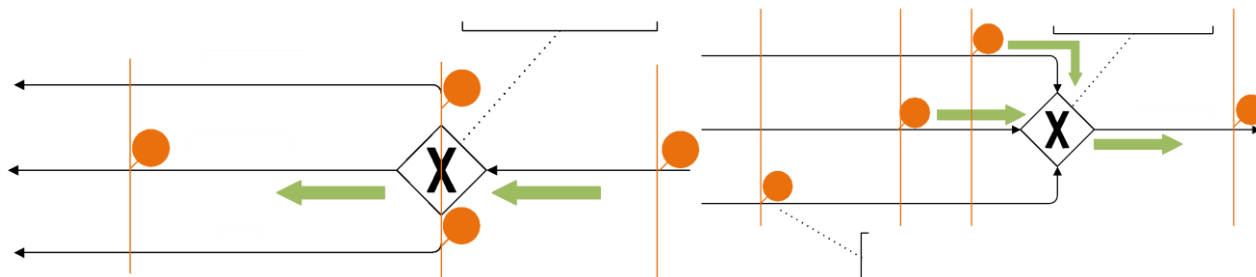
هر کدام از انواع گذرگاه‌های موجود در طبقه‌بندی ذکر شده، با توجه به نوع تست مورد بررسی نیز قابل تفکیک هستند. هر کدام از طبقه‌بندی‌های بالا دارای انواع تست خود هستند که در ادامه به تفکیک توضیح داده می‌شود.

در اینجا لازم می‌دانم که مفهوم گذرگاه را کمی بیشتر توضیح دهم. گذرگاه، محلی برای هدایت گزینشی است ولیکن تصمیم‌سازی برای این گزینش باید قبل از آن صورت گرفته باشد. شاید در این انتخاب، وقوع یک واقعه یا ارزش یک متغیر و یا خیلی موارد دیگر تاثیرگذار باشد به همین دلیل هر آیت مورد ارزیابی قبل از رسیدن به گذرگاه، باید جهت ارزیابی، آماده شده باشد.

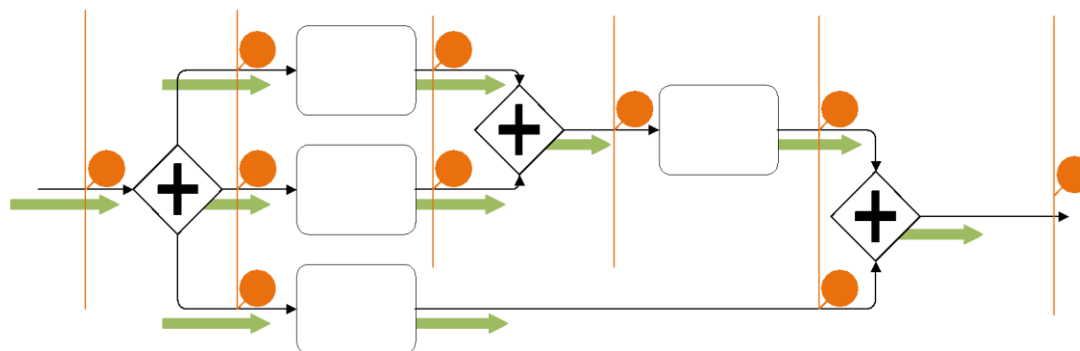
## گذرگاه مبتنی بر قاعده

این گذرگاه‌های دارای انواع تست زیر هستند:

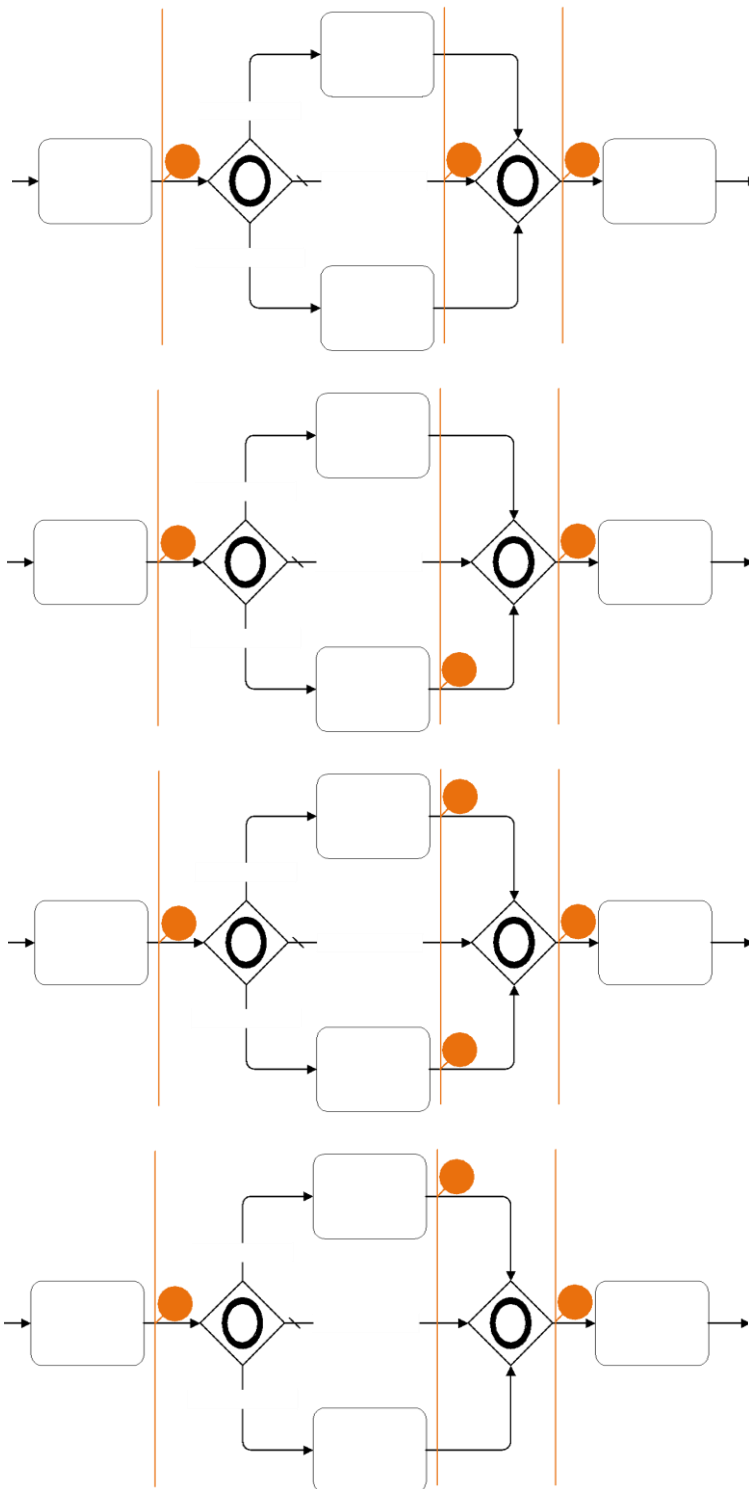
- **Exclusive Data:** این گذرگاه که گذرگاه انتخاب انحصاری یا گذرگاه انحصاری شرطی نیز نامیده می‌شود، کلیه قواعد موجود در جریان‌های ورودی و یا خروجی را مورد بررسی قرار می‌دهد. اولین قاعده که در بررسی، مقدار true را بازگرداند می‌تواند توکن را آزاد کرده (از جریان ورودی) یا دریافت کند (در جریان خروجی). البته پس از گذر یک توکن از این گذرگاه، توکن دیگری نمی‌تواند از این گذرگاه عبور کند. بهتر است اولویت بررسی قواعد در آن رعایت شود. این شرط، مشابه حالت XOR است که با نمادهای {X} یا {◇} نشان داده می‌شوند.



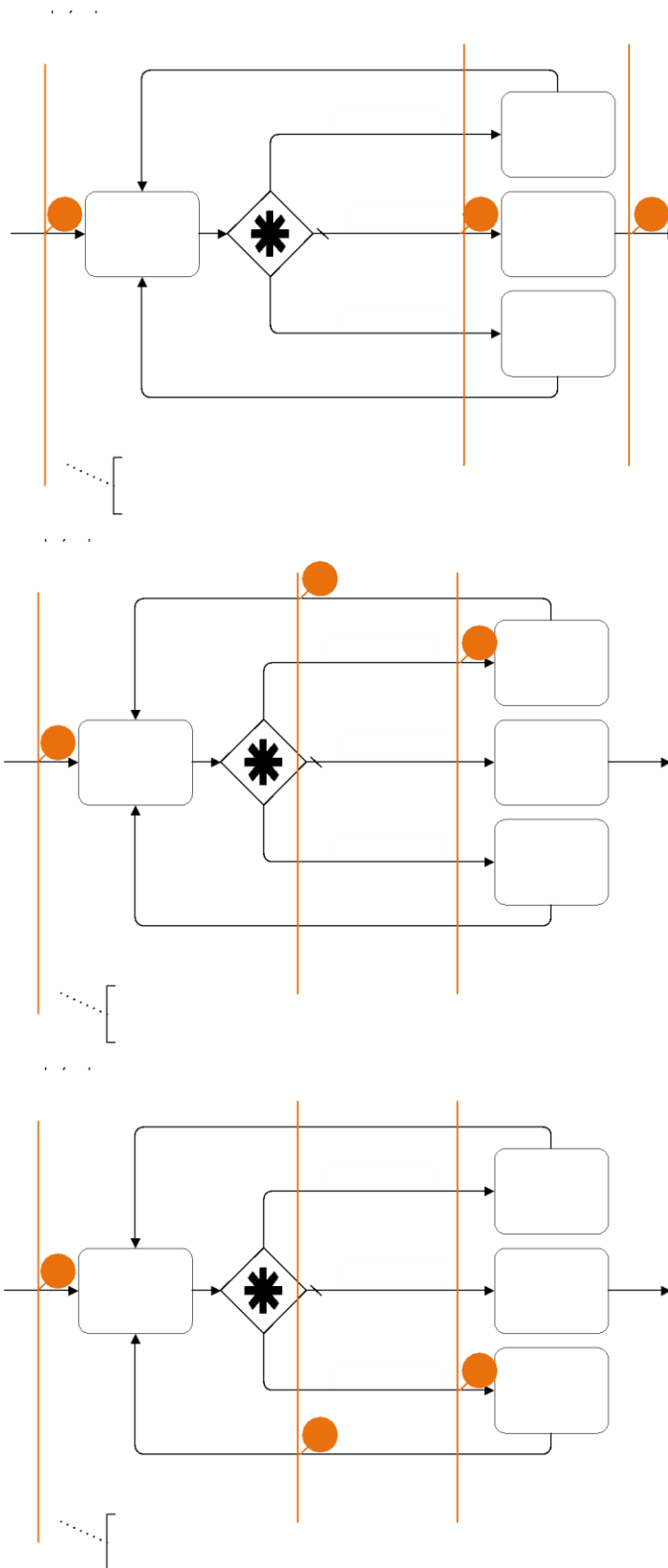
- **Parallel Data:** این گذرگاه که گذرگاه موازی نیز نامیده می‌شود، کلیه قواعد موجود در جریان‌های ورودی و یا خروجی را مورد بررسی قرار می‌دهد. تا زمانی که کلیه قواعد قابل بررسی، مقدار true را بازگرداند منتظر مانده و سپس توکن را آزاد کرده (از جریان ورودی) یا دریافت کند (در جریان خروجی). این شرط، مشابه حالت AND است که با نماد {+} نشان داده می‌شود.



- Inclusive Data:** این گذرگاه که گذرگاه انتخاب اجتماعی نیز نامیده می‌شود، کلیه قواعد موجود در جریان‌های ورودی و یا خروجی را مورد بررسی قرار می‌دهد. هر قاعده که در بررسی، مقدار true را بازگرداند می‌تواند توکن را آزاد کرده (از جریان ورودی) یا دریافت کند (در جریان خروجی). در این گذرگاه، ارتباط یا نگاشت مجموعه ورودی و خروجی اهمیت دارد؛ یعنی باید مشخص شود که کدام ورودی‌ها به کدام خروجی/خروجی‌ها مرتبط هستند. البته برخلاف حالت XOR پس از گذر هر توکن از این گذرگاه، توکن‌های دیگر هم در صورت مناسب بودن نتیجه بررسی، می‌توانند از این گذرگاه عبور کنند. این شرط، مشابه حالت OR است که با نماد  $\{ \bigcirc \}$  نشان داده می‌شود. در تصاویر زیر حالت‌های مختلف این نوع گذرگاه، بررسی شده است.

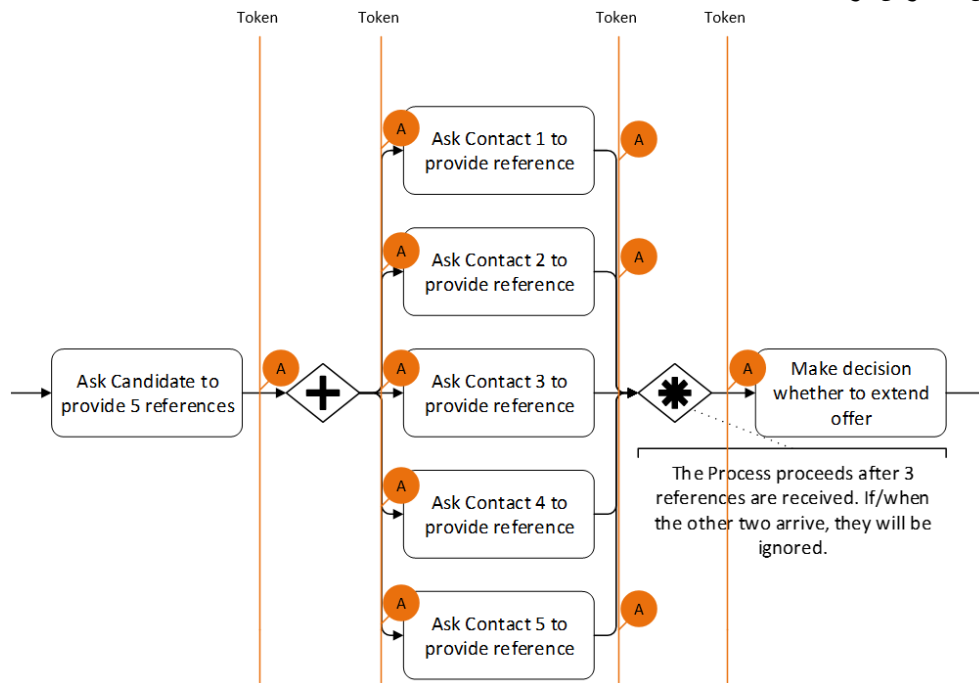


- **Complex Data:** این گذرگاه که گذرگاه پیچیده نیز نامیده می‌شود، قواعد متعدد و پیچیده‌ای را از ورودی‌های خود بررسی می‌کند و با همان ترکیب و اولویتی که طراح مشخص کرده است، عملیات بررسی را صورت می‌دهد. این شرط، مشابه هیچ حالت قاعده‌پذیری نیست و با نماد { \* } نشان داده می‌شود. برای درک بهتر این گذرگاه به تصاویر زیر توجه کنید.





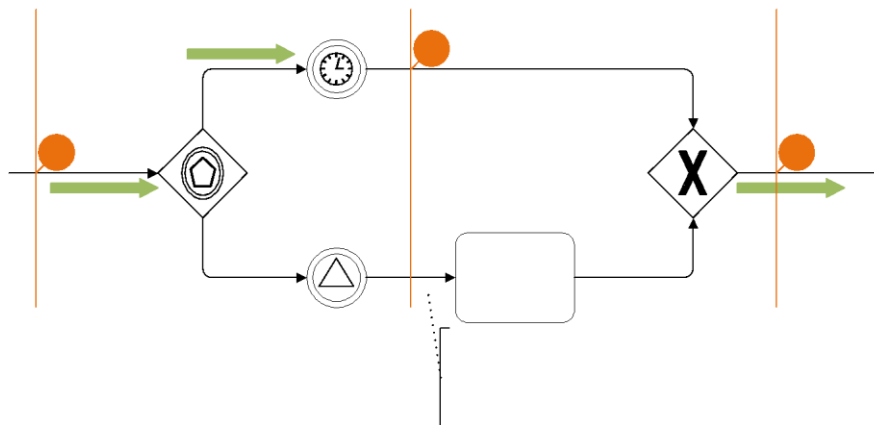
به مثال دیگری در این خصوص توجه کنید:



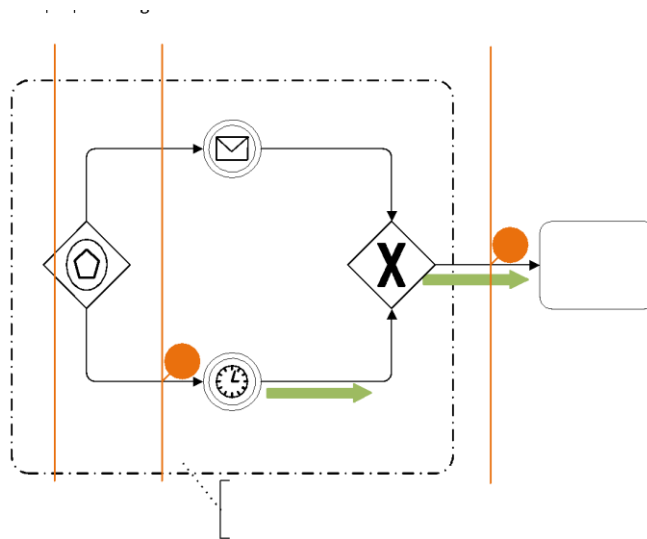
### گذرگاه مبتنی بر رویداد

این گذرگاه‌های دارای انواع تست زیر هستند:

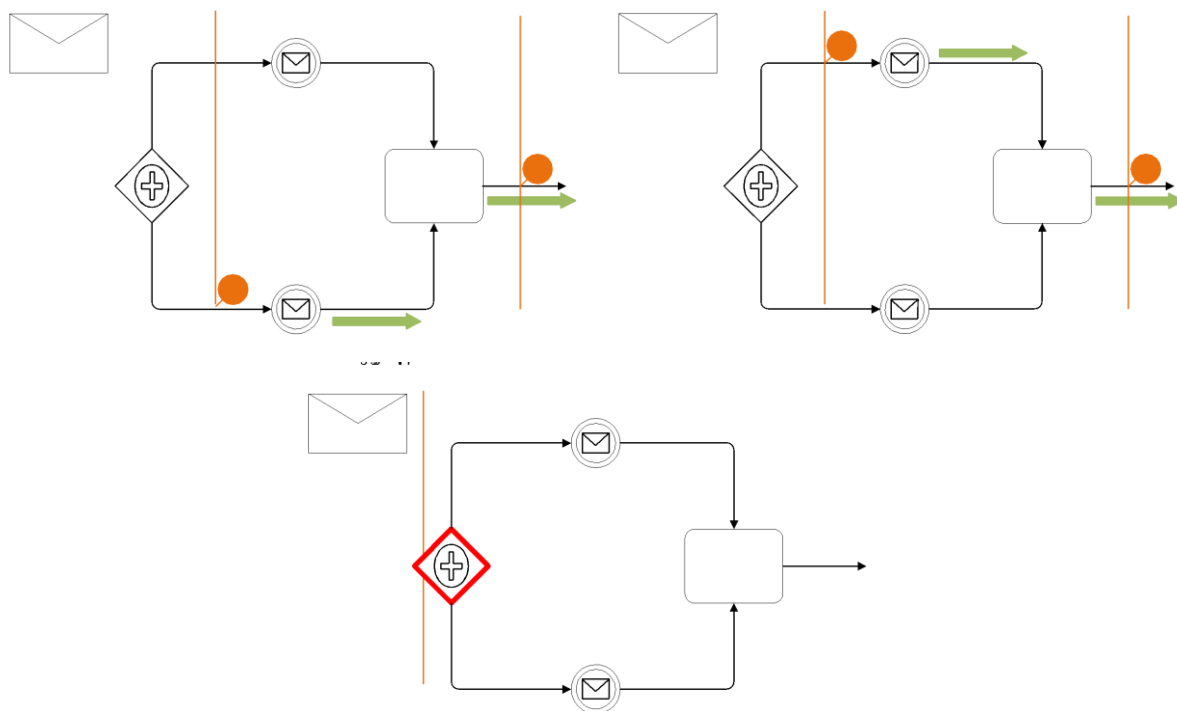
- **Exclusive Event:** این گذرگاه که گذرگاه انتخاب رویداد انحصاری نیز نامیده می‌شود، منتظر مجموعه‌ای از رویدادهای مختلف است و یا به عبارت دیگر این گذرگاه به گونه‌ای طراحی شده است که می‌تواند رویدادهای مختلف را پشتیبانی نماید و بلافاصله پس از تحقق اولین رویداد، توکن را از جریان ورودی آن رویداد به تنها جریان خروجی خود انتقال دهد. پس از گذر یک توکن از این گذرگاه، دیگر توکن دیگری نمی‌تواند از این گذرگاه عبور کند. بهتر است اولویت تحقق رویدادها در آن رعایت شود. البته در نمودارها برای نمایش و تفکیک رویدادهای قابل استفاده، می‌توان و باید در جریان‌های خروجی این نوع گذرگاه بلافاصله یک رویداد دریافت‌کننده میانی از نوعی که مورد نیاز طراح است قرار گیرد. این شرط، مشابه حالت XOR است که با نماد { } نشان داده می‌شود.



- **Exclusive Event for instantiate:** این گذرگاه که گذرگاه انتخاب رویداد انحصاری آغازگر نیز نامیده می‌شود، از نظر عملکرد خیلی شبیه به گذرگاه Exclusive Event است ولیکن هیچ جریان ورودی نخواهد داشت و فقط جریان خروجی دارد. این گذرگاه، می‌تواند یک نود آغازکننده فرآیند باشد. البته در نمودارها برای نمایش و تفکیک رویدادهای قابل استفاده، می‌توان و باید در جریان‌های خروجی این نوع گذرگاه بلافاصله یک رویداد دریافت‌کننده میانی از نوعی که مورد نیاز طراح است قرار گیرد. این شرط، مشابه حالت XOR است که با نماد { } نشان داده می‌شود.



- Parallel Event:** این گذرگاه که گذرگاه رویداد موازی نیز نامیده می‌شود، منتظر مجموعه‌ای از رویدادهای مختلف است و یا به عبارت دیگر این گذرگاه به گونه‌ای طراحی شده است که می‌تواند رویدادهای مختلف را پشتیبانی نماید و تا زمان تحقق هر رویداد منتظر می‌ماند و سپس توکن آن جریان وابسته به رویداد را به جریان خروجی خود انتقال می‌دهد به عبارت دیگر این گذرگاه زمانی کار خود را خاتمه می‌دهد که از تمام جریان‌های وابسته به رویدادها، توکن را به سمت جریان خروجی خود انتقال داده باشد. البته در نمودارها برای نمایش و تفکیک رویدادهای قابل استفاده، می‌توان و باید در جریان‌های خروجی این نوع گذرگاه بلافاصله یک رویداد دریافت‌کننده از نوعی که مورد نیاز طراح است قرار گیرد. این شرط، تقریباً مانند حالت AND است که با نماد {⊕} نشان داده می‌شود. برای درک بهتر این نوع گذرگاه به تصاویر زیر توجه کنید و دقت کنید که کار گذرگاه در این تصاویر زمانی خاتمه می‌یابد که رنگ حاشیه آن قرمز شده باشد.



فعالیت‌ها از جمله با ارزش‌ترین نماد درون یک نمودار فرآیند هستند. در واقع این دسته از نمادها هستند که الگوریتم‌های اصلی یک روال را داخل خود نگهداشته و یا تعریف می‌کنند. همانطور که از تعریف اخیر نیز مشخص است؛ این نمادها باید بخش اعظم رفتار یک فرآیند را تعریف کنند و با توجه به اینکه رفتارها می‌توانند شامل مجموعه متنوعی از الگوها و یا فرم‌ها باشند، طبقه‌بندی این فعالیت‌ها نیز بسیار متنوع خواهد بود. برای ساده‌تر شدن این طبقه‌بندی، سعی خواهیم کرد که این طبقه‌بندی‌ها را در یک ساختار سلسله مراتبی توضیح دهیم. کلیه فعالیت‌ها در دو طبقه اصلی زیر قرار می‌گیرند:

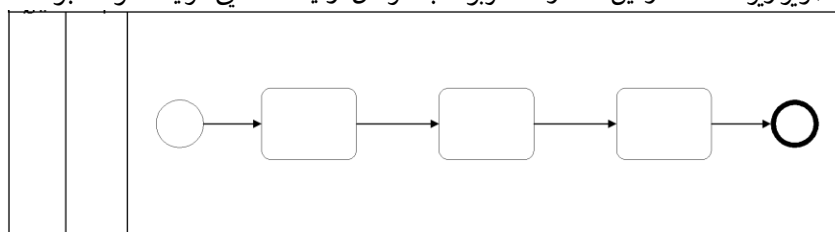
- فعالیت‌های مبتنی بر وظایف
- فعالیت‌های مبتنی بر زیرفرآیندها

هر کدام از انواع فعالیت‌های موجود در طبقه‌بندی ذکر شده، نیز قابل تفکیک هستند. در ادامه به ترتیب جزئیات هر یک از این طبقه‌بندی‌ها توضیح می‌دهیم.

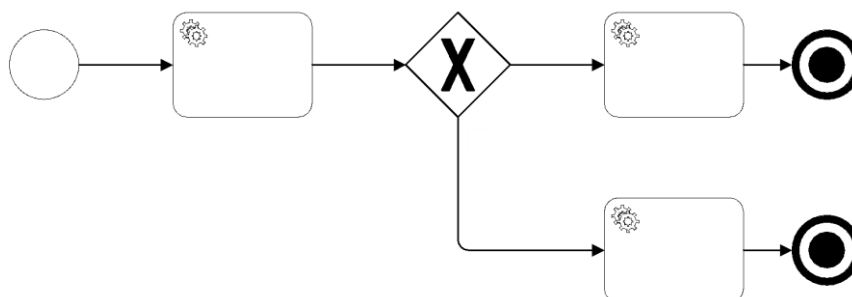
### فعالیت‌های مبتنی بر وظایف

ساده‌ترین و پرکاربردترین نوع فعالیت‌ها که همان وظایف هستند دارای انواع زیر هستند:

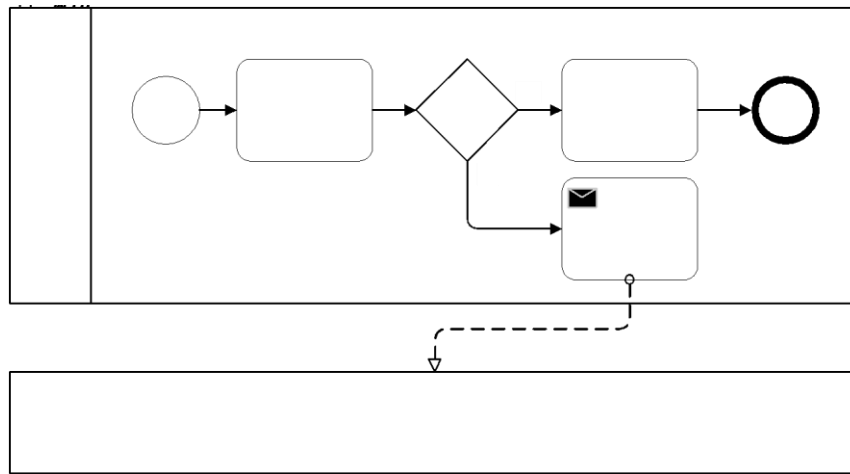
- **Abstract Task:** این وظیفه که وظیفه مفهومی نام دارد محلی است برای بیان یک وظیفه نامشخص و یا ناشناس که قرار است در طی مراحل شناسایی فرآیند، جزئیات بیشتر آن مشخص شود. این وظایف در زمان طراحی، بجز یک نام یا عنوان کسب‌وکاری چیز دیگری ندارند. البته برخی از BPMS ها با این نوع وظیفه، به شما اجازه می‌دهند که وظیفه سفارشی خود را تعریف کنید. این وظایف در نمودارهای مبتنی بر BPMN با نماد { } نشان داده می‌شوند. تصویر زیر استفاده از این نماد را که مربوط به مراحل اولیه شناسایی فرآیند خواهد بود، نشان می‌دهد.



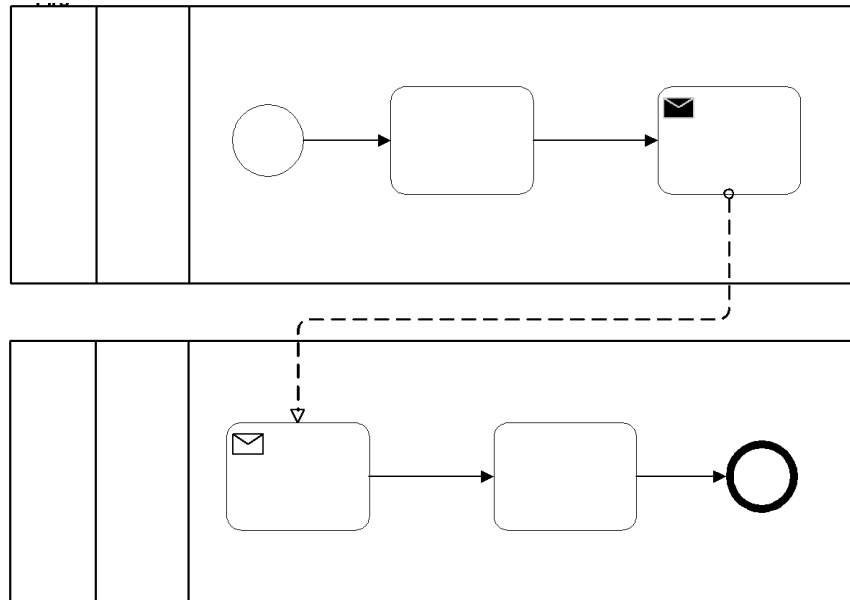
- **Service Task:** این وظیفه که گاهی وظیفه سرویس نیز گفته می‌شود، برای نمایش قسمتی از الگوریتم فرآیند استفاده می‌شود که در آن یک وب سرویس و یا برنامه خودکار دیگر، فراخوانی می‌شود. این وظایف در زمان طراحی، نام سرویس و یا عنوان برنامه خودکار (برنامه‌ای که از دامنه یا فضای فرآیند در حال بررسی، خارج است) را نمایش می‌دهند. این وظایف، تراکنش‌هایی را به واسطه رفتار سرویس‌ها، اجرا می‌کنند که اصولاً به صورت مستقل و غیر قابل تفکیک، اجرا می‌شوند. این وظایف در نمودارهای مبتنی بر BPMN با نماد { } نشان داده می‌شوند. تصویر زیر استفاده از این نماد را نشان می‌دهد.



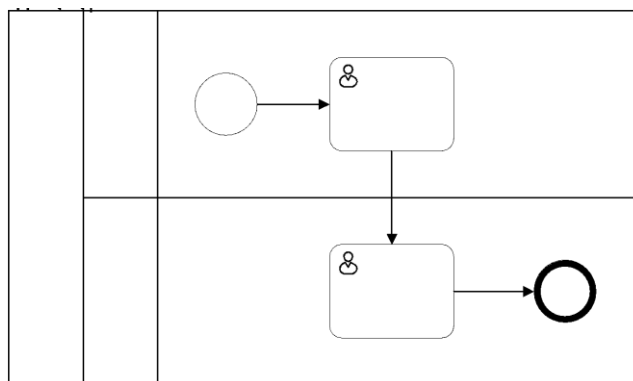
- **Send Task:** این وظیفه که گاهی وظیفه ارسال پیام نیز گفته می‌شود، برای نمایش عمل ارسال یک پیام خاص به خارج از حوزه این فرآیند استفاده می‌شود. به عبارت دیگر زمانیکه از این نوع وظیفه استفاده می‌کنید باید اطمینان داشته باشید که حتماً یک کنترل‌کننده سرویس یا باس کنترلر برای سرویس پیام‌رسان در فضای متعامل خود داشته باشید. این نوع از وظیفه زمانی استفاده می‌شود که ارسال پیام، نیاز به یک بازه زمانی داشته باشد و نمی‌تواند در یک لحظه، آغاز و خاتمه یابد. برخی مواقع BPMS ها خود دارای یک کنترل‌کننده سرویس درونی هستند که به صورت مصطلح به آن کارتابل می‌گویند. رابط کاربری و یا فرم این پیام‌ها یک فرم کاربری خواهد داشت و اصولاً در داخل نمایشگرها، به نمایش در خواهد آمد. این وظایف در نمودارهای مبتنی بر BPMN با نماد { } نشان داده می‌شوند. تصویر زیر استفاده از این نماد را نشان می‌دهد.



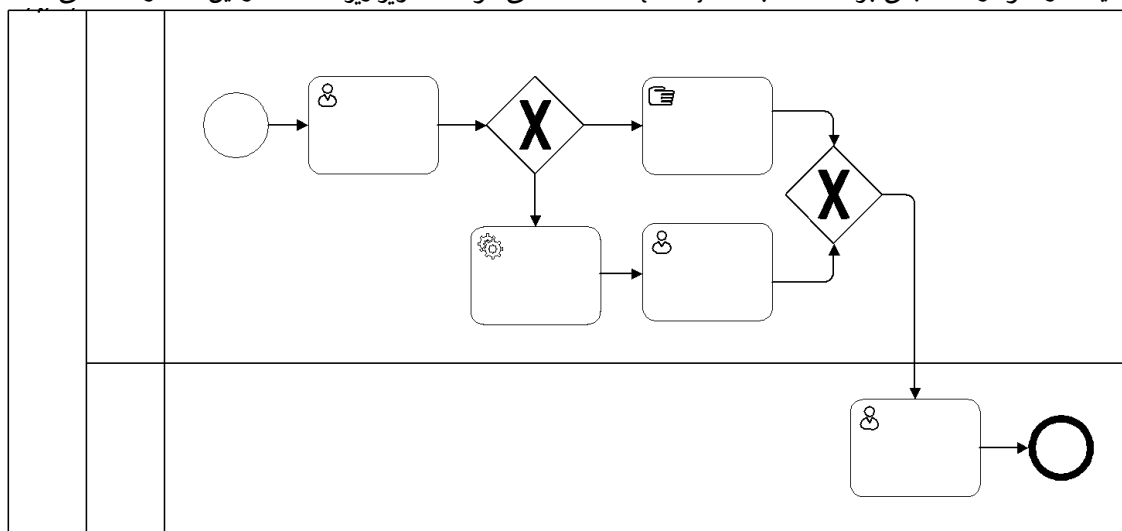
- **Receive Task:** این وظیفه که گاهی وظیفه دریافت پیام نیز گفته می‌شود، برای نمایش عمل دریافت یک پیام خاص از خارج از حوزه این فرآیند استفاده می‌شود. به عبارت دیگر زمانیکه از این نوع وظیفه استفاده می‌کنید باید اطمینان داشته باشید که حتماً یک کنترل‌کننده سرویس یا باس کنترلر برای سرویس پیام‌رسان در فضای متعامل خود داشته باشید. این نوع از وظیفه زمانی استفاده می‌شود که دریافت پیام، نیاز به یک بازه زمانی داشته باشد و نمی‌تواند در یک لحظه، آغاز و خاتمه یابد. برخی مواقع BPMS ها خود دارای یک کنترل‌کننده سرویس درونی هستند که به صورت مصطلح به آن کارتابل می‌گویند. رابط کاربری و یا فرم این پیام‌ها یک فرم کاربری خواهد داشت و اصولاً در داخل نمایشگرها، به نمایش در خواهد آمد. این وظایف در نمودارهای مبتنی بر BPMN با نماد { } نشان داده می‌شوند. تصویر زیر استفاده از این نماد را نشان می‌دهد.



- **User Task:** این وظیفه که گاهی وظیفه کاربری نیز گفته می‌شود، برای نمایش حضور یک فرم کاربری در روند یک فرآیند است که اصولاً توسط یک کاربر انجام می‌شود که البته این نوع از کاربر را عملیات نیمه خودکار می‌گویند. به عبارت دیگر زمانیکه از این نوع وظیفه استفاده می‌کنید باید حتماً آن را به یک نقش کاربری و یا یک کاربر منسوب نمایید. از طرف دیگر این نوع از وظایف، نماینده یک برنامه کاربردی و یا یک فرم از یک برنامه کاربردی خواهند بود. برخی مواقع BPMS ها خود دارای یک فرم‌ساز درونی هستند که به روش ساده‌ای، امکان تولید فرم‌های کاربری را برای طراح، فراهم می‌سازند. این وظایف در نمودارهای مبتنی بر BPMN با نماد { } نشان داده می‌شوند. تصویر زیر استفاده از این نماد را نشان می‌دهد.



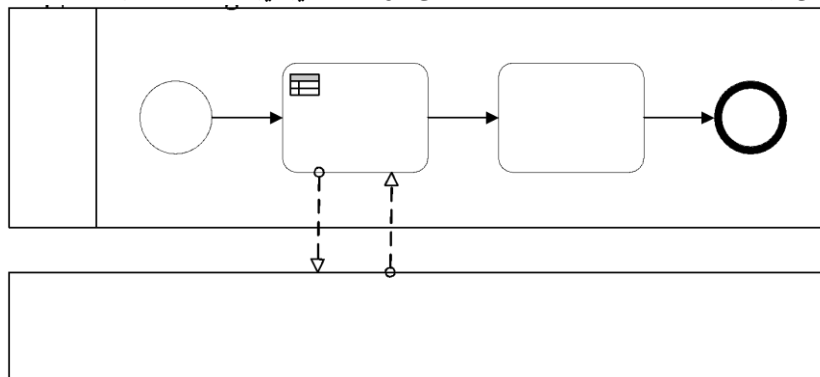
- Manual Task:** این وظیفه که گاهی وظیفه دستی نیز گفته می شود، برای نمایش کار غیر سیستمی است که یک کاربر و یا نقش کاربری (یک نقش انسانی) در روند یک فرآیند انجام خواهد داد. اگر در روند یک فرآیند، از این نوع وظیفه استفاده می کنید و فرآیند طراحی شده شما یک فرآیند سیستمی و یا به بیان دیگر یک فرآیند مکانیزه است؛ حتما به خاطر داشته باشید که نتیجه این وظیفه باید در فرآیند شما یک نمود سیستمی داشته باشد. این وظایف در نمودارهای مبتنی بر BPMN با نماد { } نشان داده می شوند. تصویر زیر استفاده از این نماد را نشان می دهد.



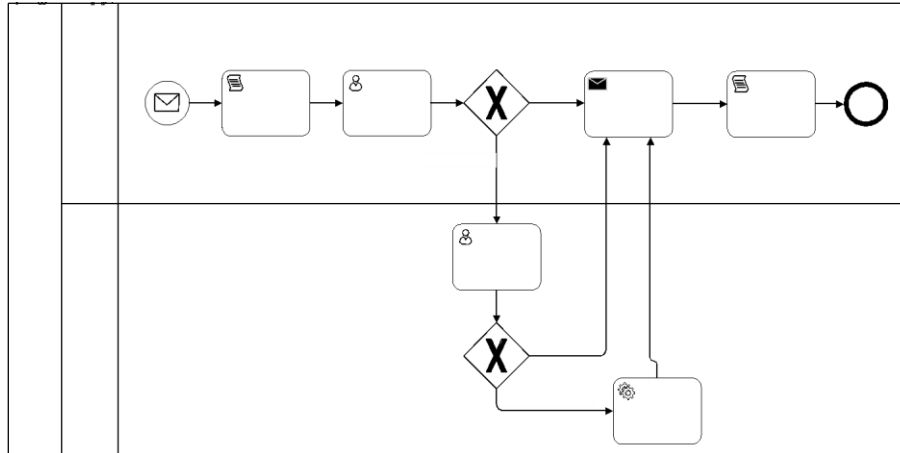
- Business Rule Task:** این وظیفه که گاهی وظیفه بررسی قواعد کسب و کار نیز گفته می شود، به تازگی در نسخه 2 این استاندارد افزوده شده است. این وظیفه برای برقراری ارتباط یک فرآیند با موتور قواعد کسب و کاری<sup>2</sup> است. به عبارت دیگر با استفاده از این نوع وظیفه، هر BPMS این امکان را پیدا می کند که با BRMS ارتباط برقرار کرده و از قواعد درون آن به عنوان تست های مربوط به شرایط مطرح در فرآیند استفاده کند و در نهایت نتیجه بررسی هر تست را که فعالیت خاصی خواهد بود مورد استفاده قرار دهد.

بیان این نکته را ضروری می دانم که اگر چه آموزش روش کار BRMS فارق از BPMS نیز مطرح است ولیکن همواره هر BRMS به یک BPMS تعلق خواهد داشت و یا سازگار خواهد بود.

این وظایف در نمودارهای مبتنی بر BPMN با نماد { } نشان داده می شوند. تصویر زیر استفاده از این نماد را نشان می دهد.

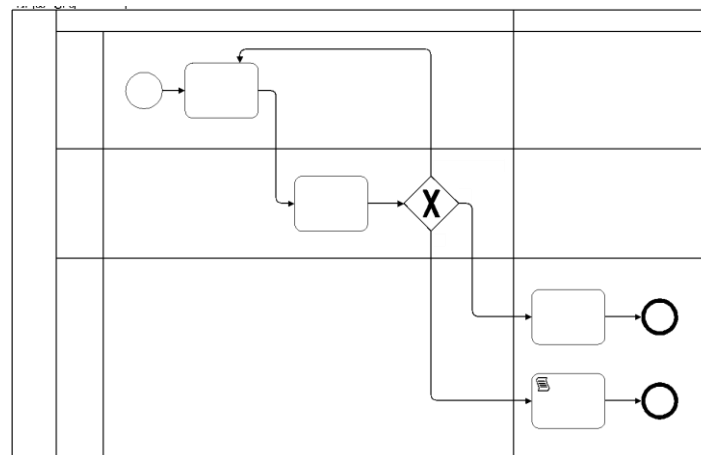


- **Script Task:** این وظیفه که گاهی وظیفه دستوری نیز گفته می‌شود، برای نمایش مجموعه دستورات سیستمی استفاده می‌شود که درون BPMS اجرا خواهند شد. با توجه به مفسر دستوری BPMS مورد استفاده، زبان مورد نیاز در تهیه این دستورات اجرایی نیز متفاوت خواهد بود. خطوط کد که این دستورات را تنظیم می‌کنند فارق از زبان کدنویسی، می‌توانند از متغیرها و پارامترهای BPMS مورد نظر نیز استفاده کنند. این پارامترها برای هر BPMS می‌تواند متفاوت باشد. این وظایف در نمودارهای مبتنی بر BPMN با نماد { } نشان داده می‌شوند. تصویر زیر استفاده از این نماد را نشان می‌دهد.



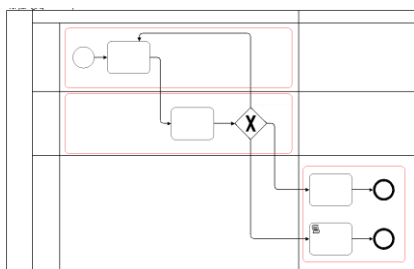
### فعالیت‌های مبتنی بر زیرفرآیندها

نوع دیگری از فعالیت‌ها، زیرفرآیندها هستند که برخی آن را راهی برای تحقق روش‌های فازبندی و طبقه‌بندی‌کننده‌های متقاطع در فرآیند نیز می‌دانند. در برخی از BPMS ها از مفهومی به نام فاز<sup>3</sup> برای اعمال مراحل اجرای یک فرآیند (فارق از نقش‌های کاری) استفاده می‌شود. این فازها، عباراتی هستند که در کسب‌وکار مورد نظر به عنوان اصطلاحات رایج، مورد استفاده خواهند داشت. تصویر زیر نمایی از فازهای Request و Booking را در یک نمودار فرآیند کسب‌وکاری نشان می‌دهد.



زیرفرآیندها جایگزین مناسب و منطقی برای این فازها هستند که در نسخه 2 این استاندارد و مستندات مختلف دیگر، بیشتر پیشنهاد شده است. نکته قابل توجه این است که زیرفرآیندها چهار دسته کلی دارند که در زیر شرحی مختصر بر آنها ذکر می‌شود:

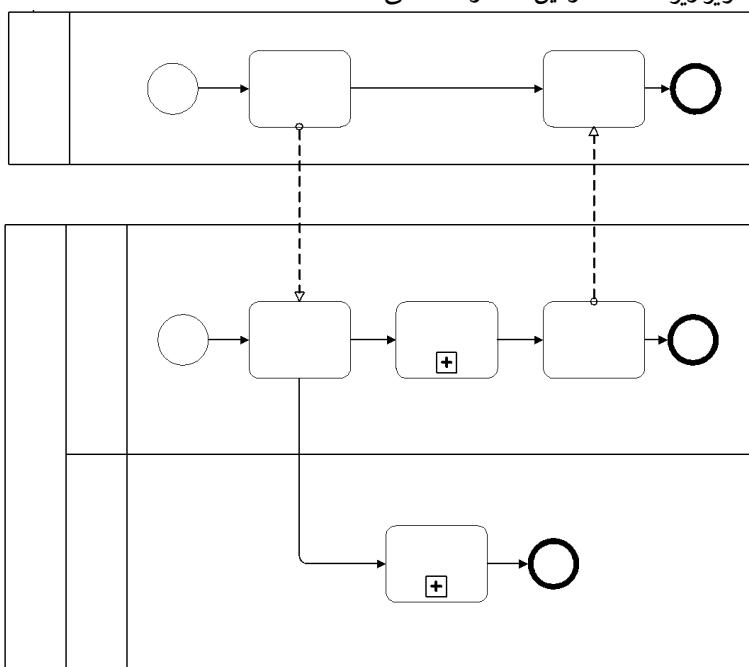
- زیرفرآیندهای وابسته یا همان زیرفرآیندهای درونی، زیرفرآیندهایی هستند که فقط توسط یک نقش کاربری و یا کاربر، دنبال می‌شوند. این نوع از زیرفرآیندها همواره به یک زیرفرآیند و یا فرآیند دیگر وابسته هستند و نمی‌توان از آنها استفاده مجدد کرد. از این زیرفرآیندها بیشتر برای درک بهتر نمودارهای فرآیند و برخی مواقع برای انتساب توسعه به افراد و یا مالکیت توسعه به افراد استفاده می‌شود. در ضمن این زیرفرآیندها همواره با



رویداد آغازین بدون فعال‌کننده، ترسیم می‌شوند. این زیرفرآیندها جایگزین، طبقه‌بندی‌کننده‌های متقاطع در نمودار فرآیند می‌شوند. خطوط قرمز رنگ در تصویر روبرو را شاید بتوان محدوده هر طبقه‌بندی‌کننده متقاطع، تصور کرد. زیرفرآیندهای وابسته دارای انواع زیر هستند:

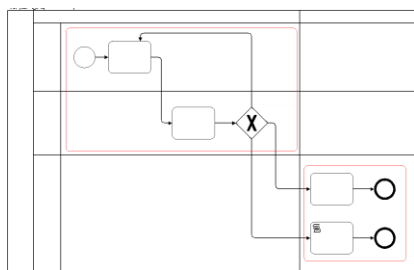
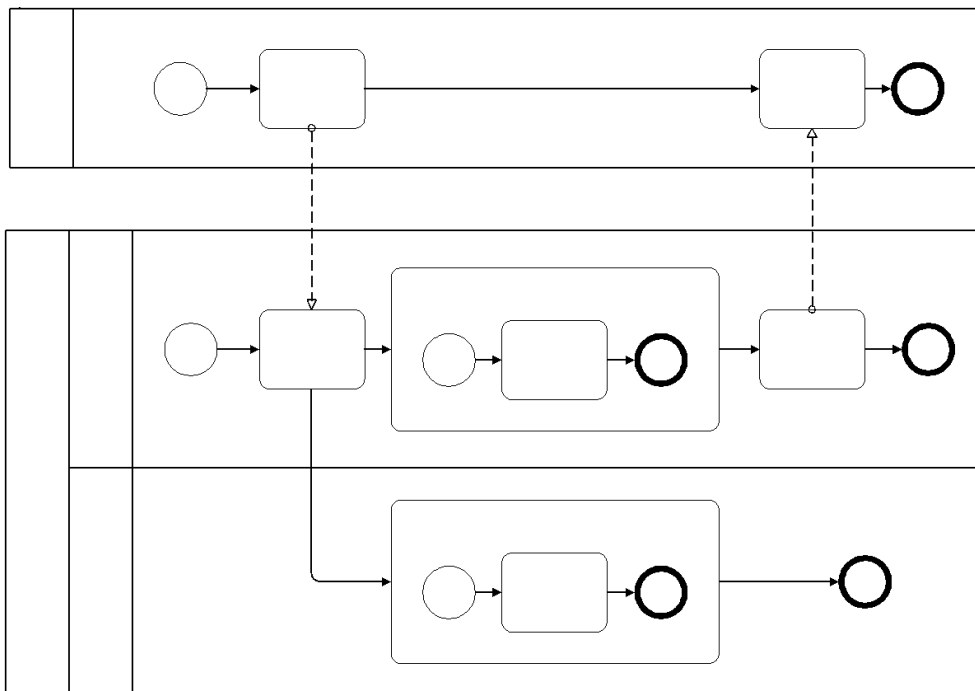
#### ○ **Collapsed Embedded SubProcess:**

گفته می‌شود، برای نمایش محدوده‌ای مبهم از وظایف و زیرفرآیندهای دیگر استفاده می‌شود. یکی از مواردی که در طراحی نمودارهای فرآیندی خیلی اهمیت دارد این است که این نمودارها باید تا حد امکان ساده و قابل درک باشند. در حالیکه برخی از فرآیندهای کسب‌وکاری، بسیار پیچیده هستند. استفاده از زیرفرآیندها راهی برای رفع این پیچیدگی در نمودارهای فرآیندی است. این زیرفرآیندها در نمودارهای مبتنی بر BPMN با نماد {+} نشان داده می‌شوند. تصویر زیر استفاده از این نماد را نشان می‌دهد.




#### ○ **Expanded Embedded SubProcess:**

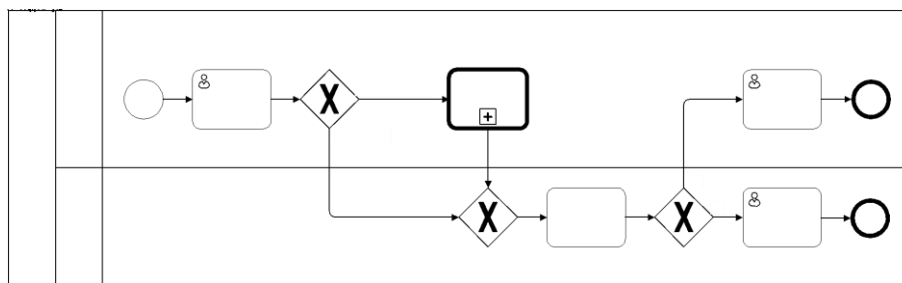
با توضیحات جامع از وظایف و زیرفرآیندهای دیگر، استفاده می‌شود. یک طراح، در اکثر مواقع در طی مراحل شناسایی، تحلیل و طراحی، ابتدا زیرفرآیندها را به صورت پنهان و یا مبهم نمایش می‌دهد و در مراحل بعدی با شناخت بیشتر، نماد زیرفرآیند پنهان را به شکل گسترده نمایش می‌دهد تا معرفی کامل‌تری را ارائه دهد. این زیرفرآیندها در نمودارهای مبتنی بر BPMN با نماد {+} نشان داده می‌شوند. تصویر زیر استفاده از این نماد را نشان می‌دهد.

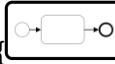


● زیرفرآیندهای مستقل و یا زیرفرآیندهای قابل استفاده مجدد که در اسناد استاندارد به آن زیرفرآیندهای فراخوانی نیز گفته می‌شود. برخلاف نام این زیرفرآیند که شنونده تصور می‌کند این فعالیت باید یک فعالیت ناقص و یا مختصر باشد، این زیرفرآیندها کاملاً مانند فرآیندها هستند ولیکن می‌توان آنها را در هر جایی که لازم باشد، فراخوانی کرد و از نتایج و یا اهداف آنها استفاده کرد. به عبارت دیگر این نوع از زیرفرآیندها، زیرفرآیندهایی هستند که می‌توانند به گونه‌ای طراحی شوند که توسط بیش از یک نقش کاربری و یا کاربر، دنبال شوند. این زیرفرآیندها مانند فرآیندهای دیگر به صورت مستقل طراحی می‌شوند و اکثر BPMS ها آنها را به عنوان دارایی‌های قابل استفاده مجدد ذخیره می‌کنند. این

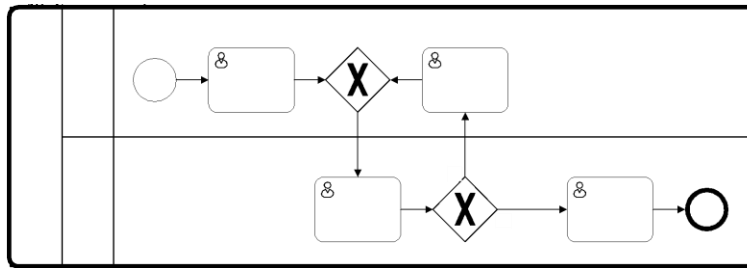
زیرفرآیندها باعث کاهش حجم کار طراحی و به دنبال آن، کاهش هزینه توسعه و پشتیبانی می‌شوند. این زیرفرآیندها جایگزین، فازها در نمودار فرآیند می‌شوند. خطوط قرمز رنگ در تصویر روبرو را شاید بتوان محدوده هر فاز، تصور کرد. زیرفرآیندهای مستقل دارای انواع زیر هستند:

○ **Collapsed Call SubProcess:** این زیرفرآیند که گاهی زیرفرآیندهای فراخوانی نیز گفته می‌شود، برای نمایش فراخوانی یک زیرفرآیند قابل استفاده مجدد، استفاده می‌شود. استفاده مجدد از طراحی در یک پروژه توسعه سیستمی، می‌تواند روشی مرقون به صرفه‌ای برای توسعه‌دهنده باشد. این زیرفرآیندها در نمودارهای مبتنی بر BPMN با نماد {} نشان داده می‌شوند. تصویر زیر استفاده از این نماد را نشان می‌دهد.

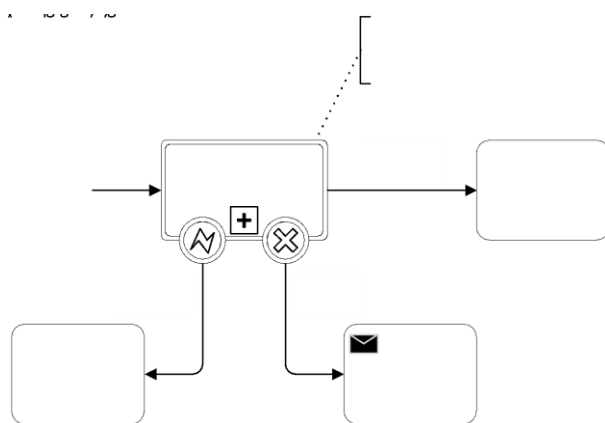


○ **Expanded Call SubProcess:** این زیرفرآیند که گاهی زیرفرآیندهای مرجع نیز گفته می‌شود، برای نمایش یک فرآیند قابل استفاده مجدد با توضیحات جامع از وظایف و زیرفرآیندهای دیگر، استفاده می‌شود. ممکن است وظیفه طراحی و توسعه این زیرفرآیندها به تیم توسعه جانبی سپرده شود و به این زیرفرآیندها به دید ابزارهای کمکی برای توسعه، نگاه شود. این زیرفرآیندها در نمودارهای مبتنی بر BPMN با نماد {} نشان داده می‌شوند. تصویر زیر استفاده از این نماد را نشان می‌دهد.





● زیرفرآیند تراکنش نوع سوم از زیرفرآیندها هستند. منظور از تراکنش، تراکنش‌های پایگاه داده نیست؛ بلکه منظور تراکنش‌های کسب‌وکاری است. تفاوت این تراکنش‌ها با تراکنش‌های پایگاه داده در این است که برعکس آنها، فعالیت‌های درون آنها، قفل نمی‌شوند و در عوض هر فعالیت درونی تراکنش در زمان و مکان مشخص اجرا می‌شود، اما اگر تراکنش به عنوان یک کل با موفقیت به پایان نرسد، هر یک از فعالیت‌های آن که قبلاً تکمیل شده است، با فراخوانی یک فعالیت جبرانی که از نظر زمانی و مکانی به صورت برعکس مسیر اجرا هستند؛ باعث می‌شوند که کل تراکنش، لغو شود. یک زیرفرآیند تراکنش همیشه بخشی از یک فرآیند یا زیرفرآیند سطح بالاتر است. روش فعال‌سازی و یا مدل آغاز یک زیرفرآیند تراکنش مشابه سایر زیرفرآیندها است ولیکن در سمت دیگر این نوع از زیرفرآیندها سه نتیجه یا سه حالت خاتمه وجود خواهد داشت.



همانطور که در تصویر روبرو مشاهده می‌شود، حالت اول، خاتمه موفقیت‌آمیز تراکنش است که در آن تمام فعالیت‌های یک تراکنش با موفقیت انجام شده‌اند، به عبارت دیگر فرآیند تراکنش به طور عادی پیش رفته است. حالت دوم، خاتمه ناموفق تراکنش (لغو تراکنش) است که در صورتی اتفاق می‌افتد که یکی از معیارهای از پیش تعیین‌شده شکست تراکنش، برآورده شود یا پیام abort از خارج از تراکنش دریافت شود. در این حالت جریان غیر عادی بجای جریان عادی از یک رویداد میانی مبتنی بر صرف نظر، اجرا می‌شود و هیچیک از فعالیت‌های درون تراکنش انجام نمی‌شوند. حالت سوم، خطر (استثنا) می‌باشد که در صورتی اتفاق می‌افتد که یک استثنا بوجود آید. در این حالت جریان غیر عادی بجای جریان عادی از یک رویداد میانی مبتنی بر خطا، اجرا می‌شود و هیچیک از فعالیت‌های درون تراکنش انجام نمی‌شوند.

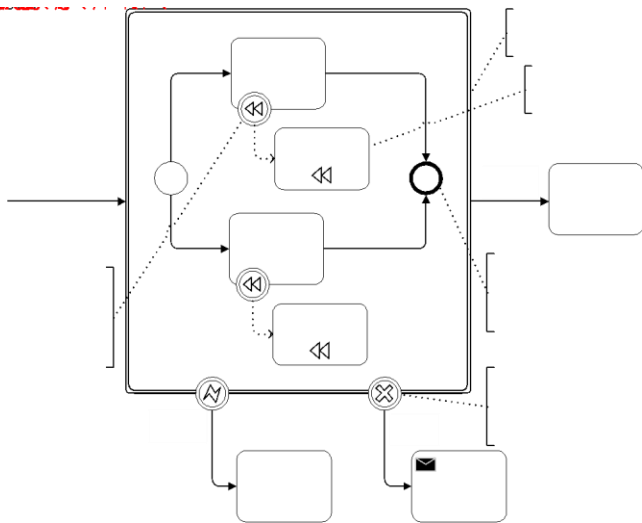
رفتار خاتمه موفقیت‌آمیز در پایان زیرفرآیند تراکنش با رفتار خاتمه یک زیرفرآیند

معمولی متفاوت است. هنگامی که هر یک از مسیرهای زیرفرآیند تراکنش به رویداد پایانی می‌رسد، جریان‌توالی بلافاصله به فرآیند والد سطح بالاتر باز نمی‌گردد. ابتدا، پروتکل تراکنش باید تأیید کند که کلیه بخش‌های تراکنش با موفقیت انجام شده است. اگر این تأیید ارائه نگردد، یک جریان لغو تراکنش یا استثنا ایجاد می‌شود.

از دیدگاه داخلی نیز، یک زیرفرآیند تراکنش با یک زیرفرآیند معمولی متفاوت است، زیرا باید شامل عناصر اضافی باشد که خاص تراکنش می‌باشد. این عناصر اضافه و الزامی در زیر بیان شده‌اند:

- 1) زیرفرآیند تراکنش
- 2) فعالیت جبران یا
- 3) رویداد پیوست‌شده دریافت‌کننده و متوقف‌کننده درخواست انصراف از اجرا
- 4) رویداد پایانی ارسال‌کننده و متوقف‌کننده درخواست انصراف از اجرا
- 5) رویداد آغازین دریافت‌کننده و متوقف‌کننده درخواست بازپایی موارد قبلی
- 6) رویداد پیوست‌شده دریافت‌کننده و متوقف‌کننده درخواست بازپایی موارد قبلی
- 7) رویداد میانی ارسال‌کننده و متوقف‌کننده درخواست بازپایی موارد قبلی
- 8) رویداد پایانی ارسال‌کننده و متوقف‌کننده درخواست بازپایی موارد قبلی

تصویر روبرو یک نمونه زیرفرآیند تراکنش را نشان می‌دهد که قبلاً فقط نمای خارجی آن را ترسیم کردیم. نمای داخلی این نمونه نشان می‌دهد که یک زیرفرآیند تراکنش یک زیرفرآیند است که از چندین فعالیت اصلی مختلف تشکیل شده است. علاوه بر این فعالیت‌ها، که یک جریان عادی را نشان می‌دهند، یک زیرفرآیند تراکنش شامل فعالیت‌های جبرانی ویژه نیز می‌باشد. این فعالیت‌ها بخشی از یک جریان عادی نیستند و نشان دهنده



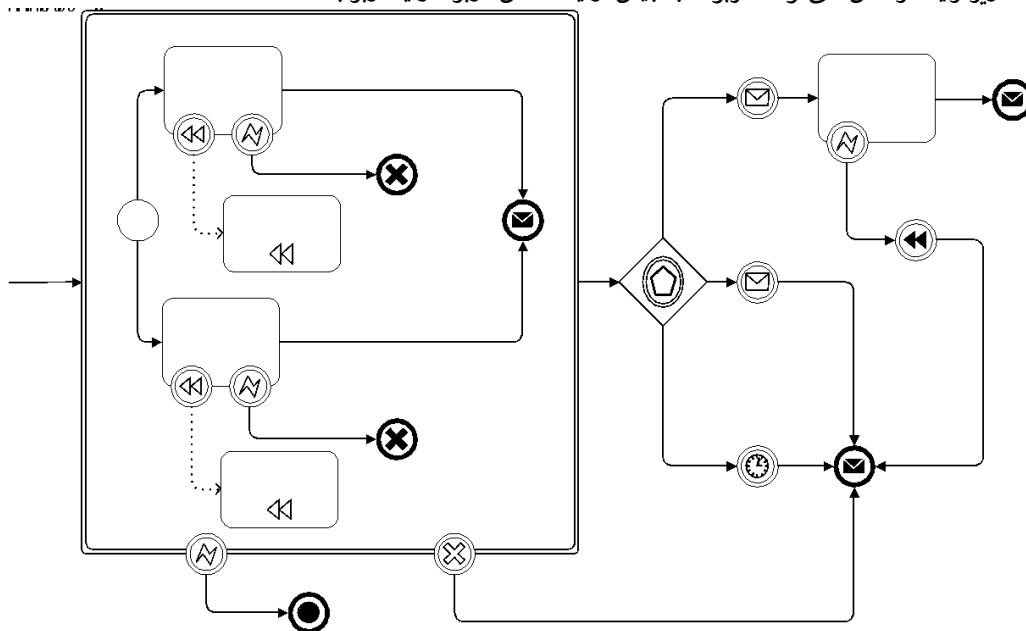
جبران یا بازگشت تراکنش هستند. یک فعالیت جبرانی به یک فعالیت عادی در جریان متناظر متصل می‌شود که این اتصال به واسطه یک ارتباط جبرانی (فلش‌های نقطه‌چین‌دار) ایجاد می‌شود. یک ارتباط جبرانی خارج از جریان عادی فرآیند اتفاق می‌افتد.

همانطور که قبلاً هم توضیح داده شد اصطلاح جبران مربوط به بازگرداندن مراحل است که قبلاً با موفقیت انجام شده‌اند، زیرا نتایج و احتمالاً عوارض جانبی آنها، دیگر مورد نظر نیستند و باید معکوس شوند. اگر فعالیتی همچنان فعال باشد، نمی‌توان آن را جبران کرد، بلکه باید لغو شود. در صورت وجود یک فرآیند فرعی، عمل لغو به نوبه خود می‌تواند منجر به جبران بخش‌هایی از یک فعالیت فعال شود که قبلاً با موفقیت انجام شده است.

نمودار بعدی بر اساس نمودار قبلی است. این نمودار شامل

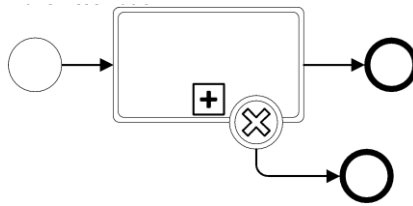
مکانیزمی است که به مشتری امکان می‌دهد تراکنش را پس از انجام موفقیت آمیز، لغو کند. در این حالت، رویداد لغو نمی‌تواند برای خنثی کردن تراکنش استفاده شود، زیرا رویداد ارسال‌کننده لغو باید در درون زیرفرآیند تراکنش قرار گیرد. در این نمودار با استفاده از یک رویداد جبران ارسال‌کننده (Undo Bookings)، که می‌تواند خارج از تراکنش قرار گیرد، این مشکل را حل می‌کنیم. رویداد جبران ارسال‌کننده تمام فعالیت‌های جبران را در زیرفرآیند تراکنش آغاز می‌کند.

نکته بسیار مهم در خصوص زیرفرآیندهای تراکنش این است که در این زیرفرآیندها کل روال اجرای کار مربوط به یک نقش کاربری و یا کاربر است و جریان‌توالی درونی یک زیرفرآیند تراکنش نمی‌تواند مربوط به بیش از یک نقش کاربری و یا کاربر باشد.



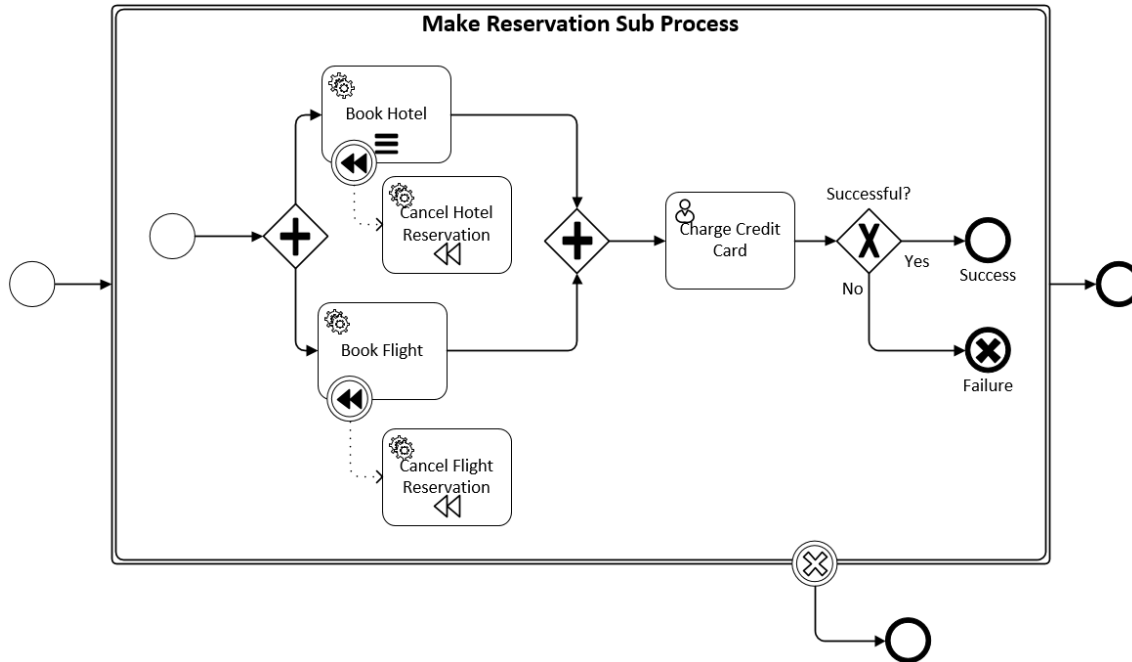
زیرفرآیندهای تراکنش دارای انواع زیر هستند:

○ **Collapsed Transaction SubProcess:** این زیرفرآیند که گاهی تراکنش پنهان نیز گفته می‌شود، برای نمایش یک فعالیت پیچیده ولی کپسوله شده مورد استفاده قرار می‌گیرد. ممکن است در قسمتی از روال یک فرآیند، نیاز به اجرای چندین رفتار مختلف توسط چندین نقش مختلف باشد و یا در مراحل مختلف زیرمحصولات مختلفی تولید شود که در کنار هم یک نتیجه (اصلی) حاصل نمایند. اگر حصول به نتیجه مورد نظر، الزامی باشد و نتیجه مورد نظر نیاز به تمام اجرای وابسته خود باشد؛ وضعیت خواهد بود که به آن یک تراکنش می‌گوییم. این زیرفرآیندها در نمودارهای مبتنی بر BPMN با نماد { } نشان داده می‌شوند. تصویر زیر استفاده از این نماد را نشان می‌دهد.

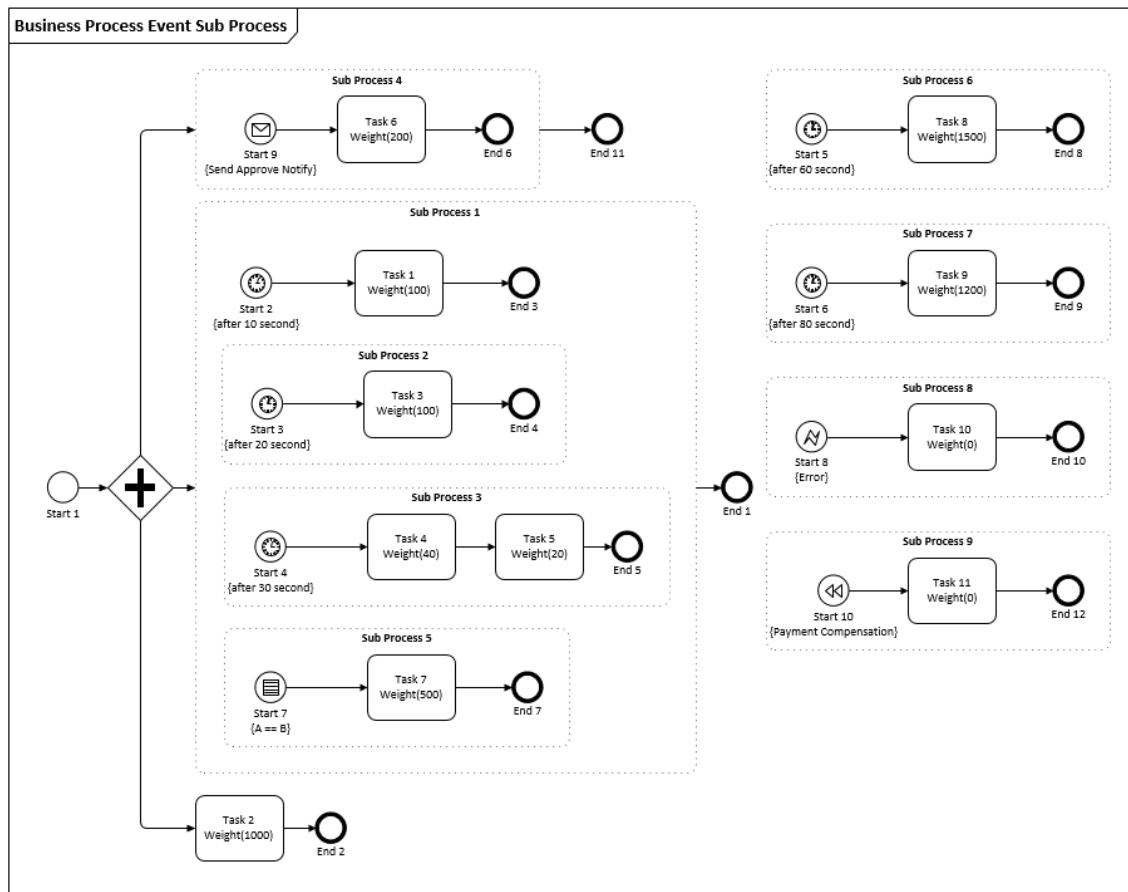


○ **Expanded Transaction SubProcess:** این زیرفرآیند که گاهی زیرفرآیند تراکنش تشریحی نیز گفته می‌شود، برای نمایش گسترده یک

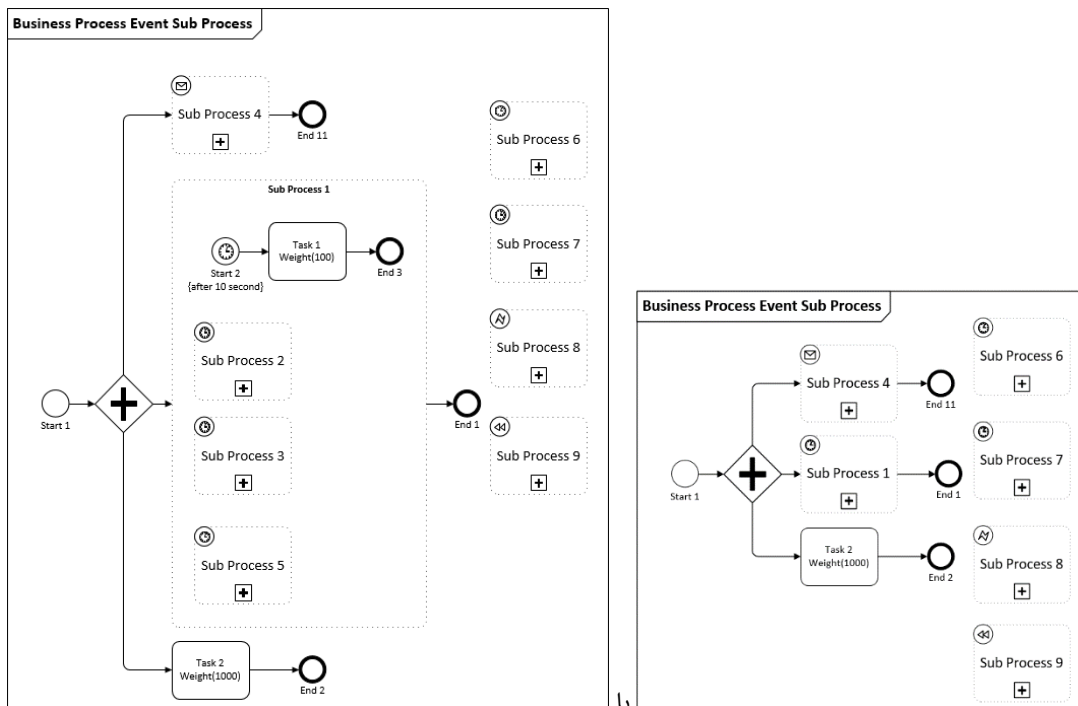
زیرفرآیند از نوع تراکنش پنهان، استفاده می‌شود. این زیرفرآیندها در نمودارهای مبتنی بر BPMN با نماد { } نشان داده می‌شوند. تصویر زیر استفاده از این نماد را نشان می‌دهد.



● زیرفرآیند رویداد نوع چهارم زیرفرآیندها هستند. این نوع زیرفرآیند، زیرفرآیندی است که فعال‌کننده آن یک رویداد است. این زیرفرآیندها می‌توانند در جریان توالی یک فرآیند قرار گیرند و یا به صورت سراسری و یا محلی در فرآیند حضور داشته باشند و بر اثر حدوث رویداد مورد نظر، آغاز شوند و روالی که درون خود دارند را اجرا کنند. کسانی که با زیرفرآیندهای رویداد آشنا هستند بیشتر آنها را با رویدادهای زمان، پیام و خطا، مرتبط می‌دانند ولیکن این زیرفرآیندها با تمام رویدادهای شناخته شده در این استاندارد قابل استفاده هستند. تصویر زیر نمایی از کاربرد زیرفرآیندهای رویداد را در یک نمودار فرآیند فرضی نشان می‌دهد.



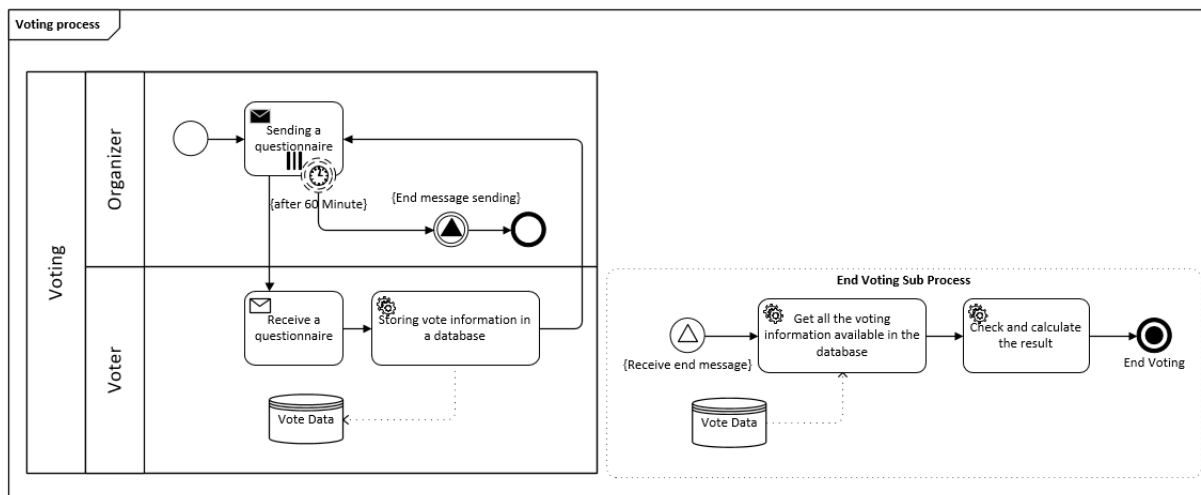
در تصویر فوق سه شکل از کاربرد زیرفرآیند رویداد را مشاهده می‌کنید. شکل اول، زیرفرآیندهایی هستند که به صورت سراسری در یک محیط عملیات فرآیند، تعریف می‌شوند. با توجه به اینکه این زیرفرآیندها به هنگام وقوع یک رویداد، فعال می‌شوند، کلیه بررسی‌ها و معیارهای وقوع رویدادها از زمان آغاز فرآیند مذکور و کلیه پارامترها نیز از محیط همین فرآیند، مورد ارزیابی قرار خواهند گرفت. به عبارت دیگر زمانیکه رویداد Start 1 وقوع می‌یابد (فرض بر این است که فرآیند والد آغاز می‌شود)، بعد از 60 ثانیه، زیرفرآیند رویداد Sub Process 6 و بعد از 80 ثانیه Sub Process 7 آغاز می‌شود. در همین حوزه مورد بررسی، چنانچه خطایی در فرآیند والد ایجاد شود زیرفرآیند رویداد Sub Process 8 و اگر رویداد جبرانی ارسال شود Sub Process 9 آغاز می‌شود. آغاز این چهار زیرفرآیند رویداد هیچ ارتباطی به جریان‌توالی فرآیند ندارد و شرط آغاز اجرای آنها، وقوع رویداد Start 1 است. شکل دوم، زیرفرآیندهایی هستند که به صورت مستقیم در جریان‌توالی فرآیند والد قرار دارند مثلاً زمانیکه توکن اجرای فرآیند والد از گذرگاه Parallel Data عبور کرد و به واسطه آن تبدیل به 3 توکن شد؛ چنانچه یک اعلام پیام تأیید دریافت شود، زیرفرآیند رویداد Sub Process 4 و چنانچه 10 ثانیه از زمان گذر توکن از گذرگاه Paralle Data بگذرد زیرفرآیند رویداد Sub Process 1 اجرا می‌شود. شکل سوم، زیرفرآیندهایی هستند که به صورت محلی درون یک زیرفرآیند، تعریف می‌شوند. کلیه بررسی‌ها و معیارهای وقوع رویدادها از زمان آغاز زیرفرآیند مذکور و کلیه پارامترها نیز از محیط همین زیرفرآیند، مورد ارزیابی قرار خواهند گرفت. به عبارت دیگر زمانیکه رویداد Start 2 وقوع می‌یابد (فرض بر این است که زیرفرآیند رویداد Sub Process 1 آغاز می‌شود)، بعد از 20 ثانیه، زیرفرآیند رویداد Sub Process 2 و بعد از 30 ثانیه Sub Process 3 آغاز می‌شود. در همین حوزه مورد بررسی، چنانچه مقدار پارامتر A با مقدار پارامتر B که در محیط Sub Process 1 در دسترس هستند با هم برابر شود زیرفرآیند رویداد Sub Process 5 آغاز می‌شود. البته اگر تصویر بالا را به صورت خلاصه نمایش دهیم چیزی شبیه به تصاویر زیر خواهد بود.



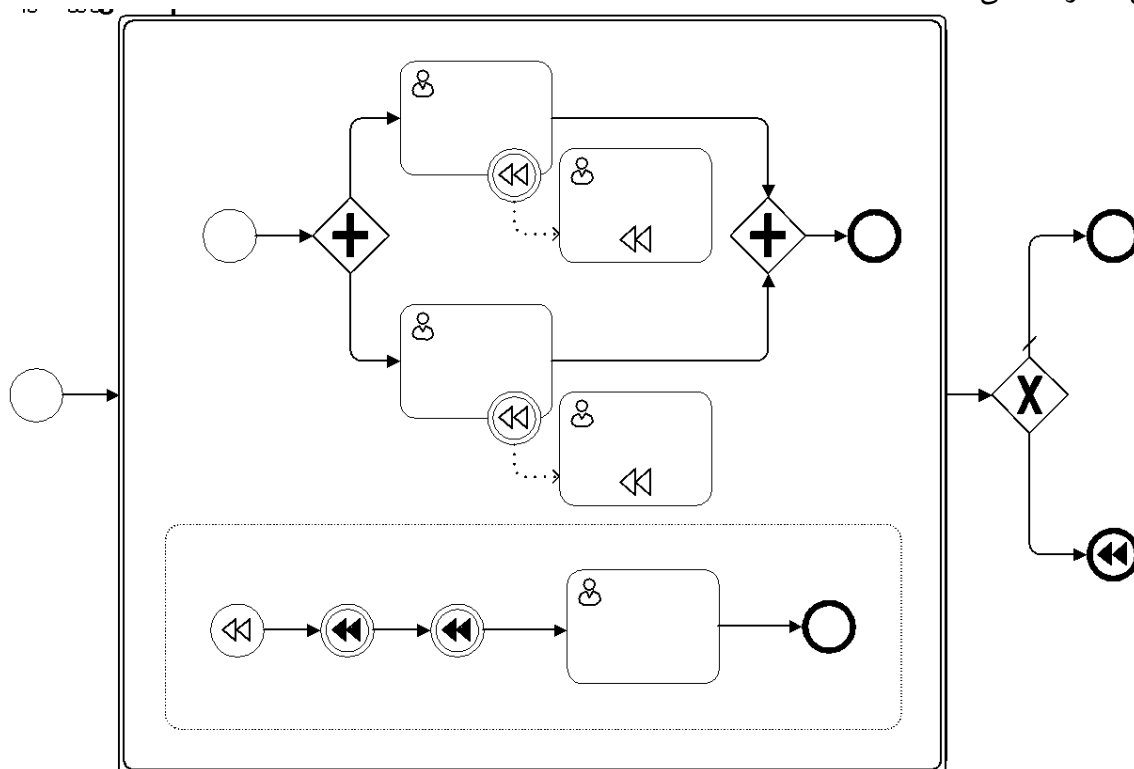
یا

در بخش رویدادها، ذکر شده است که رویدادها می‌توانند متوقف‌کننده و یا بدون توقف باشند. این خصوصیت رویدادها نیز در زیرفرآیندهای رویداد مورد استفاده دارد. به عبارت دیگر، چنانچه در یک زیرفرآیند رویداد، رویداد وابسته به آن یک رویداد متوقف‌کننده باشد و این زیرفرآیند رویداد در مسیر یک جریان‌توالی قرار داشته باشد؛ اگر توکن اجرای فرآیند به این زیرفرآیند برسد، زیرفرآیند مورد نظر تا زمانیکه رویداد وابسته به آن وقوع نیابد، حرکت توکن مورد نظر را در جریان‌توالی متوقف می‌کند و انتظار برای تحقق رویداد وابسته، آغاز می‌شود. پس از وقوع رویداد وابسته، زیرفرآیند رویداد مذکور اجرا شده و پس از خاتمه اجرای آن، توکن به جریان خروجی آن انتقال می‌یابد. ولیکن اگر رویداد وابسته به زیرفرآیند رویداد یک رویداد بدون توقف باشد و این زیرفرآیند رویداد در مسیر یک جریان‌توالی قرار داشته باشد؛ اگر توکن اجرای فرآیند به این زیرفرآیند برسد، حرکت توکن مورد نظر در جریان‌توالی ادامه یافته و به جریان خروجی این زیرفرآیند انتقال می‌یابد و از طرف دیگر، با ایجاد یک توکن جدید انتظار برای تحقق رویداد وابسته آغاز می‌شود. پس از وقوع رویداد وابسته، زیرفرآیند رویداد مذکور اجرا شده و پس از خاتمه اجرای آن، توکن وابسته به آن آزاد می‌گردد. بدیهی است که زیرفرآیندهای رویداد سراسری و یا محلی (زیرفرآیندهای رویدادی که به طور مستقیم با یک جریان‌توالی ارتباط ندارند) از این قاعده مستثنی هستند. فارق از اینکه زیرفرآیند رویداد را می‌توان به شکل پنهان یا گسترده نمایش داد این زیرفرآیندها دارای انواع زیر نیز خواهند بود:

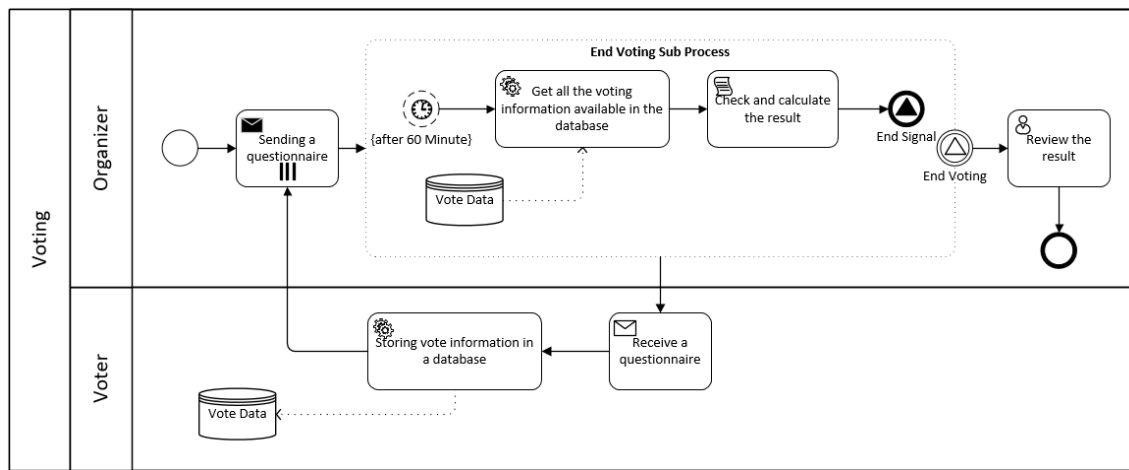
- **Global Event SubProcess:** این زیرفرآیند که گاهی زیرفرآیند رویداد عمومی نیز گفته می‌شود، برای نمایش فعالیت‌هایی که باید به صورت اتفاقی پس از وقوع رویدادهای خاصی در زمان اجرای یک فرآیند آغاز شوند، استفاده می‌شود. این زیرفرآیندها هیچ جریان‌توالی خروجی و یا ورودی ندارند و رویداد آغاز آنها یک رویداد غیرمفهومی (غیر خالی و یا مشخص شده) متوقف‌کننده است. این زیرفرآیندها در نمودارهای مبتنی بر BPMN با نمادهایی مانند آنچه در جدول پایان همین گفتار قرار دارد، نشان داده می‌شوند. تصویر زیر استفاده از این نماد را نشان می‌دهد.



Local Event SubProcess: این زیرفرآیند که گاهی زیرفرآیند رویداد محلی نیز گفته می‌شود، برای نمایش فعالیت‌هایی که باید به صورت اتفاقی پس از وقوع رویدادهای خاصی در زمان اجرای یک زیرفرآیند آغاز شوند، استفاده می‌شود. این زیرفرآیندها هیچ جریان‌توالی خروجی و یا ورودی ندارند و رویداد آغاز آنها یک رویداد غیرمفهومی (غیر خالی و یا مشخص شده) متوقف‌کننده است. این زیرفرآیندها در نمودارهای مبتنی بر BPMN با نمادهایی مانند آنچه در جدول پایان همین گفتار قرار دارد، نشان داده می‌شوند. تصویر زیر استفاده از این نماد را نشان می‌دهد.



Sequence Flow Event SubProcess: این زیرفرآیند که گاهی زیرفرآیند رویداد جریان نیز گفته می‌شود، برای نمایش فعالیت‌هایی که باید در جریان‌توالی یک فرآیند پس از وقوع رویدادهای خاصی آغاز شوند، استفاده می‌شود. این زیرفرآیندها دارای جریان‌توالی خروجی و یک جریان‌توالی ورودی هستند و رویداد آغاز آنها یک رویداد غیرمفهومی (غیر خالی و یا مشخص شده) است که هم می‌تواند متوقف‌کننده باشد و هم می‌تواند بدون توقف باشد. این زیرفرآیندها در نمودارهای مبتنی بر BPMN با نمادهایی مانند آنچه در جدول پایان همین گفتار قرار دارد، نشان داده می‌شوند. تصویر زیر استفاده از این نماد را نشان می‌دهد.



البته رویدادهایی که می‌توانند در این زیرفرآیندهای رویداد مورد استفاده قرار گیرند شامل موارد زیر هستند:

نماد	رویداد مرتبط	نماد	رویداد مرتبط	نماد	رویداد مرتبط	نماد	رویداد مرتبط
	متوقف کننده مبتنی بر پیام		متوقف کننده مبتنی بر خطا		متوقف کننده مبتنی بر قواعد		متوقف کننده مبتنی بر چند فعال کننده به صورت انتخابی
	بدون توقف مبتنی بر پیام		بدون توقف مبتنی بر جبران		بدون توقف مبتنی بر قواعد		بدون توقف مبتنی بر چند فعال کننده به صورت انتخابی
	متوقف کننده مبتنی بر زمان		متوقف کننده مبتنی بر سطح اجرا		متوقف کننده مبتنی بر سیگنال		متوقف کننده مبتنی بر چند فعال کننده به صورت موازی
	بدون توقف مبتنی بر زمان		بدون توقف مبتنی بر سطح اجرا		بدون توقف مبتنی بر سیگنال		بدون توقف مبتنی بر چند فعال کننده به صورت موازی

نکته: تمام زیرفرآیندها فقط می‌توانند یک رویداد آغاز داشته باشند که البته این رویداد آغاز نیز باید از نوع بدون فعال کننده (خالی) باشد.

فارق از طبقه‌بندی اخیر برای فعالیت‌ها، نوعی طبقه‌بندی دیگر که از نظر مفهومی یک طبقه‌بندی ثانویه محسوب می‌شود نیز برای فعالیت‌ها وجود دارد که در زیر به آن می‌پردازیم.

### طبقه‌بندی ثانویه فعالیت‌ها

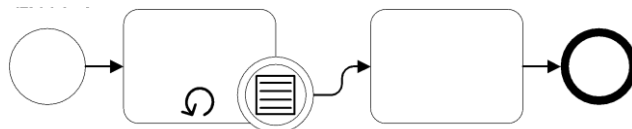
فارق از اینکه فعالیت مورد ارزیابی یک وظیفه است و یا یک زیرفرآیند، می‌تواند الگوی رفتاری مشابه موارد زیر داشته باشد:

- **Loop:** این دسته از فعالیت‌ها، فعالیت‌هایی هستند که یک حلقه تکرار در برنامه‌نویسی را نشان می‌دهند. در یک حلقه ابتدا یک وظیفه یا زیرفرآیند، اجرا می‌شود و سپس شرایط خروج از حلقه مورد ارزیابی قرار می‌گیرد. توجه داشته باشید که ارزیابی مورد نظر یک عبارت منطقی است که اطلاعات فرآیندی<sup>4</sup> را مورد بررسی قرار می‌دهد و ارتباطی به اطلاعات کسب‌وکاری<sup>5</sup> ندارد. اگر نتیجه بررسی، مقدار false باشد، فعالیت مورد نظر دوباره اجرا می‌شود و سپس یک بار دیگر شرایط ارزیابی می‌شود. این روند تا زمانی ادامه می‌یابد که نتیجه بررسی مورد نظر مقدار true باشد که در آن صورت، توکن اجرای فرآیند از جریان‌توالی خروجی عادی فعالیت، خارج می‌شود. برای فعالیت‌هایی که ممکن است برای حصول نتیجه مناسب، نیاز به تلاش‌های متعدد داشته باشید، این الگو بسیار مفید است. برای این الگو، امکان تنظیم حداکثر تعداد تکرار مجاز وجود دارد. این الگو در واقع جایگزینی برای یک فعالیت غیر حلقه‌ای است که در ادامه جریان‌توالی خروجی خود یک گذرگاه Exclusive یا XOR دارد که یکی از جریان‌های توالی خروجی همان گذرگاه نیز به فعالیت مورد نظر بازمی‌گردد. ترسیم یک نمای حلقه تکرار به روش اخیر، باعث پیچیدگی نمودار می‌شود. از طرف دیگر جایگزین کردن یک فعالیت با طبقه‌بندی کننده Loop نیز باعث می‌شود جزئیات خروج از حلقه، پنهان بماند. به همین منظور، معرف‌های خاصی که هر کدام مدل‌های مختلفی از شرایط اجرا و خروج را نمایش می‌دهند، در نظر گرفته شده است. این معرف‌ها دارای انواع مختلف زیر است:

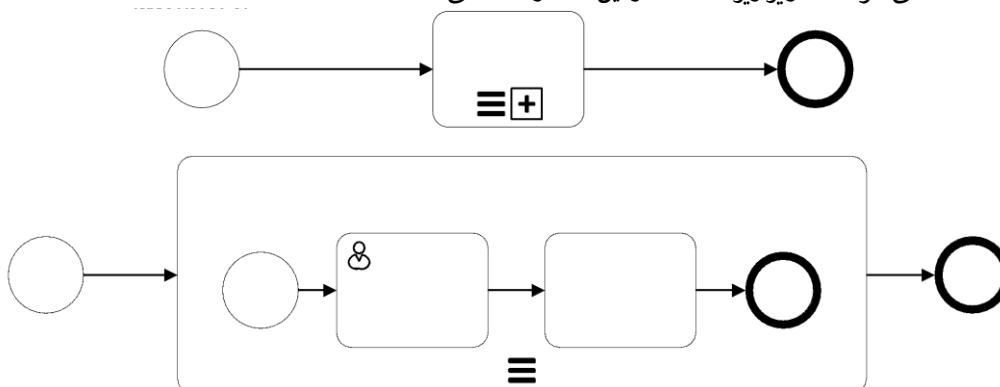
<sup>4</sup> آن دسته از اطلاعات هستند که روند اجرای فرآیند را کنترل می‌کنند و هرگز به عنوان یک داده با ارزش کسب‌وکاری ذخیره نمی‌شوند.

<sup>5</sup> آن دسته از اطلاعات هستند که دارای ارزش کسب‌وکاری هستند و بیشتر در مخازن اطلاعاتی ذخیره شده و ممکن است توسط سیستم‌های دیگر در هر زمانی مورد استفاده قرار گیرند.

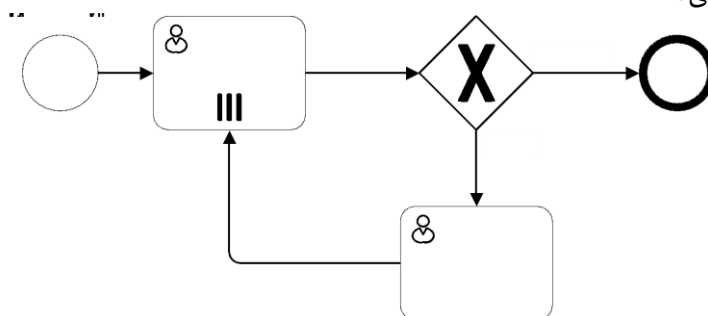
- **Multi-instance Standard:** معرف یک حلقه تکرار do-while است. تکرارها کاملاً متوالی هستند به عبارت دیگر دومین تکرار تا زمانی که اولین تکرار کامل نشده است شروع نمی‌شود. وقتی حلقه تکرار آغاز می‌شود، تعداد تکرار قابل پیش‌بینی نیست. فعالیت تکرار شوند تا زمانیکه شرط خروج از حلقه برآورده نشود، یا به حداکثر تعداد تکرار نرسد، تکرار می‌شود. عبارت منطقی شرط حلقه در تمام تکرارها، ثابت است، اما هر تکرار می‌تواند به واسطه اطلاعات فرآیندی مورد استفاده در شرایط خروج حلقه، نتیجه بررسی شرایط خروج را تغییر دهد. بدون انجام عملیات ارزیابی مورد نظر، نمی‌توان نتیجه شرط خروج از حلقه را از false به true تبدیل کرد. در نمودارهای مبتنی بر BPMN با نماد { } در وسط بخش پایین هر کدام از انواع فعالیت‌ها نشان داده می‌شود. تصویر زیر استفاده از این نماد را نشان می‌دهد.



- **Multi-instance Sequential:** معرف یک حلقه تکرار for-each است. همانند معرف قبلی، تکرارها کاملاً متوالی هستند یعنی دومین تکرار تا زمانی که اولین تکرار کامل نشده است، شروع نمی‌شود. وقتی حلقه تکرار آغاز می‌شود، تعداد تکرار قابل پیش‌بینی است. در این حالت تعداد تکرار می‌تواند، همان تعداد موارد موجود در یک لیست باشد. هر تکرار یک نمونه مستقل است، بنابراین مقادیر داده‌های یک تکرار معمولاً بر منطق نمونه‌های دیگر تأثیر نمی‌گذارند. حتی اگر بتوانید این مقادیر داده را از طریق یک ذخیره‌داده مشترک بین نمونه‌ها، ذخیره کنید باز هم نباید از آن به صورت اشتراکی بین نمونه‌ها استفاده کنید. در واقع این حلقه تکرار تا زمانیکه نمونه دیگری برای اجرا وجود نداشته باشد، تکرار خواهد شد. در نمودارهای مبتنی بر BPMN با نماد {=} در وسط بخش پایین هر کدام از انواع فعالیت‌ها نشان داده می‌شود. تصاویر زیر استفاده از این نماد را نشان می‌دهند.

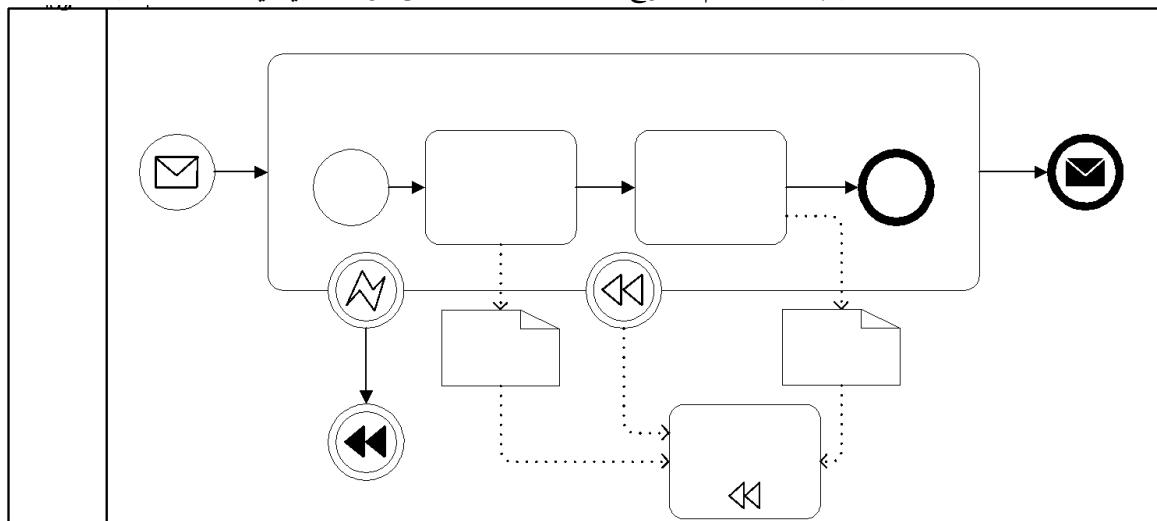


- **Multi-instance Parallel:** معرف یک حلقه تکرار Parallel-for-each است. تکرارها در این حالت چیزی شبیه به موازی هستند یعنی هر نمونه‌ای بدون اینکه ترتیب اجرا را رعایت کند و در هر زمانی می‌تواند آغاز شود و در نتیجه در هر زمان دیگری نیز می‌تواند خاتمه یابد. وقتی حلقه تکرار آغاز می‌شود، تعداد تکرار قابل پیش‌بینی نیست. در این حالت تعداد تکرار می‌تواند، تعداد مراجعه به فعالیت مورد نظر باشد که یا تابع الگوریتم فرآیند، حادث می‌شود و یا به صورت اختیاری و فارق از مسیر اصلی فرآیند، درخواست می‌شود. هر تکرار یک نمونه مستقل است، بنابراین مقادیر داده‌های یک تکرار معمولاً بر منطق نمونه‌های دیگر تأثیر نمی‌گذارند. در واقع این حلقه تکرار تا زمانیکه نتیجه یک تکرار در داده‌های آن باعث تحقق شرط حلقه تکرار (نتیجه بررسی شرط حلقه مقدار true شود) نشود، ادامه خواهد داشت. در نمودارهای مبتنی بر BPMN با نماد {|||} در وسط بخش پایین هر کدام از انواع فعالیت‌ها نشان داده می‌شود. تصویر زیر استفاده از این نماد را نشان می‌دهد.

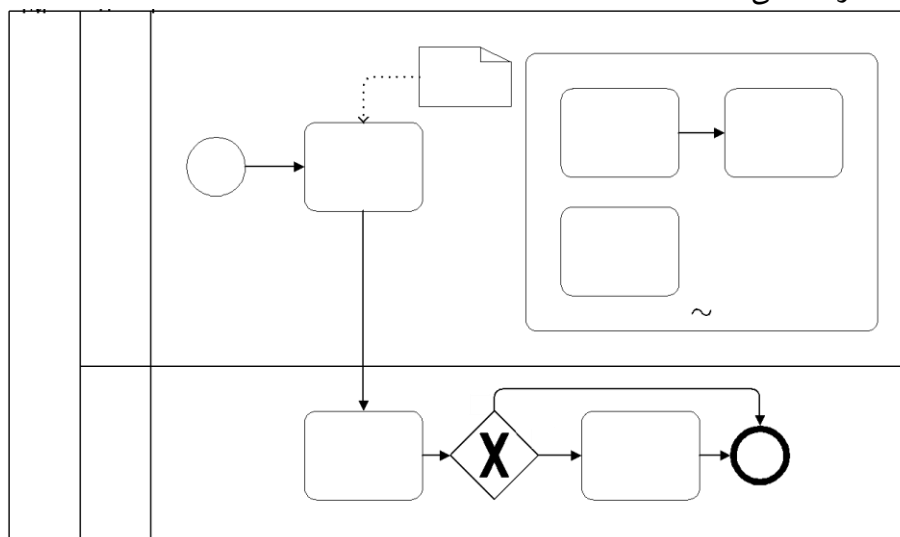




- **Compensation:** این دسته از فعالیت‌ها، فعالیت‌هایی هستند که حتما پس از اجرای یک فعالیت دیگر و به جهت جبران موارد مشخص در فعالیت قبلی و یا بازگشت به حالت قبل از آغاز فعالیت اول، استفاده می‌شوند. نکته مهم دیگر این است که آغاز اجرای این فعالیت‌ها حتما در پی تحقق یک رویداد آغازین و یا میانی دریافت‌کننده درخواست بازپایی موارد قبلی، که شرح آن در بخش رویدادها آمده است، می‌باشد. به عبارت دیگر هر گاه یک رویداد از نوع ذکر شده، حادث شود؛ نتیجه آن، اجرای یک فعالیت جبران است که به آن وابسته شده است. در نمودارهای مبتنی بر BPMN با نماد {<K} در وسط بخش پایین هر کدام از انواع فعالیت‌ها نشان داده می‌شود. تصویر زیر استفاده از این نماد را نشان می‌دهد.





- **Ad-hoc:** این دسته از فعالیت‌ها، که زیرفرآیند هستند به واسطه روند جریان فرآیند، آغاز نمی‌شوند و آغاز اجرای آنها فارق از جریان فرآیند می‌تواند باشد. به دلیل اینکه آغاز هر فعالیت در یک فرآیند، فعال‌کننده‌ای دارد (به طور مثال فعال‌کننده اجرای هر فعالیت ممکن است دریافت توکن از یک جریان‌توالی ورودی باشد؛ انتظار می‌رود که این زیرفرآیندها نیز فعال‌کننده‌ای داشته باشند. فعال‌کننده این زیرفرآیندها درخواست برنامه‌ریزی نشده یک یا چند کاربر خواهد بود. این زیرفرآیندها هیچ جریان خروجی و یا ورودی نخواهند داشت. نکته اینجا است که در طراحی یک فرآیند، می‌توان قابلیت آغاز این زیرفرآیندها را به یک نقش یا کاربر منسوب کرد و یا آنها را خارج از هر نقش و یا کاربری قرار داد تا قابلیت آغاز آن عمومی باشد. البته از خصوصیات این زیرفرآیندها این است که پس از آغاز اجرای آنها، می‌توان ادامه کار آنها را به شخص دیگری منسوب کرد. این زیرفرآیندها دارای وضعیت‌های مختلفی خواهند بود که در اغلب BPMS ها این وضعیت‌ها را برای هر نمونه از این زیرفرآیندها، نگهداری می‌کنند. این وضعیت‌ها شامل موارد Started (آغاز شده)، Comment (اظهار نظر شده)، Owner changed (مالک عوض شده)، Completed (خاتمه یافته) و Canceled (لغو شده) می‌باشند. در نمودارهای مبتنی بر BPMN با نماد {~} در وسط بخش پایین زیرفرآیندها نشان داده می‌شود. تصویر زیر استفاده از این نماد را نشان می‌دهد.






- **Call:** پیشتر در خصوص زیرفرآیندهای فراخوانی صحبت کردیم ولیکن این قسمت در خصوص وظایف فراخوانی صحبت خواهیم کرد. این دسته از وظایف، یک ارجاع به یک وظیفه سراسری یا فراخوانی یک وظیفه مرجع هستند. نکته مهم اینجا است که وظیفه فراخوانی شده؛ در محدوده تعریف فرآیند در حال طراحی نخواهد بود. از این دسته وظایف برای استفاده مجدد در فرآیندها، بهره ببرید. تفاوت اصلی این وظایف با وظایفی

مثل Service Task و یا Business Rule Task در این است که فراخوانی آنها به خارج از BPMS اشاره دارد؛ در حالی که فراخوانی این وظایف به محتوای طراحی شده درون BPMS اشاره دارد. با وجود اینکه، وظایف می‌توانند فراخوانی شوند ولیکن وظایفی ارزش فراخوانی و یا به عبارت دیگر استفاده مجدد دارند که در محیط BPMS مورد نظر ساختار آنها به صورت یک Asset (دارایی) ذخیره شود. به همین دلیل اگر BPMS مورد نظر دارای یک BRMS تعریف شده درونی باشد؛ می‌توان بجای استفاده از یک Business Rule Task، یک وظیفه از نوع فراخوانی

وظیفه بررسی قواعد کسب و کار را استفاده کرد که در آن صورت نماد آن به شکل  خواهد بود. یا اگر BPMS مورد نظر یک فرم‌ساز<sup>6</sup>

داشته باشد؛ در نتیجه می‌توان یک وظیفه از نوع فراخوانی وظیفه کاری را استفاده کرد که در آن صورت نماد آن به شکل  خواهد بود. یا اگر BPMS مورد نظر بتواند وظایف از نوع وظیفه دستوری، وظیفه دستی و یا وظیفه مفهومی را به صورت مستقل ذخیره نماید؛ در نتیجه می‌توان

یک وظیفه از نوع فراخوانی هر یک از آنها را استفاده کرد که در آن صورت نماد آنها به ترتیب به شکل ،  و یا  خواهند بود.

همانطور که دیده می‌شود، در نمودارهای مبتنی بر BPMN هیچ نماد متفاوتی برای این دسته از وظایف وجود ندارد فقط خط حاشیه آن به شکل توپر نمایش داده می‌شود.

نمادهای جریان، دنباله‌هایی هستند که دو نقطه آغاز و پایان دارند و به ترتیب به نقاط خروجی و ورودی سایر نمادهای معرفی شده در این استاندارد، متصل می‌شوند. این دسته از نمادها وظایف مختلفی به عهده دارند به عنوان مثال برخی از آنها وظیفه انتقال توکن اجرای فرآیند را به عهده دارند، برخی دیگر وظیفه سازگارسازی اطلاعات درون فرآیند با ساختارهای خارجی را به عهده دارند، برخی شرط‌های مختلفی را برای انتقال توکن‌ها کنترل می‌کنند، برخی پیام‌هایی را انتقال می‌دهند و برخی وظایف دیگر دارند. کلیه جریان‌ها در سه طبقه‌بندی اصلی زیر قرار می‌گیرند:

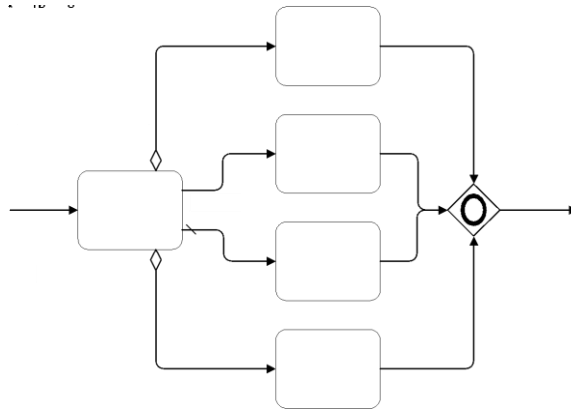
- جریان‌توالی
- جریان‌پیام
- ارتباطات

هر کدام از انواع جریان‌های موجود در طبقه‌بندی ذکر شده، نیز قابل تفکیک هستند. در ادامه به ترتیب جزئیات هر یک از این طبقه‌بندی‌ها را توضیح می‌دهیم.

## جریان‌های توالی

جریان‌های توالی، ترتیب انجام کارها در یک فرآیند کسب‌وکاری را مشخص می‌کنند. جریان‌های توالی، عناصر جریان را در فرآیند کسب‌وکار به هم متصل می‌کنند و مسیری را که یک توکن فرآیند در فرآیند کسب‌وکار باید دنبال کند، تعیین می‌کنند. یک جریان توالی همیشه در حوزه فرآیندی که به آن وابسته است باقی می‌ماند و نمی‌توان یک جریان توالی داشت که نقطه آغاز آن در یک حوزه فرآیندی باشد و پایان آن در حوزه فرآیندی دیگر باشد. از طرف دیگر اگر جریان توالی داخل یک زیرفرآیند باشد، نمی‌تواند از مرز این زیرفرآیند به بیرون هدایت شود. دقت داشته باشید که یک جریان توالی ورودی، جریان توالی است که به یک عنصر جریان منتهی می‌شود. جریان توالی خروجی، جریان توالی است که مسیر فرآیند خارج از عنصر جریان را تعیین می‌کند. اکثر عناصر جریان شامل جریان توالی ورودی و خروجی هستند. رویدادهای آغاز و پایان استثنای این قاعده محسوب می‌شوند. رویدادهای آغاز فقط می‌توانند جریان توالی خروجی داشته باشند. رویدادهای پایان فقط می‌توانند شامل جریان توالی ورودی باشند. فارق از این مسائل، جریان‌های توالی دارای انواع زیر می‌باشند:

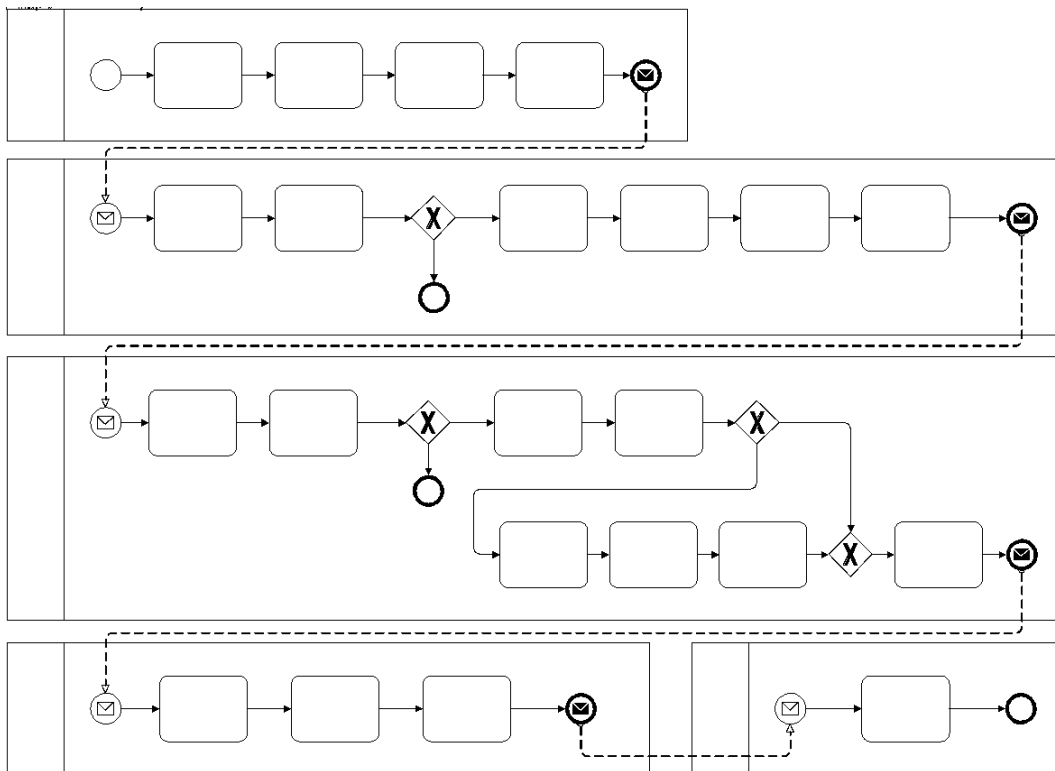
- **Unconditional Sequence Flow:** این جریان توالی که جریان توالی بدون شرط نام دارد نشان دهنده مسیر معمولی بین دو عنصر جریان است. این جریان توالی در نمودارهای مبتنی بر BPMN با نماد {→} نشان داده می‌شود. اکثر عناصر جریان فقط می‌توانند یک جریان توالی بدون شرط خروجی داشته باشند. فقط گذرگاه موازی {+} است که می‌تواند حاوی چندین جریان توالی از این نوع باشد. این جریان‌های توالی در این حالت مسیرهای موازی فرآیند کسب‌وکار را نشان می‌دهند. گذرگاه انتخاب انحصاری {X}، گذرگاه انحصاری شرطی {◇}، گذرگاه انتخاب اجتماعی {⊙}، گذرگاه انتخاب رویداد انحصاری {⊙}، گذرگاه انتخاب رویداد انحصاری آغازگر {⊙} و گذرگاه رویداد موازی {⊕}، نمی‌توانند جریان‌های توالی خروجی بدون شرط داشته باشند. این گذرگاه‌ها از جریان‌های توالی شرطی و جریان‌های توالی پیش‌فرض برای تعیین جریان فرآیند کسب‌وکار استفاده می‌کنند.
- **Conditional Sequence Flow:** این جریان توالی که جریان توالی شرطی نام دارد برای کنترل جریان یک فرآیند کسب‌وکار بر اساس شرایط خاص مورد استفاده قرار می‌گیرد. این جریان توالی در نمودارهای مبتنی بر BPMN با نماد {→} نشان داده می‌شود. با وجود اینکه بجز رویدادها، همه عناصر جریان می‌توانند از جریان توالی شرطی خروجی استفاده کنند ولیکن برخی از BPMS ها استفاده از این جریان‌های توالی خروجی را فقط برای گذرگاه‌ها مجاز می‌دانند. در این دسته از BPMS ها این نوع از جریان‌های توالی خروجی فقط در گذرگاه انتخاب انحصاری {X}، گذرگاه انتخاب اجتماعی {⊙}، گذرگاه انتخاب رویداد انحصاری {⊙}، گذرگاه انتخاب رویداد انحصاری آغازگر {⊙} و گذرگاه رویداد موازی {⊕} مورد استفاده قرار خواهند گرفت. برای تعریف و بیان شرط‌های هر یک از این جریان‌های توالی می‌توانید از عبارات منطقی و یا عبارات محاوره‌ای در داخل علامت {} و در نزدیکی این جریان‌ها، استفاده کنید.
- **Default Sequence Flow:** این جریان توالی که جریان توالی پیش‌فرض نام دارد برای نمایش حالتی در یک فرآیند کسب‌وکار استفاده می‌شود که تمایل داشته باشید، در هر شرایطی، توکن اجرای فرآیند از این جریان توالی عبور کند. این جریان توالی در نمودارهای مبتنی بر BPMN با نماد {→} نشان داده می‌شود. موارد کاربرد و یا محدودیت‌های کاربردی جریان توالی پیش‌فرض خروجی مشابه جریان توالی شرطی خروجی است. تصویر زیر استفاده از این سه نماد را نشان می‌دهد.



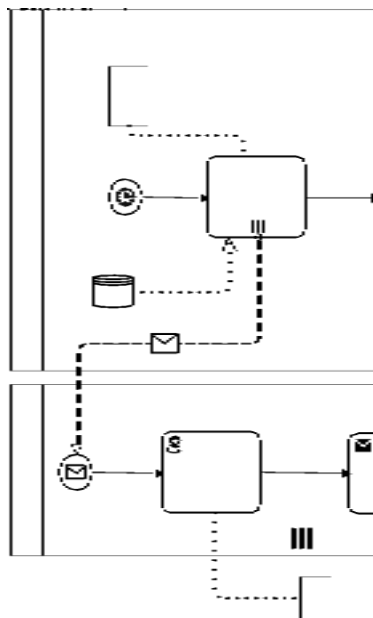
## جریان‌های پیام

اگر بخواهیم یک سیستم را به گونه‌ای سلسله‌مراتبی از عناصر طبقه‌بندی کنیم؛ شاید بتوان هر سیستم را به چندین برنامه و هر برنامه را به چندین فرآیند، طبقه‌بندی کرد. با این نوع طبقه‌بندی، برای داشتن یک سیستم کارآمد باید بتوانیم بین برنامه‌های آن ارتباط مناسب و در داخل برنامه‌ها بین فرآیندهای آنها ارتباط مورد نیاز را برقرار کنیم. در این خصوص، استاندارد BPMN نمادهای جریان‌پیام را معرفی کرده است که یک طراح بتواند این نوع ارتباطات را در نمودارهای همکاری فرآیندی، ترسیم نماید. نمودارهای همکاری فرآیندی نشان می‌دهد که چگونه دو یا چند فرآیند یا دو یا چند مشارکت‌کننده بدون کنترل مرکزی با یکدیگر به روشی هماهنگ شده، تعامل دارند. جریان‌پیام نحوه ارتباط و همکاری دو فرآیند یا دو مشارکت‌کننده با کنترل غیر متمرکز را نشان می‌دهد. یک حوزه‌فرآیند یا فعالیتی یا رویدادی در این حوزه‌فرآیندی می‌تواند پیامی را به حوزه‌فرآیند دیگر یا رویدادی یا مشارکت‌کننده در یک حوزه‌فرآیندی دیگر و یا حتی به یک کنشگر در جایگاه یک مشارکت‌کننده، ارسال کند ولیکن هرگز نمی‌تواند که دو فعالیت را در یک حوزه‌فرآیندی به هم مرتبط نماید. امروزه اکثر BPMS این امکان را پشتیبانی کرده‌اند ولیکن برخی از BPMS ها از روش‌های ابتکاری دیگری این نیاز را مرتفع کرده‌اند. فارق از این که تمایل به استفاده از کدام BPMS دارید، خود را مکلف می‌دانم که شرح کاملی برای این نمادها ارائه نمایم. جریان‌های پیام دارای انواع زیر می‌باشند:

- **Message Flow:** این جریان‌پیام که جریان‌پیام‌بدون‌تزیین نام دارد نشان دهنده مسیر تبادل پیام هماهنگی بین دو فرآیند مختلف و یا دو عنصر درونی دو فرآیند مختلف است. این جریان‌پیام در نمودارهای همکاری مبتنی بر BPMN با نماد { — — — } نشان داده می‌شود. عناصری مثل خود حوزه‌فرآیند { } و انواع نمادهای رویداد مبتنی بر پیام { } که یکی از این دو نماد { } را درون خود دارند، نمادهای وظیفه‌ارسال‌پیام { }، وظیفه‌دریافت‌پیام { } و برخی دیگر از وظایف به شرط آنکه درون یک فرآیند نباشند، می‌توانند در دو طرف یک جریان‌پیام قرار داشته باشند. تصویر زیر استفاده از این نماد را نشان می‌دهد.

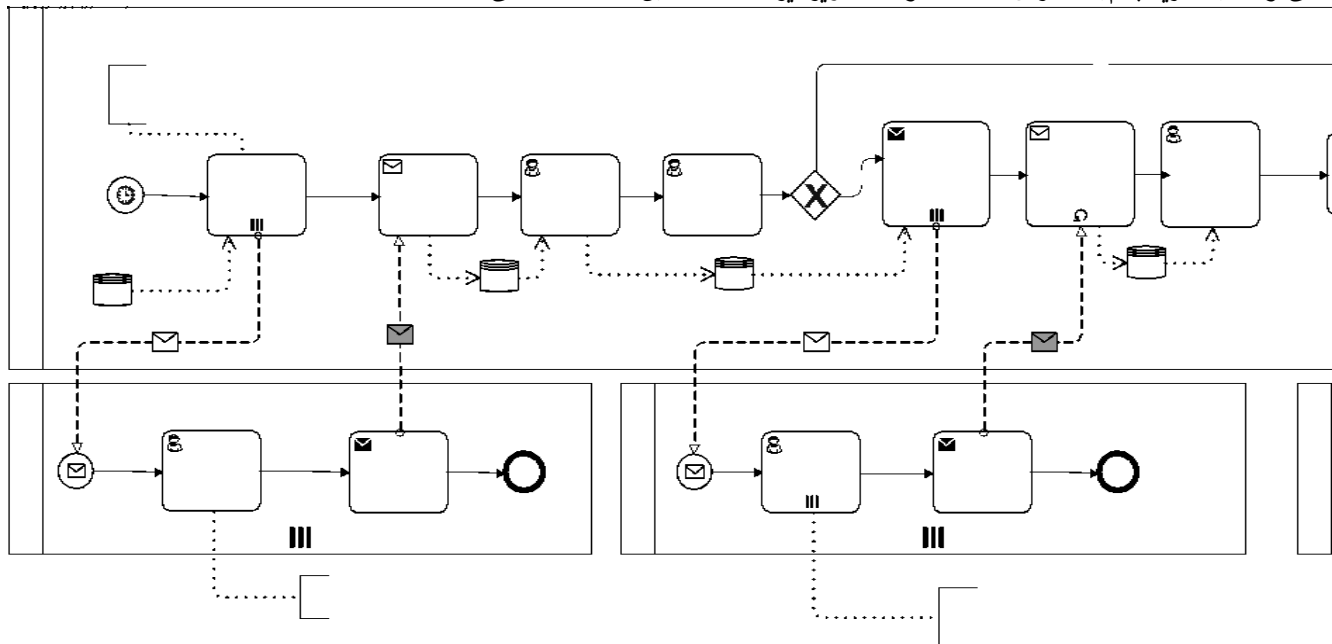


- **Initiating Message Flow with Decorator:** این جریان پیام که جریان پیام شروع کننده نام دارد همان جریان پیام بدون تزئین است که فقط با توجه به جهت جریان، مشخص می کند که نماد نقطه آغاز (اولیه) پیامی را برای اولین بار ایجاد کرده و برای نماد نقطه پایان (ثانویه) ارسال می کند. به عبارت دیگر، نقطه آغاز، پیام مورد نظر را ایجاد کرده و در مسیر این جریان پیام، به طرف نقطه پایان ارسال می نماید. استفاده از این نماد، می تواند به درک سریع تر نمودار طراحی شده کمک کند و در برخی از BPMS ها استفاده از این نماد به شما امکان می دهد که قاعده و یا ساختار پیام را نیز به صورت پارامتریک مشخص نمایید. این جریان پیام در نمودارهای همکاری مبتنی بر BPMN با نماد { }-> [Envelope Icon] نشان داده می شود. عناصری که می توانند به این جریان پیام متصل شوند مانند عناصری هستند که می توانند به جریان پیام بدون تزئین متصل شوند. تصویر زیر استفاده از این نماد را نشان می دهد.



- **Non-Initiating Message Flow with Decorator:** این جریان پیام که جریان پیام پاسخ دهنده نام دارد شبیه جریان پیام شروع کننده است فقط مشخص می کند که نماد نقطه آغاز (اولیه) پیامی را که قبلا دریافت کرده است، تغییر داده (نظر خود را به عنوان پاسخ به آن افزوده است) و برای نماد نقطه پایان (ثانویه) ارسال می کند. استفاده از این نماد، می تواند به درک سریع تر نمودار طراحی شده کمک کند و در برخی از BPMS ها

استفاده از این نماد به شما امکان می‌دهد که قاعده و یا ساختار پاسخ به پیام را نیز به صورت پارامتریک مشخص نمایید. این جریان‌پیام در نمودارهای همکاری مبتنی بر BPMN با نماد { }- [ ]- [ ] نشان داده می‌شود. عناصری که می‌توانند به این جریان‌پیام متصل شوند مانند عناصری هستند که می‌توانند به جریان‌پیام بدون تزیین متصل شوند. تصویر زیر استفاده از این نماد را نشان می‌دهد.

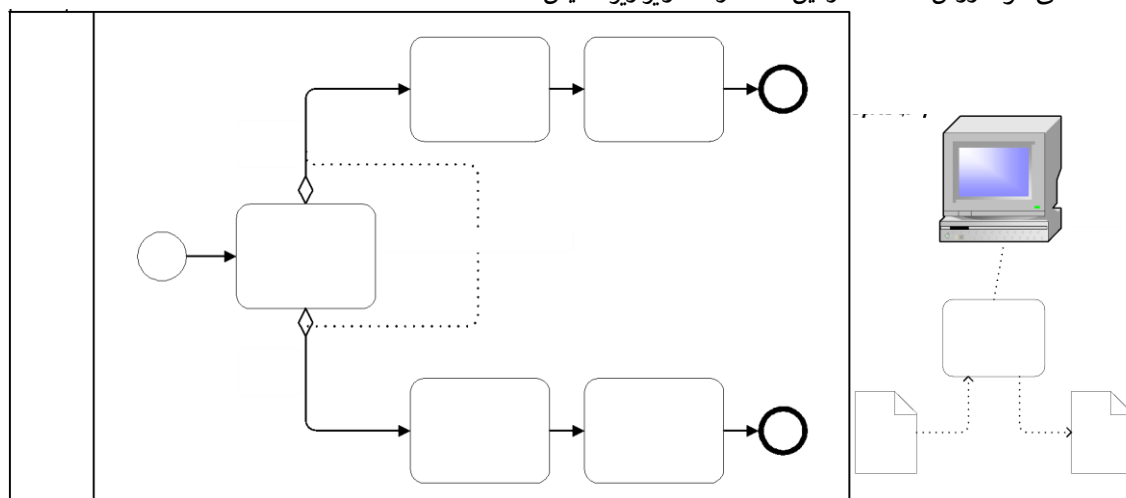


نماد پیام نشان دهنده آیتمی است که در طول فرآیند منتقل، دستکاری یا ذخیره می‌شود. اینها می‌توانند عناصر فیزیکی در جهان واقعیت و یا یک ماهیت اطلاعاتی باشند.

## ارتباطات

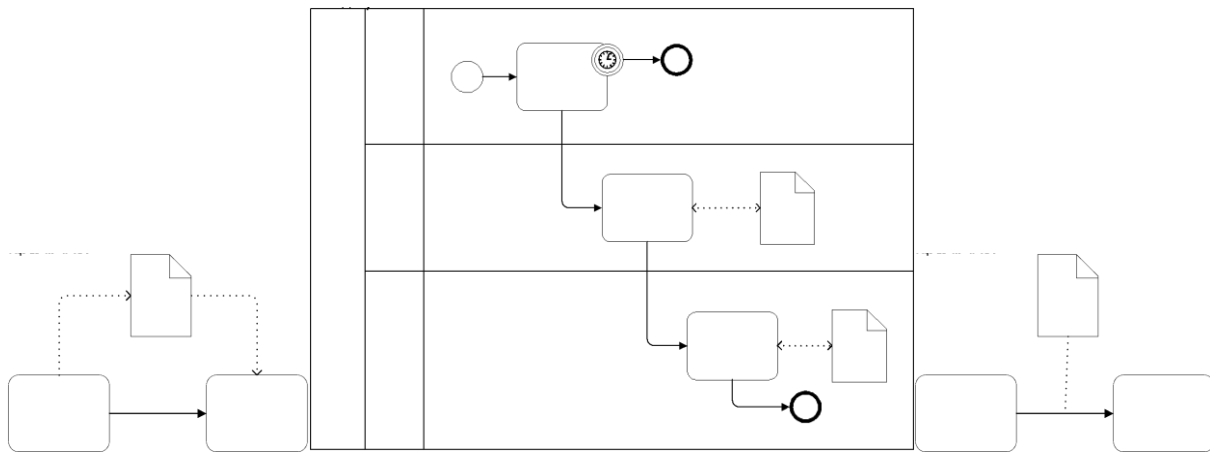
ارتباطات نوع خاصی از اتصال بین عناصر یا نمادهای BPMN هستند که در جریان فرآیند تأثیر مستقیم ندارند و به عبارت دیگر توکن‌ها در مسیر آنها و یا در جهتی که برخی مواقع در آنها مشخص شده است، حرکت نمی‌کنند. ارتباطات به سه نوع، طبقه‌بندی می‌شوند که در زیر توضیح داده شده است:

- **Association:** از یک نماد ارتباط برای مرتبط کردن اطلاعات با عناصر جریان یک فرآیند استفاده می‌شود. متن و اشیاء گرافیکی که تأثیری در جریان فرآیند ندارند را می‌توان با استفاده از این نماد به نمادهای جریان فرآیند مرتبط کرد. این ارتباط در نمودارهای مبتنی بر BPMN با نماد { } ..... نشان داده می‌شود. روش استفاده از این نماد در تصاویر زیر نمایش داده شده است.



- **Data Association:** این نوع ارتباط که ارتباط داده نیز نامیده می‌شود برای مدل کردن نحوه هدایت یا پیرون کشیدن داده‌ها به عناصر داده‌محور، استفاده می‌شود. همانطو که گفته شد، توکن‌های یک فرآیند در امتداد یک ارتباط داده، جریان نخواهند یافت؛ بنابراین هیچ تأثیر مستقیمی بر جریان یک فرآیند ندارند بلکه فقط ممکن است به عنوان عناصر کمکی برای نماد شیء داده استفاده شوند. این ارتباط داده در نمودارهای مبتنی بر

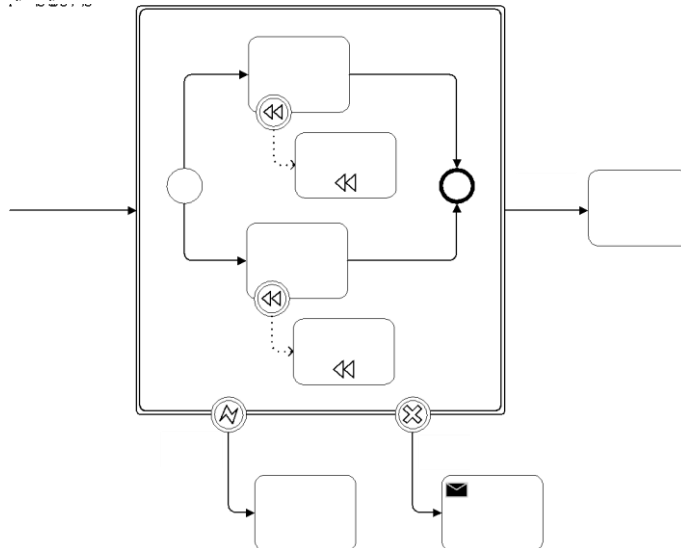
BPMN با نمادهای { } یا { } و یا بنا به شرایط خاص با نماد دوطرفه { } نشان داده می‌شود. همانطور که در تصاویر زیر مشاهده می‌کنید؛ در جهت جریان توالی، ارتباط داده‌ای نیز مشاهده می‌شود که نشان می‌دهد در فرآیند مورد نظر پس از اینکه توکن از فعالیت اول به دوم انتقال یافت، شیء داده که یک شیء داده‌محور مشترک بین هر دو فعالیت است، به فعالیت دوم انتقال می‌یابد. به عبارت دیگر زمانیکه توکن درون فعالیت اول است، اطلاعات از فعالیت اول به شیء داده انتقال یافته و زمانیکه توکن به فعالیت دوم وارد می‌شود، اطلاعات ذخیره شده در شیء داده به فعالیت دوم تحویل داده می‌شود. در کل همواره یک طرف این ارتباط یک شیء داده قرار خواهد داشت.



این نوع ارتباط داده، نماینده دستورات و فرامین سازگاری اطلاعات است. این دستورات و قواعد مشخص می‌کند که هر داده‌ای در چه ساختاری قرار خواهد گرفت و هر ساختار در محل مبدا با کدام ساختار در محل مقصد، سازگاری دارد. این عمل را نگاشت داده یا سازگاری داده نیز می‌گویند.

● **Compensation Association:** این نوع ارتباط که ارتباط جبران نیز نامیده می‌شود برای نمایش ارتباط بین یک رویداد پیوست شده با یک فعالیت جبران (وظیفه جبران یا زیرفرآیند جبران) استفاده می‌شود. نکته بسیار مهمی که باعث شناسایی و درک بهتر این نماد می‌شود این است که بدانیم هر فعالیت جبران، خارج از جریان عادی فرآیند اتفاق می‌افتد و مانند این است که بخواهید یک رویداد از یک فرآیند را به یک زیرفرآیند، وظیفه و یا فرآیند دیگر که در حوزه فرآیند اولیه قرار ندارد مرتبط کنید. بر اساس این فرض فعالیت‌های جبران مانند فعالیت‌های اتفاقی هستند با این تفاوت که یک رویداد سیستمی آنها را اجرا خواهد کرد. نقطه پایان این نوع از ارتباط همواره باید به یک فعالیت جبران باشد. این ارتباط در

نمودارهای مبتنی بر BPMN با نماد { } نشان داده می‌شود. روش استفاده از این نماد در تصویر زیر نمایش داده شده است.



خیلی‌ها مدل‌سازی با استاندارد BPMN را فقط مدل‌سازی وابسته به جریان‌های توالی و رفتارها در محدوده حوزه‌های فرآیند می‌دانند. شاید همین نگاه باشد که اغلب BPMS ها هم با همین الزامات و نیازها، به گونه‌ای طراحی و توسعه می‌یابند که کمتر نمونه‌ای از آنها را می‌توان یافت که در خصوص داده‌ها و یا چرخه داده‌ها در آنها راه‌کارهایی کاربردی وجود داشته باشد. ولیکن می‌دانیم که بیشترین کاربرد این استاندارد در خصوص موضوعات مبتنی بر داده است؛ پس نقش داده‌ها در این استاندارد کجاست؟

همانطور که در تحلیل و طراحی شیء‌گرا با متودولوژی UML، جریان داده‌ها و داده‌ها در نموداری به نام نمودار جریان اطلاعات مورد شناسایی قرار می‌گیرند و یا معرفی می‌شوند، استاندارد BPMN نیز دارای نمادهای مشخصی برای نمایش داده‌ها می‌باشد. با توجه به اینکه محل قرارگیری داده‌ها درون ماشین میزبان فرآیند است و یا خارج از آن، داده‌ها به سه نوع مختلف، طبقه‌بندی می‌شوند. انواع مورد نظر شامل موارد زیر هستند:

- اشیاء داده (محل ایجاد، نگهداری، تغییر و بازیابی، در ماشین میزبان فرآیند قرار دارد)
- مخزن داده (محل ایجاد، نگهداری، تغییر و بازیابی، در بیرون از ماشین میزبان فرآیند قرار دارد و بعضی‌ها آن را در ماشین میزبان پایگاه داده می‌دانند)
- اشیاء پیام (محل ایجاد، نگهداری، تغییر و بازیابی، معمولاً در بیرون از ماشین میزبان فرآیند قرار دارد و بعضی‌ها آن را در ماشین پیام‌ها می‌دانند در حالیکه خود ماشین میزبان فرآیند نیز می‌تواند آنها را مدیریت کند)

همین سه محل (ماشین میزبان فرآیند، ماشین میزبان پایگاه داده و ماشین پیام‌ها) تنها محل‌هایی نیستند که ممکن است مکانی برای نگهداری اطلاعات باشند! در اکثر راهکارهای کاربردی، فرم‌ها نیز ممکن است محلی برای نگهداری بخشی از اطلاعات باشند. اگر روش توسعه یک سیستم، توسعه فرآیند محور باشد، آن بخشی که داده‌های خود را در اشیاء داده نگهداری می‌کنند؛ فارق از وضعیت (Stateless) هستند و آن بخشی که داده‌های خود را در مخازن داده نگهداری می‌کنند؛ وابسته به وضعیت (Statefull) هستند ولیکن آن بخشی که داده‌های خود را درون فرم‌ها نگهداری می‌کنند؛ شبه وضعیت (Semi Statefull) هستند.

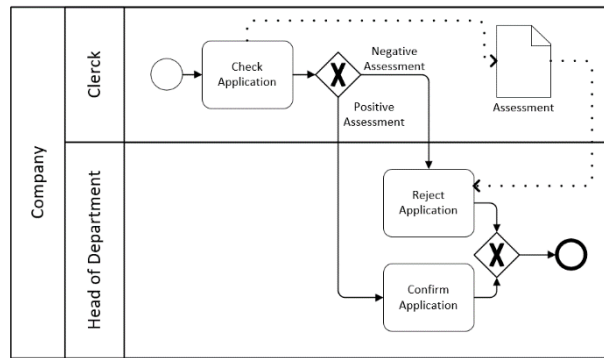
با معرفی وظیفه کاربری که یکی از انواع فعالیت‌ها است در خصوص فرم‌ها توضیح مختصری داده شده است. البته این استاندارد نوع خاصی از اشیاء داده را برای نمایش ساختارهای اطلاعاتی مورد نیاز این نوع از وظایف (به نوعی می‌توان آنها را نماینده اطلاعات شبه وضعیتی دانست)، استفاده می‌کند که در ادامه بیشتر در خصوص آنها توضیح داده می‌شود. همچنین اگر قرار باشد، پیام‌ها توسط خود ماشین میزبان فرآیند مدیریت شوند از وظیفه ارسال پیام و وظیفه دریافت پیام استفاده خواهد شد و اگر ماشین پیام‌ها، مدیریت را به عهده داشته باشد از شیء پیام استفاده می‌شود.

### اشیاء داده

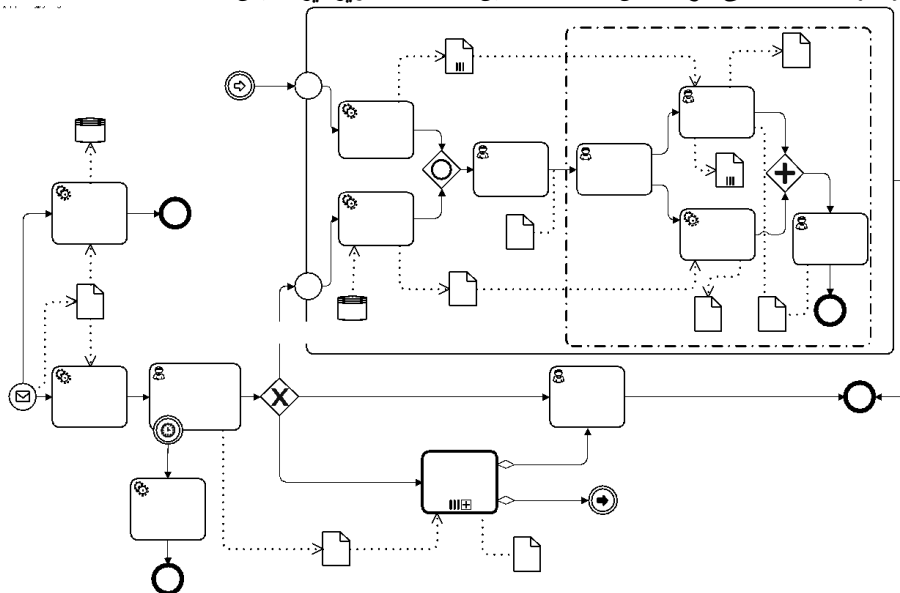
اشیاء داده به ما امکان می‌دهند که داده‌های جاری در روند یک فرآیند را نشان دهیم. با وجود اینکه این نمادها داخل یک فرآیند یا زیرفرآیند قرار خواهند داشت (هرگز خارج از حوزه یک فرآیند و یا زیرفرآیند، حضور ندارند) ولیکن هرگز در جریان‌های توالی و یا جریان‌های پیام قرار نخواهند گرفت زیرا در این جریان‌ها چیزی که در حال گذر است، توکن می‌باشد. این نمادها در ارتباطاتی به نام ارتباط داده که در این مستند توضیح داده شده است، قرار خواهند گرفت و به صورت مفهومی ممکن است جریان ثانویه‌ای را ایجاد کنند. استفاده از اشیاء داده راهی برای استفاده مکرر از ساختارهای اطلاعاتی در ترسیم نمودارهای فرآیند است. با وجود اینکه یک شیء داده یک ساختار اطلاعاتی را نمایندگی می‌کند ولی هرگز در خصوص ساختار داخلی خود، اطلاعی به ما نمی‌دهد. هر بار ترسیم یک شیء داده ممکن است نمایش دهنده حالت خاصی از ساختار داخلی آن باشد. هر کدام از این وضعیت‌ها، عنوانی خواهند داشت که می‌توان آن را در داخل علامت {} در ادامه نام شیء داده آورد. اشیاء داده با شش نماد مختلف که هر یک نماینده الگوی خاصی از رفتار با اطلاعات است، نمایش داده می‌شوند:

- **Data Object:** که به آن شیء داده ساده نیز گفته می‌شود ممکن است اسناد یا اطلاعات مورد استفاده در یک فرآیند را نمایش دهند. این اسناد می‌توانند فیزیکی و یا دیجیتالی باشند. این نوع از اشیاء داده عموماً تا زمانی که یک نمونه از فرآیند، فعال است، در دسترس خواهند بود. این شیء داده در نمودارهای مبتنی بر BPMN با نماد {} نشان داده می‌شود. روش استفاده از این نماد در تصویر زیر نمایش داده شده است.

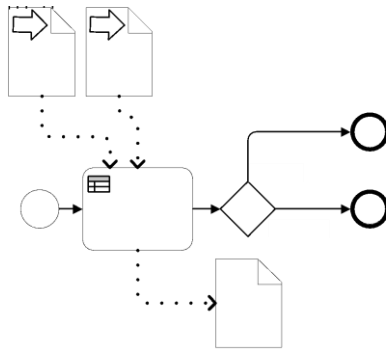




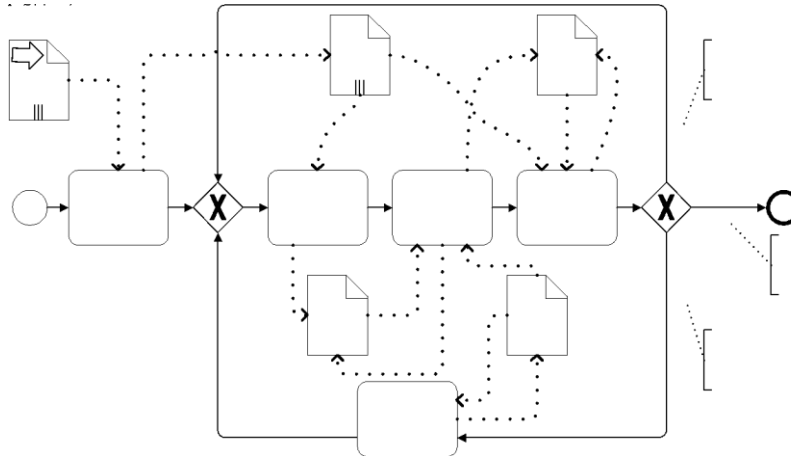
- Data Object Collection:** که به آن شیءمجموعه داده ساده نیز گفته می شود می تواند نمایی از یک مجموعه اطلاعات از اطلاعات درون یک شیء داده ساده باشد. اگر فرض کنید که شیء داده ساده یک ساختار اطلاعاتی برای یک فرآیند محسوب می شود یک شیءمجموعه داده ساده، لیستی از آن ساختار اطلاعاتی را نمایش می دهد. هرجایی از فرآیند که نمادها و عناصر درون جریان توالی با یک ساختار اطلاعاتی نمونه و منفرد کار دارند؛ از شیء داده ساده استفاده نمایید و هر جایی که با مجموعه ای از نمونه های یک ساختار اطلاعاتی، کار دارند از شیءمجموعه داده ساده استفاده کنید. همچنین می توانید با استفاده از علامت {} مقدار یا محدوده تعداد آیتم های فهرست مورد نظر را مشخص نمایید. این شیء داده در نمودارهای مبتنی بر BPMN با نماد { } نشان داده می شود. روش استفاده از این نماد در تصویر زیر نمایش داده شده است.



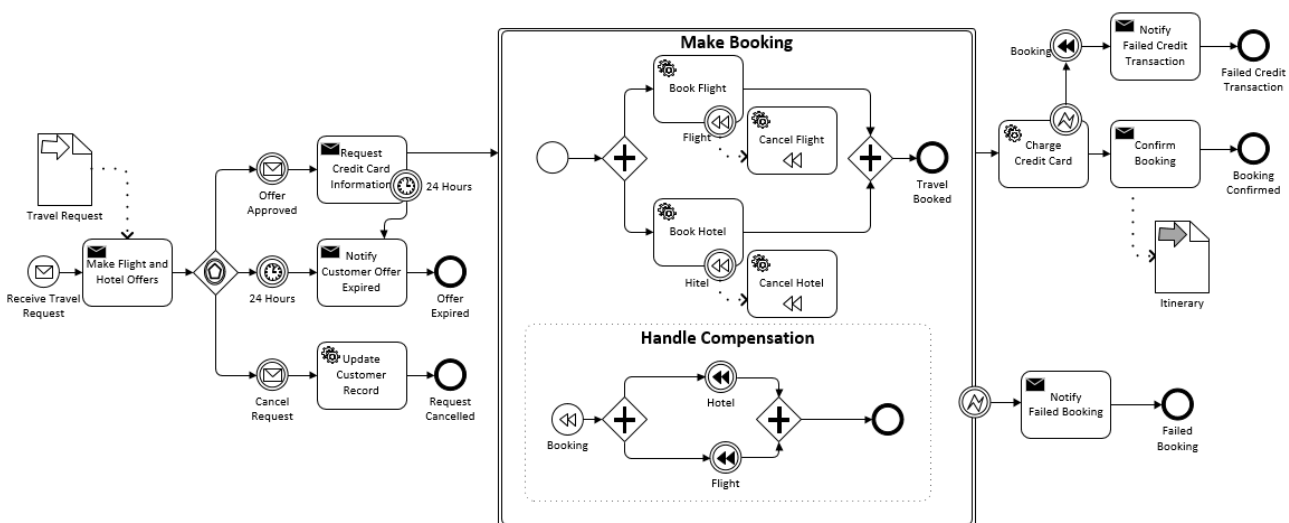
- Data Input:** که به آن داده ورودی نیز گفته می شود نماینده ساختار اطلاعاتی منفرد و غیر تکراری است که وظایف یا به طور کلی فعالیت ها (بجز زیرفرآیندهای وابسته) و رویدادها و فرآیندها نیاز دارند تا کار خود را انجام دهند. برخی مواقع به این ساختار اطلاعاتی، پارامتر هم گفته می شود. به عبارت دیگر اگر انجام یک فعالیت در روند اجرای فرآیند، تحقق یک رویداد و یا بررسی یک آزمون نیاز به اطلاعاتی خاص داشته باشد؛ در قالب داده ورودی نمایش داده می شود. به بیان ساده، اگر در جریان توالی یک نماد فعالیت، رویداد و یا گذرگاه وجود داشته باشد؛ فرض بر این است که این نمادها برای آغاز رفتار خود باید به داده ورودی مورد نیاز خود دسترسی داشته باشند و چنانچه توکن به این نمادها رسیده باشد ولیکن داده ورودی آماده نباشد، این نمادها، رفتار مورد نظر را آغاز نخواهند کرد. تصور داده ورودی برای اجرای رفتار یک فعالیت، رویداد و یا گذرگاه آسان است ولیکن تصور اینکه یک فرآیند نیز به داده ورودی نیاز داشته باشد کمی دشوار است. همانطور که قبلاً هم گفته شد؛ یک سیستم از تعدادی برنامه و هر برنامه از تعدادی فرآیند تشکیل شده است. می توان اینطور تجسم کرد که فرآیندها به داده های سراسری که در حوزه برنامه در دسترس هستند، نیاز دارند تا رفتار خود را آغاز کنند. این نوع از داده ها در نمودارهای مبتنی بر BPMN با نماد { } نشان داده می شود. روش استفاده از این نماد در تصویر زیر نمایش داده شده است.



- Data Input Collection:** که به آن مجموعه داده ورودی نیز گفته می‌شود نماینده فهرستی از یک ساختار اطلاعاتی است که مشابه داده ورودی، فعالیت‌ها برای انجام کار خود ممکن است به آن نیاز داشته باشند. برخی مواقع به این فهرست، مانند پارامتر هم برخورد می‌شود. به عبارت دیگر اگر انجام یک فعالیت در روند اجرای فرآیند، نیاز به فهرستی از اطلاعات خاص داشته باشد؛ در قالب مجموعه داده ورودی نمایش داده می‌شود. همچنین می‌توانید با استفاده از علامت  $\{\}$  مقدار یا محدوده تعداد آیتم‌های فهرست مورد نظر را مشخص نمایید. این نوع از داده‌ها در نمودارهای مبتنی بر BPMN با نماد  $\{\}$  نشان داده می‌شود. روش استفاده از این نماد در تصویر زیر نمایش داده شده است.

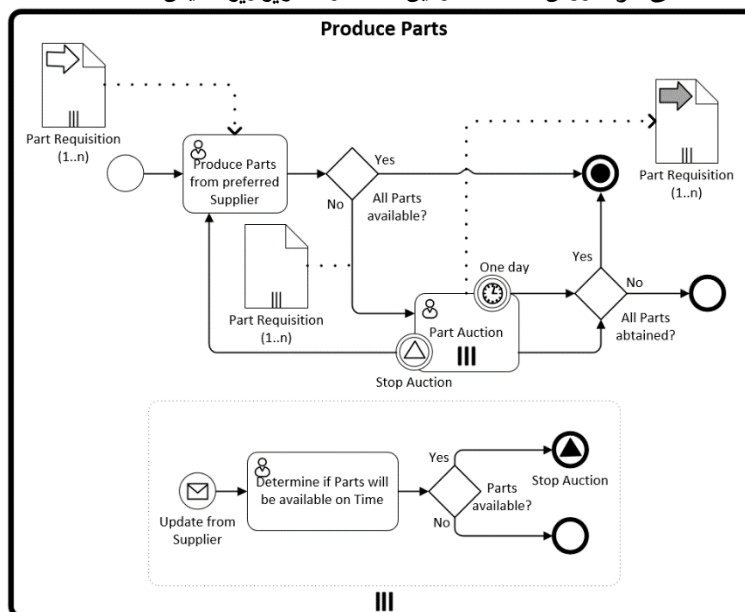


- Data Output:** که به آن داده خروجی نیز گفته می‌شود نماینده داده تولید شده یک فعالیت در جریان یک فرآیند است. اگر به خاطر داشته باشید در بخش مربوط به UML گفتیم که هر طبقه‌بندی‌کننده کپسوله که نماینده یک سرویس است، می‌تواند دو دسته داده خروجی داشته باشد. پارامتر خروجی و پارامتر بازگشتی هر دو نمونه‌هایی از داده خروجی هستند که به صورت مشابه از یک وظیفه سرویس در جریان یک فرآیند می‌توانند حاصل شوند. این نوع از داده‌ها در نمودارهای مبتنی بر BPMN با نماد  $\{\}$  نشان داده می‌شود. روش استفاده از این نماد در تصویر زیر نمایش داده شده است.



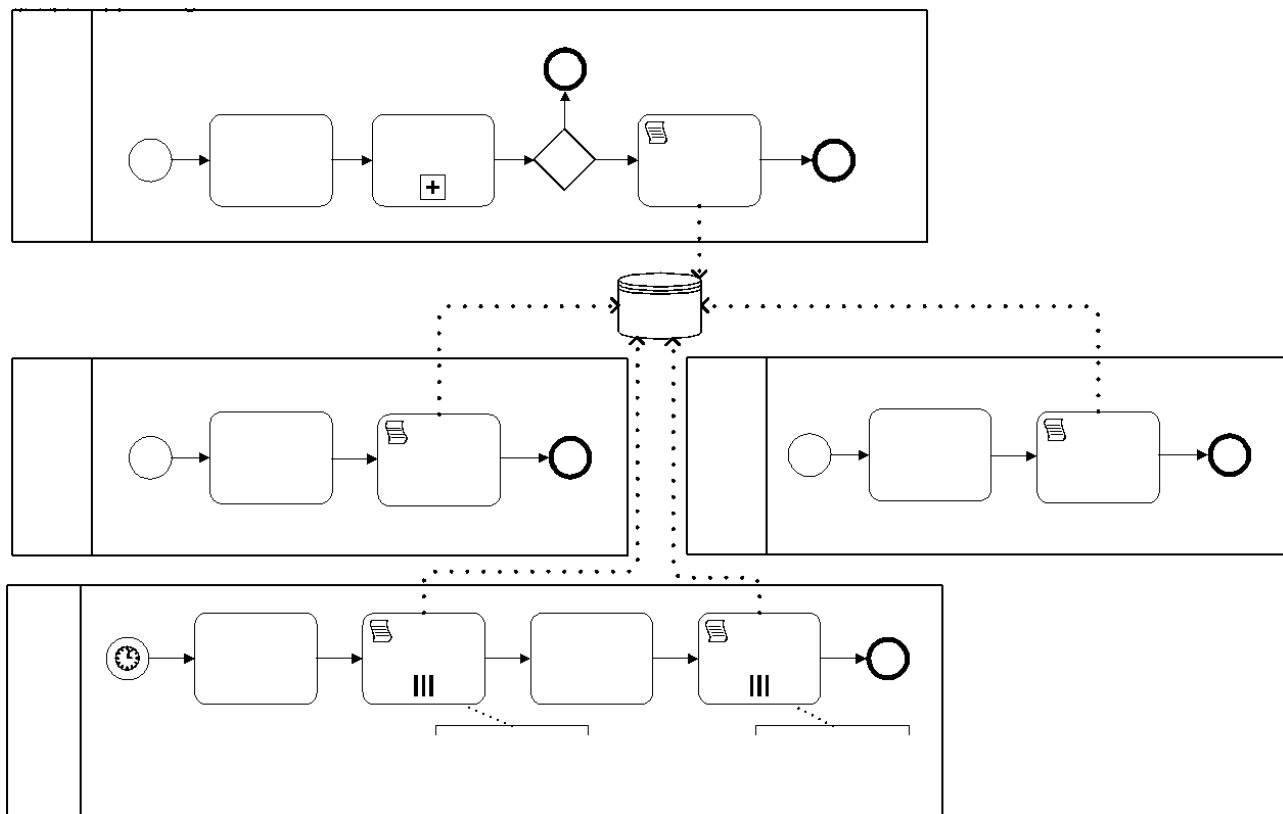
- Data Output Collection:** که به آن مجموعه داده خروجی نیز گفته می‌شود نماینده فهرستی از یک ساختار اطلاعاتی است که مشابه داده خروجی، ممکن است نتیجه فعالیت‌ها باشد. همانطور که گفته شد هر فعالیت ممکن است خروجی‌ها و یا نتایج مختلفی داشته باشد که از نظر انتزاعی با

هم متفاوت باشند؛ در چنین حالتی از چندین داده خروجی برای نمایش آن استفاده کنید زیرا همانطور که تفاوت انتزاعی را متصور می‌شوید باید دقت کنید که تفاوت ساختار داخلی نیز خواهند داشت. مجموعه داده خروجی فقط زمانی استفاده می‌شود که شما یک فهرست از اطلاعات داشته باشید که همه آنها از یک ساختار داخلی تبعیت می‌کنند. برخی مواقع به این فهرست، مانند پارامتر هم برخورد می‌شود. به عبارت دیگر اگر انجام یک فعالیت در روند اجرای فرآیند، ممکن است باعث ایجاد فهرستی از اطلاعات خاص شود که در قالب مجموعه داده خروجی نمایش داده می‌شود. همچنین می‌توانید با استفاده از علامت {} مقدار یا محدوده تعداد آیتم‌های فهرست مورد نظر را مشخص نمایید. این نوع از داده‌ها در نمودارهای مبتنی بر BPMN با نماد {} نشان داده می‌شود. روش استفاده از این نماد در تصویر زیر نمایش داده شده است.





## مخزن داده

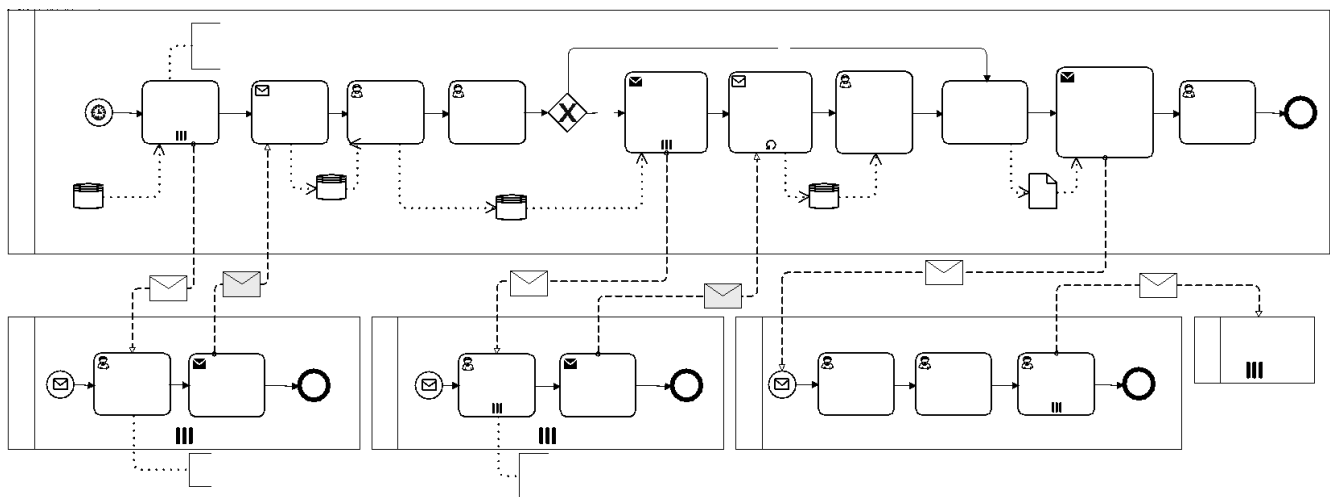
مخزن داده به ما امکان می‌دهند که نگهداری طولانی‌تر داده‌های جاری در روند یک فرآیند و بعد از خاتمه آن را نشان دهیم. با توجه به اینکه ذخیره‌سازی طولانی‌تر داده‌ها، نیازمند خروج از محیط حوزه فرآیند دارد، این نماد هم می‌تواند در داخل یک فرآیند یا زیرفرآیند قرار گیرد و هم بیرون آنها؛ ولیکن هرگز در جریان‌های توالی و یا جریان‌های پیام قرار نخواهد گرفت زیرا در این جریان‌ها چیزی که در حال گذر است، توکن نام دارد. این نماد در ارتباطاتی به نام ارتباط داده که در این مستند توضیح داده شده است، قرار خواهد گرفت و به صورت مفهومی ممکن است جریان ثانویه‌ای را ایجاد کند. استفاده از مخزن داده راهی برای استفاده مکرر از موجودیت‌های اطلاعاتی<sup>7</sup> در ترسیم نمودارهای فرآیند است. به عبارت دیگر، این نماد مشخص می‌کند که پس از آزادسازی هر نمونه از یک فرآیند، همچنان داده‌های مورد نظر شما در داخل محلی دیگر که مخزن اطلاعات نام دارد، قابل بازیابی است. با وجود اینکه یک مخزن داده یک موجودیت اطلاعاتی را نمایندگی می‌کند ولی هرگز در خصوص ساختار داخلی خود و یا جزئیات دستور یا جستار مورد نیاز، اطلاعی به ما نمی‌دهد. هر بار ترسیم یک مخزن داده ممکن است نمایش دهنده جستار خاصی از موجودیت اطلاعاتی مورد نظر باشد. هر کدام از این جستارها، عنوانی خواهند داشت که می‌توان آن را در داخل علامت {} در ادامه نام مخزن داده آورد. این موجودیت‌های اطلاعاتی یا به عبارت دیگر جستارها در نمودارهای مبتنی بر BPMN با نماد {} نشان داده می‌شود. روش استفاده از این نماد در تصویر زیر نمایش داده شده است.



## اشیاء پیام

اشیاء پیام به ما امکان می‌دهند که گفتمان جاری بین مشارکت‌کنندگان را نشان دهیم. این نمادها هرگز وابسته به هیچ فرآیند یا زیرفرآیندی نخواهند بود (هرگز داخل حوزه یک فرآیند و یا زیرفرآیند، حضور ندارند) و همواره در جریان‌های پیام قرار خواهند گرفت زیرا این جریان‌ها ارتباط بین مشارکت‌کنندگان را مشخص کرده و این اشیاء، محتوای گفتمان این ارتباط را معرفی می‌کنند. استفاده از اشیاء پیام راهی برای تنظیم و یا معرفی قالب محتوایی پیام‌های برنامه‌ریزی شده بین فرآیندها یا مشارکت‌کنندگان است. با وجود اینکه یک شیء پیام یک محتوای پیام را نمایندگی می‌کند ولی هرگز در خصوص قالب و محتوای از پیش تعیین شده آن، اطلاعی به ما نمی‌دهد. اشیاء پیام با دو نماد مختلف که هر یک نماینده عملیات خاصی در موضوع پیام‌ها است، نمایش داده می‌شوند:


- **Initiating Message:** که به آن شیء پیام‌آغازین نیز گفته می‌شود پیامی است که آغازگر یک سلسله مکاتبه بین مشارکت‌کنندگان است. این پیام می‌تواند فیزیکی و یا دیجیتال باشد. این نوع از اشیاء پیام عموماً تا زمانیکه سیستم مدیریت پیام‌ها در حال فعالیت است، در دسترس خواهند بود. این شیء پیام در نمودارهای مبتنی بر BPMN با نماد {} نشان داده می‌شود.
- **Non-Initiating Message:** که به آن شیء پیام‌درتعامل نیز گفته می‌شود پیامی است که قبلاً ایجاد شده است و مشارکت‌کنندگان به آن پاسخ داده و یا به شخص دیگری انتقال می‌دهند. این پیام می‌تواند فیزیکی و یا دیجیتال باشد. این نوع از اشیاء پیام عموماً تا زمانیکه سیستم مدیریت پیام‌ها در حال فعالیت است، در دسترس خواهند بود. این شیء پیام در نمودارهای مبتنی بر BPMN با نماد {} نشان داده می‌شود. روش استفاده از این نماد در تصویر زیر نمایش داده شده است.

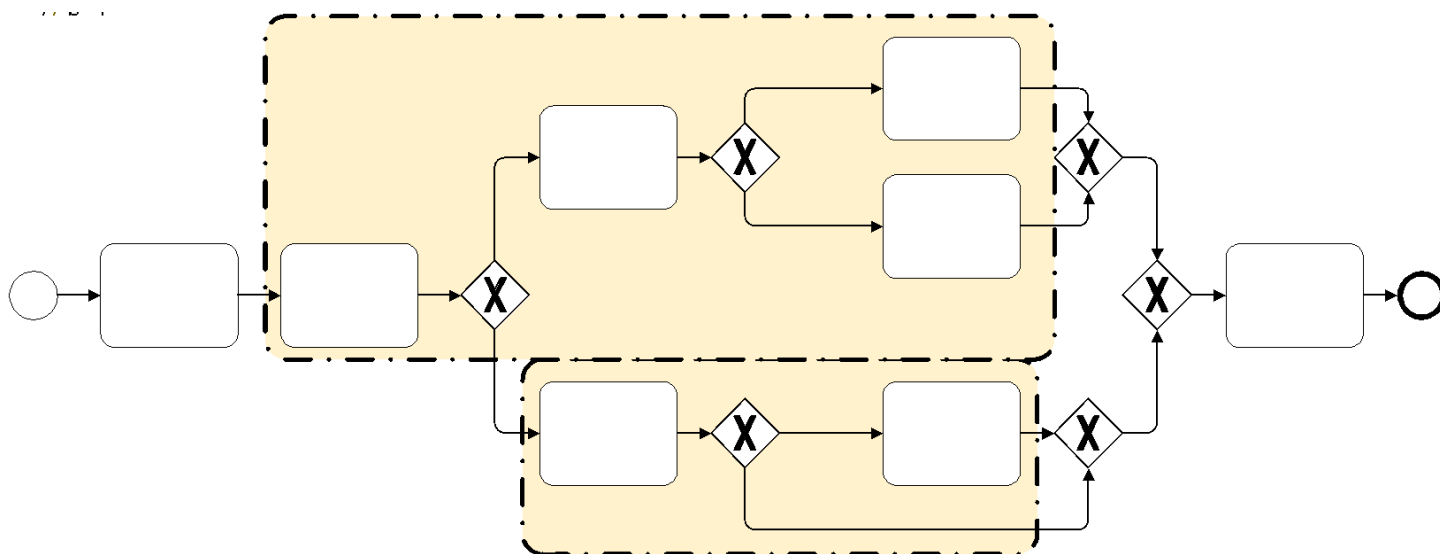


برخی از نمادهایی که در استاندارد BPMN نسخه 2 مشاهده می‌شوند، نمادهایی هستند که تأثیری در روند جریان و یا روال اجرای فرایند ندارند و فقط باعث درک بهتر نمودار مدل شده، می‌شوند. برخی از این نمادها را متولیان این استاندارد به صورت درونی در این استاندارد قرار داده‌اند ولیکن شما می‌توانید نمادهای خود یا تیم توسعه خود را نیز به این استاندارد اضافه کنید به شرط آنکه کلیشه یا همان Stereotype آن را قبلاً برای تیم توسعه خود تعریف کرده باشید. نمادهایی که به صورت درونی در این استاندارد وجود دارند شامل موارد زیر می‌باشند:


- گروه
- حاشیه‌نویسی متنی

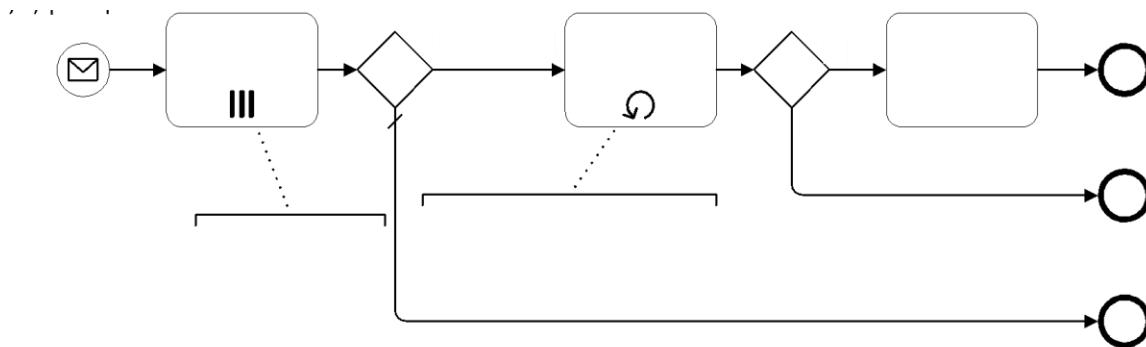
### گروه‌ها

گروه‌ها به عنوان یک ابزار سازماندهی‌کننده وظایف مشترک در یک مکان مورد استفاده قرار می‌گیرند. هر گروه به عنوان یک ظرف بصری، محیط غیررسمی و دلخواهی را برای عناصر مورد نظر در نمودار فرآیند کسب‌وکار ایجاد می‌کند. وظایف یک گروه می‌توانند بر اساس برخی ویژگی‌های مشترک به هم تعلق داشته باشند. گروه‌ها، کارکرد وظایف یا فرآیندها را تغییر نمی‌دهند، اما می‌توانند درک رفتار کلی فرآیندها را ساده‌تر کنند. اسامی گروه‌ها اصولاً از منظر خود کسب‌وکار، با ارزش نیست ولیکن این اسامی برای تیم‌های توسعه معانی خاصی داشته و با ارزش است. این عنصر طبقه‌بندی‌کننده در نمودارهای مبتنی بر BPMN با نماد  نشان داده می‌شود. روش استفاده از این نماد در تصویر زیر نمایش داده شده است.



### حاشیه‌نویسی‌های متنی

حاشیه‌نویسی متنی ابزاری برای ارائه توضیحات متنی و تشریحی در نمودارهایی است که از نمادها برای معرفی فرآیندها استفاده می‌کنند. با وجود اینکه نمودارها می‌توانند خیلی سریع‌تر یک کسب‌وکار را معرفی کنند ولیکن برخی مواقع کافی نیستند. در این حالت است که استفاده از حاشیه‌نویسی متنی می‌تواند یک ابزار راهگشا باشد و مطالب آنگونه که طراح مورد نظر دارد و یا لازم می‌داند به سایر افراد تیم توسعه، انتقال یابد. این نماد افزوده شده را می‌توان به اکثر نمادهای BPMN پیوند داد و چنانچه لازم باشد توضیحات تکمیلی و یا غیر تصویری را در آنها ذکر کرد. با وجود اینکه اکثر BPMS ها محیطی برای تنظیم نمادهای قرار گرفته در نمودارهای قابل طراحی خود هستند ولیکن همچنان از این نماد (حاشیه‌نویسی متنی) پشتیبانی می‌کنند. نکته بسیار مهم این است که این نماد هیچ تأثیری در روال فرآیند ندارد و فقط به جهت درک بهتر نمودار، در نمودارها قرار می‌گیرد. این عنصر تشریحی در نمودارهای مبتنی بر BPMN با نماد  نشان داده می‌شود. روش استفاده از این نماد در تصویر زیر نمایش داده شده است.



همواره در هر فرآیندی تعیین حدود و مرزها از اهمیت بسیار زیادی برخوردار است. این حدود و مرزها نه تنها مالکیت هر دارایی در سیستم یا فرآیند را مشخص می‌کند بلکه یکی از اهداف میانی برای کسب بهبود در فرآیندهای طراحی شده نیز می‌باشد. در هنگام طراحی فرآیند، مرزها همان حوزه‌فرآیند هستند. حوزه‌فرآیند، محلی است که در هنگام مدیریت فرآیند کسب‌وکار به عنوان یک صاحب کسب‌وکار، هماهنگی و الگوی اجرای رفتارهای مختلف در آن صورت می‌گیرد. به واسطه این تعریف، اگر فرآیندها در یک سیستم مبتنی بر فرآیند مثل یک BPMS، طراحی، توسعه، اجرا و بهینه می‌شوند؛ کاربران فرآیند که همان نقش‌های فرآیند هستند؛ مسیرهای یک حوزه‌فرآیندی را تشکیل می‌دهند. تصویر زیر نمایی از این تعریف است.



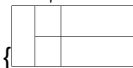
همانطور که در معرفی نمودارموارد کاربرد در همین مستند توضیح داده شده است، تعریف کاربران با کنشگران کمی تفاوت دارد. کنشگران مجموعه‌ای از کاربران یا سیستم‌هایی هستند که خارج از موضوع موردکاربرد وجود خواهند داشت. ولیکن کاربران قطعا مجموعه بزرگتری را نسبت به کنشگران تشکیل خواهند داد زیرا شامل کاربران سیستم نیز خواهند شد. پس در یک نگاه کاربران از کنشگران و کاربران سیستم تشکیل شده‌اند. در خصوص فرآیندها وقتی از کاربران صحبت به میان می‌آید منظور کاربران سیستم است و کنشگران در این گروه قرار نمی‌گیرند. شاید بپرسید که آیا BPMN، کنشگران را عقیم تصور کرده است؟ پاسخ کوتاه به این سوال، خیر است ولیکن در ادامه برای روشن شدن موضوع، ابتدا انواع حوزه‌ها و مسیرها را معرفی کرده و سپس توضیح خواهیم داد که استاندارد BPMN چگونه کنشگران را در نمودارهای خود نمایش می‌دهد.

فارق از اینکه در این استاندارد می‌توانید حوزه‌ها و مسیرهای فرآیند را به صورت افقی و یا عمودی ترسیم کنید؛ این نمادها انواع مختلفی خواهند داشت.

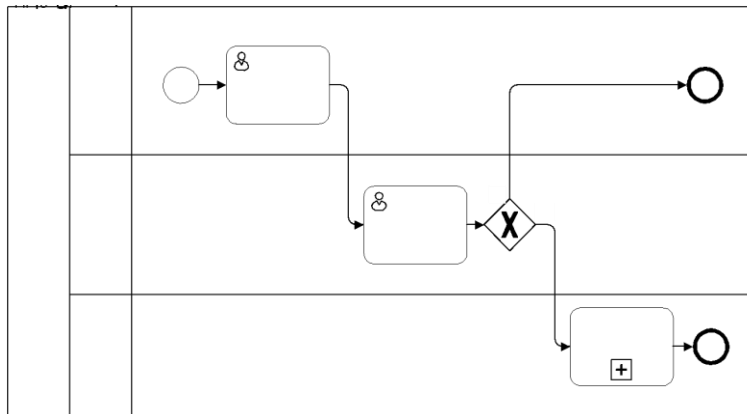
### حوزه‌های فرآیند

حوزه‌های فرآیندی سه دسته کلی دارند:

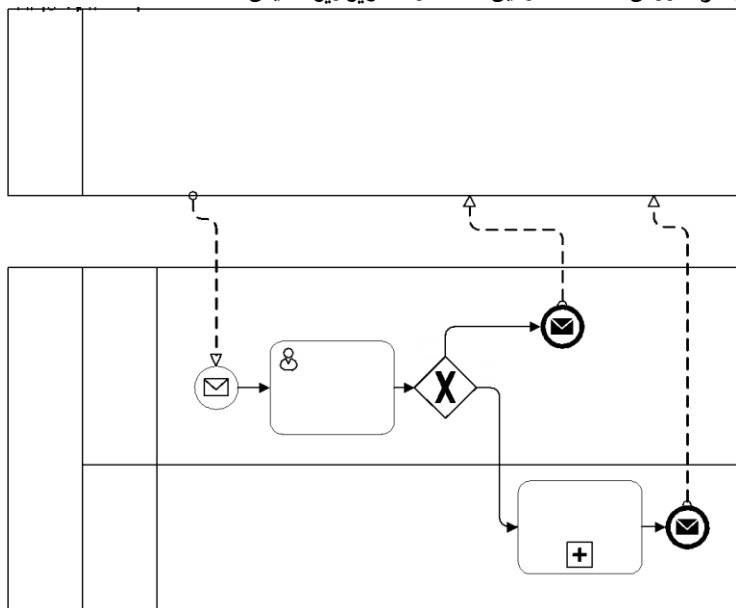
- **Expanded Pool:** که به آن حوزه‌فرآیند تشریحی نیز گفته می‌شود شامل یک کادر مستطیل شکل است که بر اساس قواعد استاندارد BPMN می‌تواند با ترسیم نمادهای مختلف این استاندارد و ایجاد نظم در اجرای رفتارهای آنها، فرآیند مورد نظر را به صورت کامل تشریح نماید. این عنصر مرزبندی در نمودارهای مبتنی بر BPMN با نماد { } نشان داده می‌شود. روش استفاده از این نماد در تصویر زیر نمایش داده شده است.





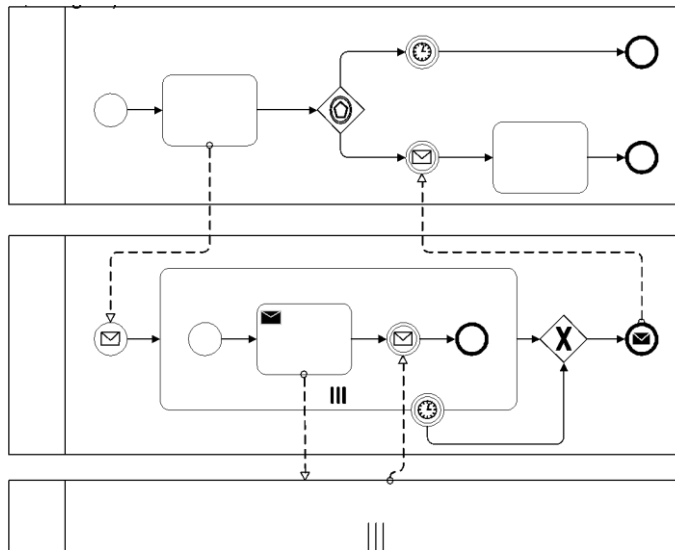


- **Collapsed Pool:** که به آن حوزه فرآیند مبهم نیز گفته می شود شامل یک کادر مستطیل شکل است که خالی از هر نماد یا عنصر BPMN می باشد و می تواند نماینده یک فرآیند، سیستم و یا کنشگر بشود. از این نماد زمانی استفاده می شود که اطلاع دقیقی در خصوص رفتار درون حوزه فرآیندی مورد نظر در اختیار نباشد و یا رفتارهای مورد نظر از اهمیت بسیار پایینی برخوردار باشند. این عنصر مرزبندی در نمودارهای مبتنی بر BPMN با نماد { } نشان داده می شود. روش استفاده از این نماد در تصویر زیر نمایش داده شده است.



تصاویر بالا را می توان به این روش تفسیر کرد:

- تصویر مربوط به حوزه فرآیند تشریحی نشان می دهد که Employee یا همان پرسنل دسترسی به این فرآیند را به صورت درونی دارند و می توانند برای ارائه درخواست خود از فرم های تعریف شده در فرآیند استفاده کنند. و پیرو آن هر نتیجه ای را در سیستم می توانند مشاهده نمایند در واقع این پرسنل از نظر فرآیند یک کاربر داخلی و یا یک نقش کاربری محسوب می شوند.
- تصویر مربوط به حوزه فرآیند مبهم نشان می دهد که Employee یا همان پرسنل دسترسی به این فرآیند را به صورت درونی ندارند و برای ارائه درخواست خود باید یک پیام خارجی به سیستم ارائه نمایند. و پیرو آن هر نتیجه ای را باید توسط پیام هایی که سیستم برای آنها ارسال می نماید، مشاهده کنند در واقع این پرسنل از نظر فرآیند کنشگر محسوب می شوند.
- **Collapsed Multi Instance Pool:** که به آن حوزه فرآیند مبهم تکرار پذیر نیز گفته می شود مانند حوزه فرآیند مبهم است با این تفاوت که یک نماد تکرار موازی را در پایین خود نمایش می دهد و می تواند نماینده چند فرآیند، سیستم و یا کنشگر بشود که به صورت موازی می توانند با فرآیند اصلی در تعامل باشند. از این نماد زمانی استفاده می شود که نه تنها اطلاع دقیقی در خصوص رفتار درون حوزه فرآیندی مورد نظر در اختیار نباشد بلکه تعداد دفعات تعامل و زمان وقوع آنها نیز قابل پیش بینی نباشد. این عنصر مرزبندی در نمودارهای مبتنی بر BPMN با نماد { } نشان داده می شود. روش استفاده از این نماد در تصویر زیر نمایش داده شده است.



در یک قاعده نانوشته بیان می‌شود که باید سعی شود در هر نمودار، یک حوزه‌فرآیند تشریحی وجود داشته باشد و در صورت نیاز از چندین حوزه‌فرآیند مبهم هم استفاده شود. همانطور که گفتیم، استاندارد BPMN روش خاصی برای ترسیم و یا نمایش کنشگران یک فرآیند دارد. استفاده از حوزه‌های فرآیند مبهم روشی برای نمایش کنشگرانی است که با فرآیند یا فرآیندهای در حال طراحی، تعامل دارند.

### مسیرهای فرآیند

با وجود اینکه نماد مسیر فرآیند فقط یک نوع دارد ولیکن جنس کسب‌وکاری آن یا به عبارتی شاخص طبقه‌بندی‌کننده آن می‌تواند متفاوت باشد. یکی از پرکاربردترین و معمول‌ترین آنها، طبقه‌بندی سازمانی است که برخی مواقع به آن طبقه‌بندی نقش‌های سازمانی یا نقش‌های کاربری نیز می‌گویند. شاخص‌های زیر ممکن است در مدل‌سازی، مورد نظر باشد:

- نقش کاربری
- سیستم‌ها
- ماشین‌ها (ایستگاه‌های سرویس دهنده و سرویس گیرنده)
- فاز یا مرحله
- دپارتمان یا گروه و...

نکته با اهمیت در خصوص این نماد این است که نام آن تا حدود زیادی باید معرف شرایط انتساب آن باشد. به عبارت دیگر هر مسیر فرآیند باید متناسب یا متنسین سیستمی، انسانی و یا مفهومی داشته باشد تا بتواند وظایفی که درون خود دارد، به آنها بسپارد. این عنصر مرزبندی در نمودارهای مبتنی بر BPMN با نماد { } نشان داده می‌شود. روش استفاده از این نماد را مکرراً در تصاویر قبلی مشاهده کرده‌اید.

وظایف و گذرگاه‌ها عناصر جریان هستند. وظایف همواره تحت شرایطی که گذرگاه‌ها مشخص می‌کنند، انجام می‌شوند. البته اگر کل الگوریتم جریان را به واسطه این دو عنصر طراحی کنیم، اینگونه به نظر می‌رسد که بدون هیچ وقفه و یا بدون هیچ بررسی واقعه‌ای، جریان مورد نظر سپری می‌شود و حتی نمی‌دانیم که در چه زمانی و به چه دلیلی آغاز می‌شود و به تبع آن، بر اساس چه واقعه و یا در چه زمانی خاتمه می‌یابد. از این رو است که اهمیت رویدادها برای مدل‌های فرآیند BPMN کمتر از وظایف یا گذرگاه‌ها نیست. ما باید با چند اصل اساسی برای استفاده از رویدادها، شروع کنیم.

رویدادها بر اساس محل قرارگیری شامل سه طبقه‌بندی هستند:

- رویداد آغازین که با یک دایره تک خط مشخص می‌شود
- رویداد میانی که با یک دایره دوخط توخالی مشخص می‌شود
- رویداد پایانی که با یک دایره دوخط توپر مشخص می‌شود

البته رویدادها فارق از طبقه‌بندی ذکر شده، می‌توانند بر اساس فعال‌کننده‌ها یا راه‌اندازهایشان نیز طبقه‌بندی شوند. قبل از اینکه این طبقه‌بندی را توضیح دهیم باید در خصوص یک طبقه‌بندی دیگر که می‌تواند طبقه‌بندی مورد نظر را تحت تاثیر قرار دهد توضیح دهیم. فعال‌کننده‌ها از هر نوعی باشند می‌توانند در یک رویداد به گونه‌ای تفسیر شوند که آن رویداد منتظر اجرای فعال‌کننده باشد و یا رویداد مورد نظر فعال‌کننده را اجرا کند. این دو نوع طبقه‌بندی شامل موارد زیر است:

- رویداد ارسال‌کننده که نمادهای (فعال‌کننده‌های) درونی آنها به رنگ تیره و یا سیاه نمایش داده می‌شوند
- رویداد دریافت‌کننده که نمادهای (فعال‌کننده‌های) درونی آنها به رنگ روشن و یا سفید نمایش داده می‌شوند

و اما طبقه‌بندی اصلی که مربوط به فعال‌کننده‌ها و یا راه‌اندازها می‌شود شامل موارد زیر است:

- رویداد بدون فعال‌کننده (خالی)
- رویداد مبتنی بر پیام به عنوان فعال‌کننده که یکی از این دو نماد {✉} را شامل می‌شود
- رویداد مبتنی بر زمان به عنوان فعال‌کننده که شامل نماد {🕒} می‌شود
- رویداد مبتنی بر قواعد به عنوان فعال‌کننده که شامل نماد {📋} می‌شود
- رویداد مبتنی بر سیگنال به عنوان فعال‌کننده که یکی از این دو نماد {⚠} را شامل می‌شود
- رویداد مبتنی بر وقوع خطا به عنوان فعال‌کننده که یکی از این دو نماد {🚨} را شامل می‌شود
- رویداد مبتنی بر سطح اجرا به عنوان فعال‌کننده که یکی از این دو نماد {⬆} را شامل می‌شود
- رویداد مبتنی بر چند فعال‌کننده به صورت انتخابی (فقط یکی از چند فعال‌کننده در دسترس خواهد بود) که یکی از این دو نماد {🔍} را شامل می‌شود
- رویداد مبتنی بر چند فعال‌کننده به صورت موازی (بیش از یک از چند فعال‌کننده در دسترس خواهد بود) که شامل نماد {+} می‌شود
- رویداد مبتنی بر پیوند نمودار (شبه رویداد انتقال نمادین جریان) که یکی از این دو نماد {➡} را شامل می‌شود
- رویداد مبتنی بر مفهوم جبران یا بازپایی موارد قبل به عنوان فعال‌کننده که یکی از این دو نماد {⏮} را شامل می‌شود
- رویداد مبتنی بر مفهوم صرف‌نظر کردن به عنوان فعال‌کننده که یکی از این دو نماد {⏏} را شامل می‌شود
- رویداد مبتنی بر خاتمه کامل به عنوان فعال‌کننده که شامل نماد {●} می‌شود

نکته: رویداد مبتنی بر پیام، زمانی استفاده می‌شود که ارسال یا دریافت پیام در یک لحظه، آغاز و خاتمه می‌یابد و نیازمند طول زمانی نیست.

همانطور که تا اینجا توضیح داده شد؛ رویدادها عناصر یا نمادهایی هستند که مانند وظایف و گذرگاه‌ها در مسیر یک جریان فرآیند قرار می‌گیرند. این برداشت خاصی از تعریف رویداد بود. البته در نسخه 2 این استاندارد، نوع دیگری از رویداد نیز مطرح شد که از نظر معرفی، شبیه به رویدادهای تعریف شده در نسخه قبلی بود ولیکن این رویدادهای اخیر رویدادهایی بودند که می‌توانستند به صورت عملی به وظایف (و از نظر مفهومی به گذرگاه‌ها) پیوست شوند.

یادآوری: نمادهایی مثل وظایف و یا گذرگاه‌ها، می‌توانند دارای 0 تا n جریان ورودی و یا خروجی باشند. فعال‌شدن هر یک از این جریان‌ها می‌تواند یا به صورت بلادرنگ (بدون فعال‌کننده یا شرط و یا رخدادها) باشد یا می‌تواند به واسطه شروط اولیه باشد که تفسیر پیچیده‌ای ندارند و یا بر اساس بروز یک واقعه خاص باشد. زمانیکه یک جریان، فعال می‌شود، یک توکن در آن ایجاد شده و عملیات خود را که کشف نقطه پایان جریان و حرکت به سمت آن است؛ انجام می‌دهد.

در واقع در نسخه 2 استاندارد BPMN بود که دو طبقه‌بندی زیر نیز به رویدادها افزوده شد:

- رویدادجریان که جریان‌های ورودی و خروجی به آن متصل می‌شوند و حداقل یک جریان‌ورودی دارند.
- رویدادپیوست‌شده که فاقد جریان‌ورودی هستند و به مرز یک فعالیت پیوست می‌شوند و فقط یک جریان خروجی دارند.

از زمانیکه رویدادهای پیوست‌شده در این استاندارد مطرح شد؛ یک قاعده دیگر نیز در تعریف رویدادها اضافه شد که البته کلیه طبقه‌بندی‌های موجود را تحت تأثیر قرار داد. این قاعده مربوط به قابلیت توقف توکن در مسیر جریان است. به عبارت دیگر رویدادها می‌توانند متوقف‌کننده و یا بدون توقف باشند. برای درک بیشتر این موضوع به مطلب زیر توجه کنید.

فرض کنید که در مسیر یک جریان، یک رویداد دریافت‌کننده پیام وجود دارد (این رویداد، رویدادی است که فعال‌کننده آن یک پیام خاص خواهد بود که نماد این رویداد منتظر دریافت آن می‌باشد). این رویداد می‌تواند توکن دریافت شده را متوقف کند تا پیام مورد نظر را دریافت کند و سپس توکن دریافت شده از جریان ورودی خود را به جریان خروجی خود انتقال دهد. از طرف دیگر، رویداد مورد نظر می‌تواند به محض دریافت توکن از جریان ورودی خود، بدون آنکه آن را متوقف کند، توکن را به جریان خروجی انتقال دهد!!!

شاید این کار را بیهوده بدانید ولیکن اگر بدانید که این رویداد می‌تواند دو جریان خروجی داشته باشد متوجه خواهید شد که رویداد مورد نظر اخیر، پس از انتقال توکن دریافت شده به خروجی اول خود، می‌تواند منتظر دریافت پیام باشد و پس از دریافت پیام مورد نظر، فعال شده و توکن جدیدی مانند توکن دریافت شده خود را کپی کند و آن را به جریان خروجی دیگر هدایت نماید.

این موضوع در خصوص رویدادهای پیوست‌شده نیز قابل تفسیر است. زمانیکه رویداد پیوست‌شده به یک وظیفه، یک رویداد بدون توقف باشد؛ وظیفه مورد نظر، پس از دریافت توکن از جریان ورودی خود، عملیاتی که باید انجام دهد را فعال کرده و پس از خاتمه آن عملیات، توکن مذکور را به جریان خروجی خود انتقال می‌دهد. نکته

اینجا است که در طی اجرای عملیات مورد نظر، اگر فعال‌کننده رویداد پیوست‌شده، عمل کند؛ یک توکن جدید ایجاد شده و به جریان خروجی این رویداد ارسال می‌شود. در این حالت 2 توکن حاصل کار این وظیفه خواهد شد. البته اگر فعال‌کننده رویداد پس از خاتمه عملیات مربوط به وظیفه مورد نظر و انتقال توکن به جریان خروجی وظیفه مورد نظر، توسط رویداد مورد نظر دریافت شود، توکن جدیدی کپی نشده و همینجا کار رویداد پیوست‌شده، خاتمه می‌یابد.

در حالت دیگر زمانیکه رویداد پیوست‌شده به یک وظیفه، یک رویداد متوقف‌کننده باشد؛ وظیفه مورد نظر، پس از دریافت توکن از جریان ورودی خود، عملیاتی که باید انجام دهد را فعال کرده و پس از خاتمه آن عملیات، توکن مذکور را به جریان خروجی خود انتقال می‌دهد. نکته اینجا است که در طی اجرای عملیات مورد نظر،

اگر فعال‌کننده رویداد پیوست‌شده، عمل کند؛ توکن موجود در جریان اصلی این وظیفه، متوقف می‌شود و یک توکن جدید ایجاد شده و به جریان خروجی این رویداد ارسال می‌شود. در این حالت فقط 1 توکن حاصل کار این وظیفه خواهد شد. البته اگر فعال‌کننده رویداد پس از خاتمه عملیات مربوط به وظیفه مورد نظر و انتقال توکن به جریان خروجی وظیفه مورد نظر، توسط رویداد مورد نظر دریافت شود، توکن جدیدی کپی نشده و همینجا کار رویداد پیوست‌شده، خاتمه می‌یابد.

همانطور که در دو مورد اخیر مشاهده می‌شود، اگر فعال‌کننده رویداد پس از خاتمه عملیات مربوط به وظیفه مورد نظر و انتقال توکن به جریان خروجی وظیفه مورد نظر، توسط رویداد مذکور، دریافت شود، توکن جدیدی کپی نمی‌شود. اگر سیستم در حال بررسی، به گونه‌ای باشد که در این حالت نیز باید توکن جدیدی تولید شود، می‌بایست از رویداد پیوست‌شده استفاده نشود و بجای آن از یک رویداد میانی که بر روی یکی از جریان‌های خروجی وظیفه مورد نظر قرار دارد، استفاده شود.

با تعریف‌ها و توضیحاتی که داده شد می‌توان طبقه‌بندی زیر را به رویدادها اضافه کرد:

- رویداد متوقف‌کننده که خطوط دایره را پیوسته نشان می‌دهد
- رویداد بدون توقف که خطوط دایره را منقطع نشان می‌دهد

به تصاویر زیر توجه کنید:



رویداد اول، یک رویداد میانی بدون توقف و دریافت کننده است که فعال کننده آن دریافت یک پیام خاص است؛ رویداد دوم، یک رویداد میانی متوقف کننده و دریافت کننده است که فعال کننده آن دریافت یک پیام خاص است و رویداد سوم، یک رویداد میانی بدون توقف و ارسال کننده است که فعال کننده آن ارسال یک پیام خاص است.

حال به تصاویر زیر توجه کنید:



رویداد اول، یک رویداد آغازین متوقف کننده و دریافت کننده است که فعال کننده آن دریافت یک پیام خاص است؛ رویداد دوم، یک رویداد آغازین بدون توقف و دریافت کننده است که فعال کننده آن دریافت یک پیام خاص است و رویداد سوم، یک رویداد آغازین متوقف کننده و دریافت کننده است که فعال کننده آن وقوع یک خطا است.

و در آخر به این تصاویر نیز توجه کنید:



رویداد اول، یک رویداد پایانی متوقف کننده و ارسال کننده است که فعال کننده آن ارسال یک پیام خاص است؛ رویداد دوم، یک رویداد پایانی متوقف کننده و دریافت کننده است که فعال کننده آن دریافت یک درخواست خاتمه سیستمی و یا دستی است و رویداد سوم، یک رویداد پایانی خاتمه کامل و ارسال کننده است که فعال کننده آن وقوع دستور جمع آوری فرآیند و آزادسازی سیستم است. جدول زیر ترکیب‌های محتمل رویدادها را نشان می‌دهد. شاید بهتر باشد به عنوان تمرین، هر یک از آنها را به گونه‌ای که توضیح داده شد، تفسیر کنید.

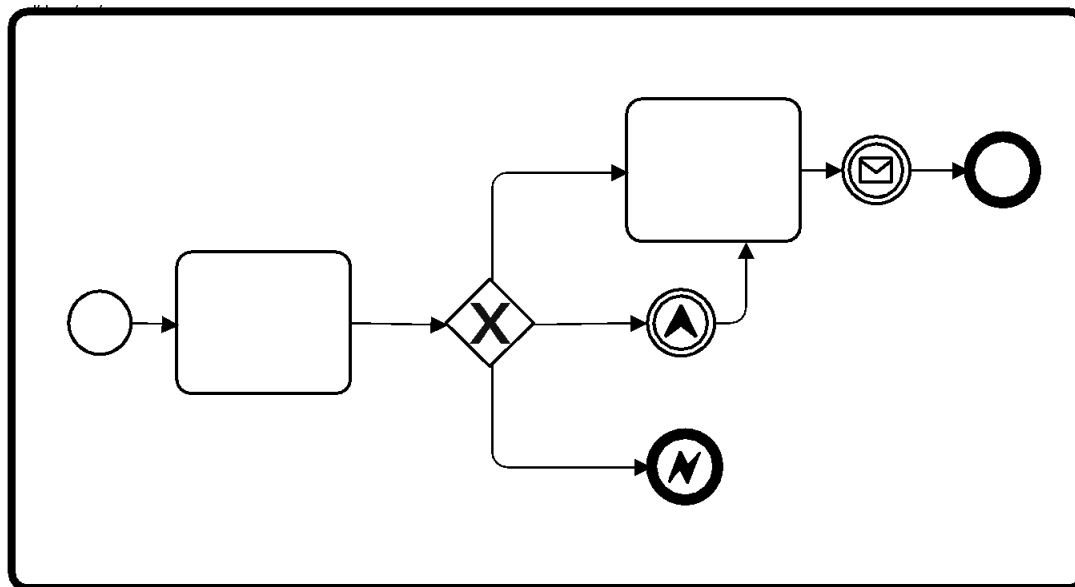
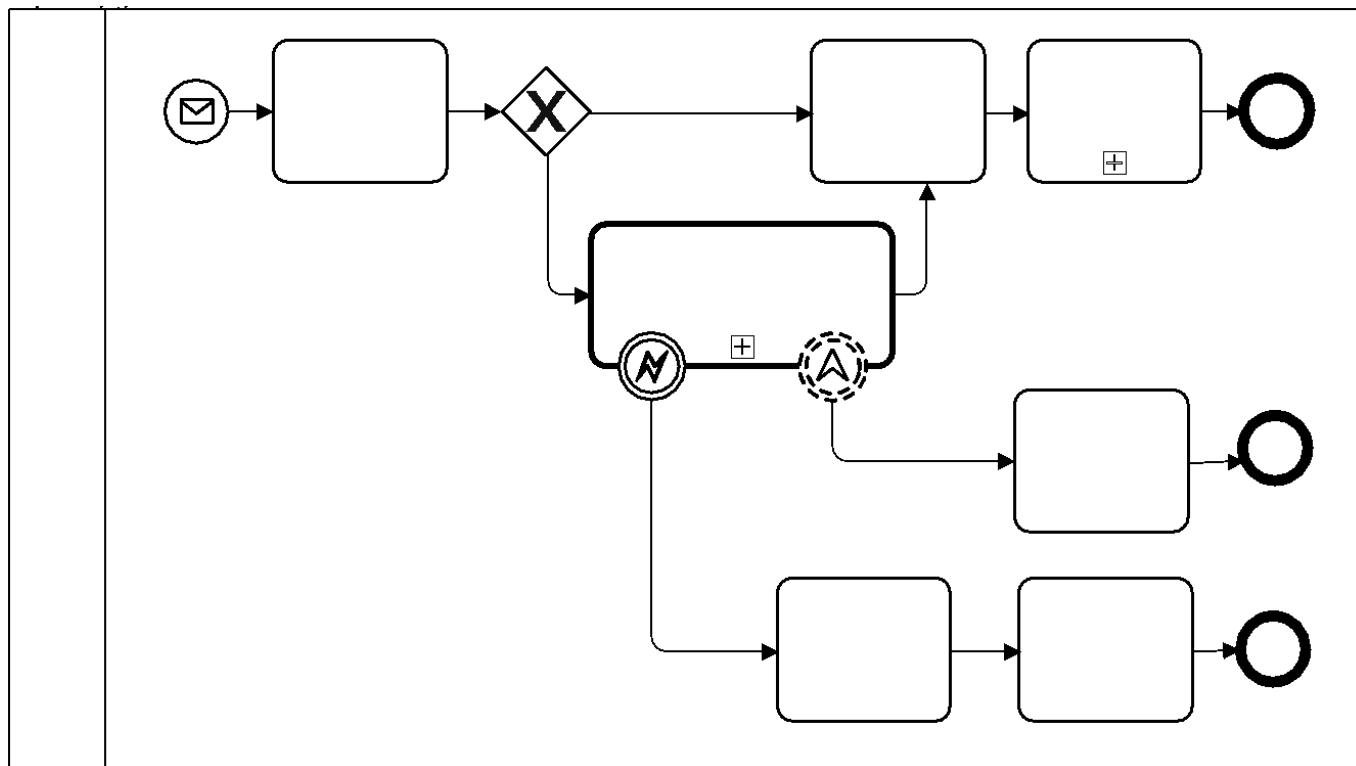
نماد	رویداد آغازین	نماد	رویداد میانی	نماد	رویداد پایانی	نماد	رویداد پیوسته شده
	خالی		خالی		خالی		دریافت کننده و متوقف کننده پیام
	دریافت کننده و متوقف کننده پیام		دریافت کننده و متوقف کننده پیام		ارسال کننده و متوقف کننده پیام		دریافت کننده و بدون توقف پیام
	دریافت کننده و بدون توقف پیام		ارسال کننده و متوقف کننده پیام		ارسال کننده سیگنال و متوقف کننده		وقوع زمان و متوقف کننده
	وقوع زمان و متوقف کننده		وقوع زمان و متوقف کننده		ارسال کننده یکی از چند فعال کننده و متوقف کننده		وقوع زمان و بدون توقف
	وقوع زمان و بدون توقف		دریافت کننده و متوقف کننده قاعده		ارسال کننده درخواست اجرا در سطح بالاتر و متوقف کننده		دریافت کننده و متوقف کننده قاعده
	دریافت کننده و متوقف کننده قاعده		دریافت کننده و متوقف کننده سیگنال		ارسال کننده خطا و متوقف کننده		دریافت کننده و بدون توقف قاعده
	دریافت کننده و بدون توقف قاعده		دریافت کننده و متوقف کننده یکی از چند فعال کننده		ارسال کننده درخواست بازایی موارد قبلی و متوقف کننده		دریافت کننده سیگنال و متوقف کننده

نماد	رویداد آغازین	نماد	رویداد میانی	نماد	رویداد پایانی	نماد	رویداد پیوست‌شده
	دریافت‌کننده سیگنال و متوقف‌کننده		ارسال‌کننده یکی از چند فعال‌کننده و متوقف‌کننده		ارسال‌کننده درخواست انصراف از اجرا و متوقف‌کننده		دریافت‌کننده سیگنال و بدون توقف
	دریافت‌کننده سیگنال و بدون توقف		دریافت‌کننده همزمان چند فعال‌کننده و متوقف‌کننده		ارسال‌کننده درخواست خاتمه کامل فرآیند و متوقف‌کننده		ارسال‌کننده سیگنال و متوقف‌کننده
	دریافت‌کننده یکی از چند فعال‌کننده و متوقف‌کننده		دریافت‌کننده درخواست اجرا در سطح بالاتر و متوقف‌کننده				دریافت‌کننده یکی از چند فعال‌کننده و متوقف‌کننده
	دریافت‌کننده یکی از چند فعال‌کننده و بدون توقف		ارسال‌کننده درخواست اجرا در سطح بالاتر و متوقف‌کننده				دریافت‌کننده یکی از چند فعال‌کننده و بدون توقف
	دریافت‌کننده همزمان چند فعال‌کننده و متوقف‌کننده		ارسال‌کننده درخواست بازیابی موارد قبلی و متوقف‌کننده				دریافت‌کننده همزمان چند فعال‌کننده و متوقف‌کننده
	دریافت‌کننده همزمان چند فعال‌کننده و بدون توقف		دریافت‌کننده نشان ادامه نمودار و متوقف‌کننده				دریافت‌کننده همزمان چند فعال‌کننده و بدون توقف
	دریافت‌کننده درخواست اجرا در سطح بالاتر و متوقف‌کننده		ارسال‌کننده نشان ادامه نمودار و متوقف‌کننده				دریافت‌کننده درخواست اجرا در سطح بالاتر و متوقف‌کننده
	دریافت‌کننده درخواست اجرا در سطح بالاتر و بدون توقف						دریافت‌کننده درخواست اجرا در سطح بالاتر و بدون توقف
	دریافت‌کننده خطا و متوقف‌کننده						دریافت‌کننده خطا و متوقف‌کننده
	دریافت‌کننده درخواست بازیابی موارد قبلی و متوقف‌کننده						دریافت‌کننده درخواست بازیابی موارد قبلی و متوقف‌کننده
							دریافت‌کننده درخواست انصراف از اجرا و متوقف‌کننده

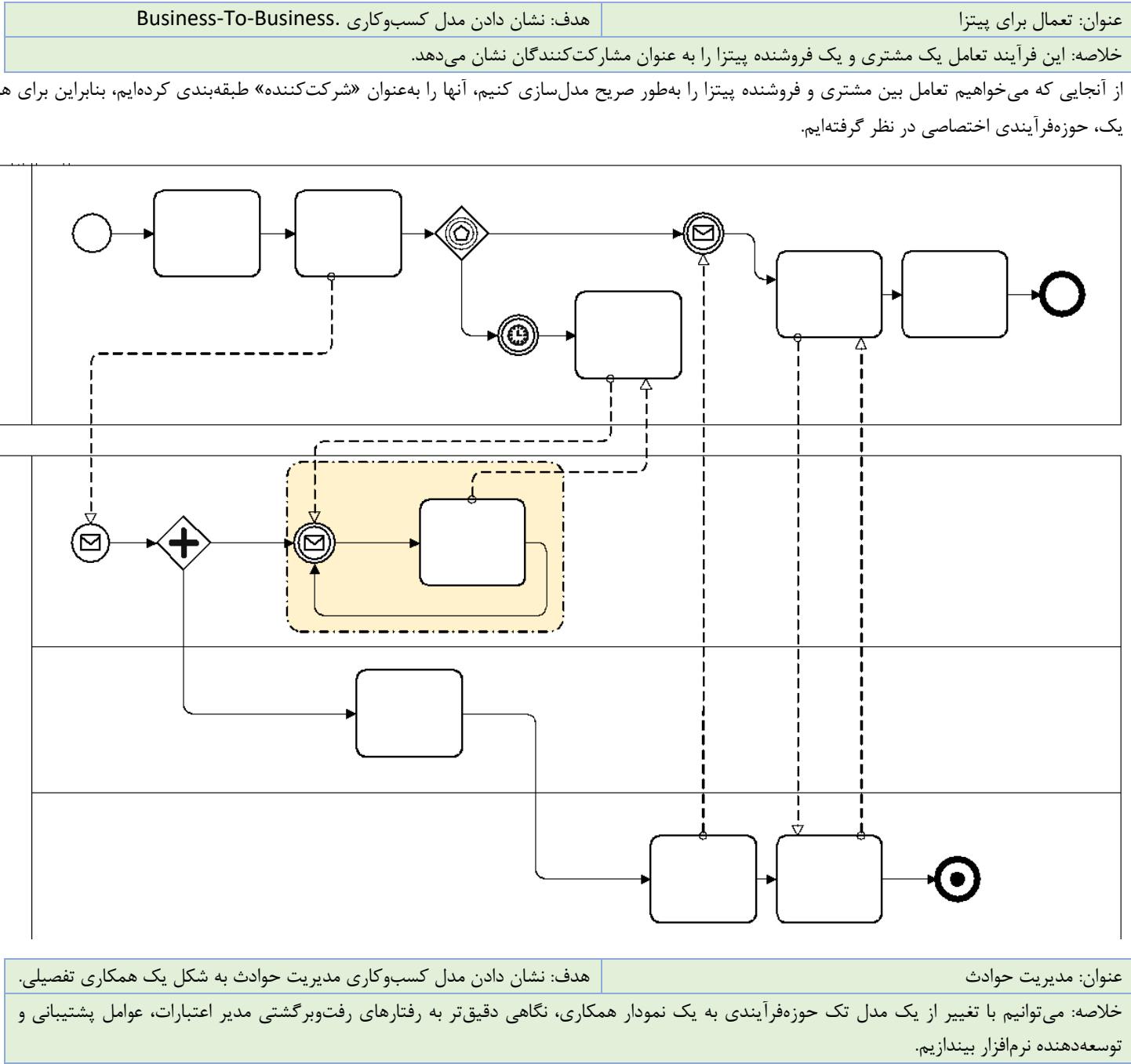
- محدوده فرآیند را با کشف پاسخ پرسش‌های چه کسی؟ چه چیزی؟ چه زمانی؟ کجا؟ و چرا؟ برای فرآیند مورد نظر، تعریف کنید و با این کار پاسخ پرسش چگونه فرآیند پیش می‌رود؟ را پیدا کنید
- مشخص کنید که هر نمونه از فرآیند چه چیزی را نشان می‌دهد سپس موارد مشکوک را شناسایی کنید
- با استفاده از رویدادهای آغازین مختلف راه‌های جایگزین بالقوه برای راه‌اندازی فرآیند را شناسایی کنید
- با استفاده از رویدادهای پایانی مختلف، وضعیت‌های نهایی جایگزین بالقوه نمونه‌های فرآیند را شناسایی کنید
- نمودارهای BPMN را به گونه‌ای طراحی کنید که در یک صفحه قابل ترسیم باشند
- عناصر درون نمودارهای BPMN خود را به طور چیدمان کنید که حداقل هم پوشانی جریان را داشته باشید تا خوانایی آن افزایش یابد
- سعی کنید از الگوی قاعده‌مندی تبعیت کنید که باعث شود جریان‌های توالی به صورت افقی و ارتباطات داده و جریان‌های پیام به صورت عمودی قرار گیرند و همچنین روند پیشروی منطق جریان اجرا از چپ به راست و از بالا به پایین باشد (البته ممکن است به دلیل منطق اجرای فرآیند، استفاده از این پیشنهاد، امکان‌پذیر نباشد)
- از چیدمان زیگزاگ عناصر استفاده نکنید
- باید مشخص باشد که مسیر اصلی (مسیر مورد انتظار یا بهترین مسیر فرآیند) چیست برای همین منظور حداقل برای این مسیر قاعده اجرای اجرا از چپ به راست و از بالا به پایین را پیاده کنید
- تا حد امکان، قواعد کسب‌وکاری را از فرآیند خارج کنید تا با استفاده از وظایف بررسی قواعد کسب‌وکار، مدل‌های فرآیند مختصر و چابک‌تری ایجاد کنید
- تجسم‌های جایگزین زیر را برای هر فرآیند ایجاد کنید تا بتوانید فرآیند مورد نظر را برای ذینفعان مختلف معرفی کنید:
  - یک نمودار خلاصه با تمام زیرفرآیندهای پنهان، وظایف فراخوانی و زیرفرآیندهای فراخوانی مبهم که در آن هیچ شیء داده‌ای ترسیم نشده باشد
  - یک نمودار مفصل با تمام زیرفرآیندهای گسترده و زیرفرآیندهای مرجع که در آن همه اشیاء داده و حاشیه‌نویسی‌ها ترسیم شده باشد
- یک سلسله مراتب چندلایه از جزئیات را برای فرآیند خود ایجاد کنید
- از زیرفرآیندها برای تقسیم فرآیند خود به "فازها" استفاده کنید
- از وظایف فراخوانی و زیرفرآیندهای فراخوانی برای استفاده مجدد از سایر فرآیندها استفاده کنید
- همیشه از رویدادهای آغازین و پایانی استفاده کنید
- بین نمونه‌های جایگزین فرآیند که با رویدادهای آغاز جداگانه آغاز می‌شوند، تمایز بگذارید
- وضعیت‌های پایانی مختلف را به رویدادهای پایانی جداگانه، متمایز کنید
- جریان‌هایی که به یک وضعیت پایانی مشابه ختم می‌شوند باید با همان رویداد پایانی ادغام کنید
- همیشه از گذرگاه‌ها برای ترسیم تقسیم یا ادغام جریان‌ها به صورت منفک شده استفاده کنید و سعی کنید از مدل ترکیبی گذرگاه‌ها (واگرا و همگرا با هم) استفاده نکنید
- همیشه یک فعالیت را که شرط (یا شرایط) واگرایی را بررسی می‌کند درست قبل از یک گذرگاه واگرا از نوع گذرگاه انتخاب انحصاری یا گذرگاه انحصاری شرطی، گذرگاه انتخاب اجتماعی و گذرگاه پیچیده قرار دهید تا نتیجه بررسی را به گذرگاه مورد نظر، ارائه نماید. با این کار، باعث می‌شود که برای بررسی شرط (یا شرایط) مورد نظر، وقفه‌ای که نیاز دارید ایجاد شود
- یک سلسله مراتب چندلایه گذرگاه‌های واگرا، زنجیره‌ای را ایجاد می‌کند که می‌توانید آن را در یک وظیفه بررسی قواعد کسب‌وکار خلاصه کنید که با این کار نمودار فرآیند خود را ساده‌تر می‌کنید
- زیرفرآیند داخلی اشاره شده در یک نمودار را به صورت یک حوزه فرآیند، مدل نکنید زیرا در این صورت قواعد نامگذاری زیرفرآیند و حوزه فرآیند باعث سردرگمی طراح و یا بیننده نمودار می‌شود

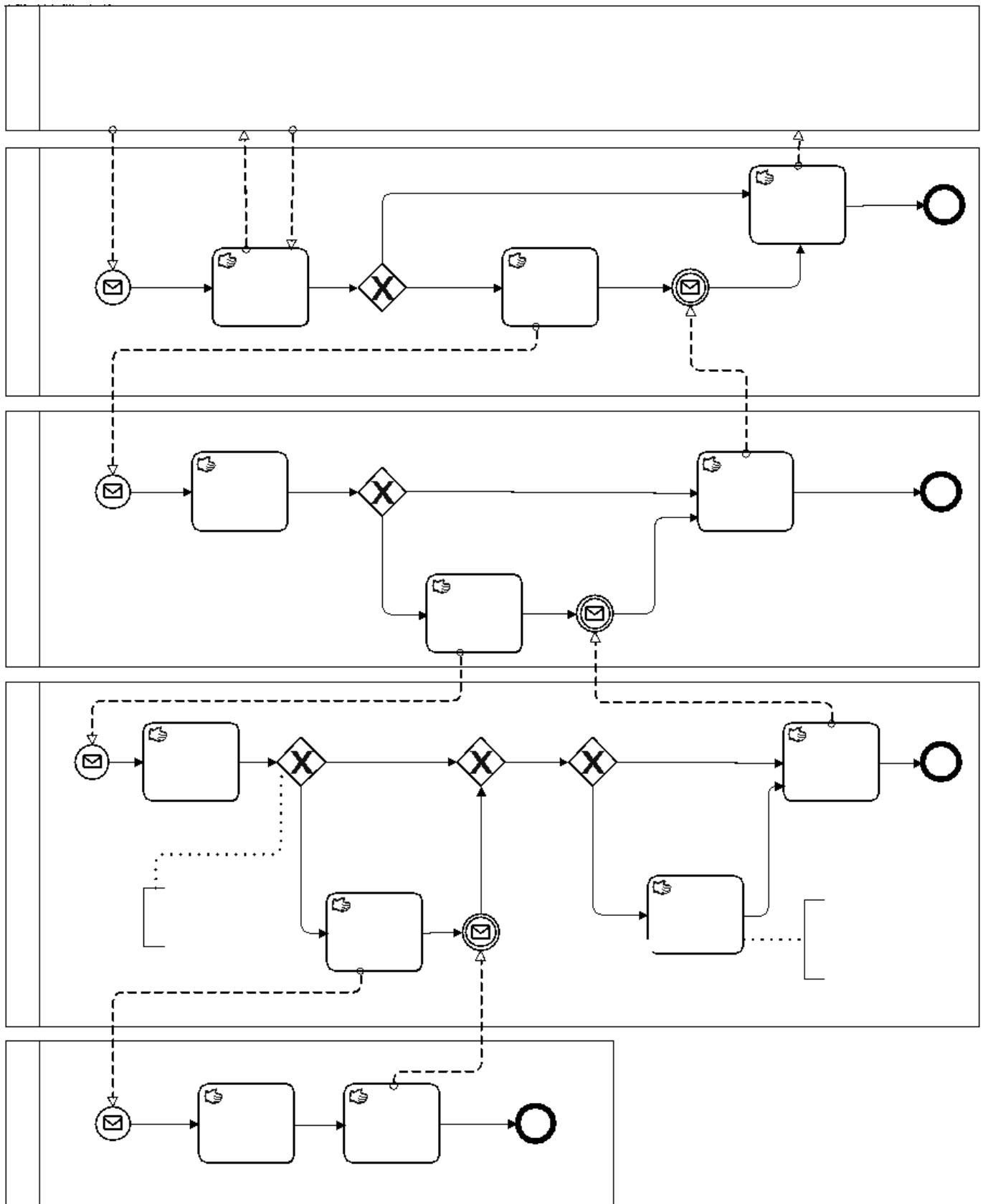
خلاصه: این فرآیند پس از دریافت پیام مربوط به سفارش، کنترل موجودی انجام می دهد. آیتم مربوط به سفارش مورد نظر به مشتری یا درخواست کننده ارسال می شود و سپس تصویه حساب مالی صورت می گیرد.

فرآیند تکمیل سفارش پس از دریافت پیام سفارش، شروع می شود و برای بررسی اینکه آیا آیتم سفارش داده شده، موجود است یا خیر، ادامه می یابد. آیتم موجود برای مشتری ارسال می شود و پس از آن تسویه حساب مالی انجام می شود که در این نمودار یک زیرفرآیند است. در صورتی که آیتمی موجود نباشد، باید با فراخوانی زیرفرآیند تدارکات، تهیه شود. لطفاً توجه داشته باشید که شکل این زیرفرآیند دارای حاشیه ضخیم است که به این معنی است که یک زیرفرآیند فراخوانی است.









فرآیندمفهومی	<b>Abstract Process</b>	فرآیندی که نشان دهنده تعاملات بین یک فرآیند کسبوکاری خصوصی و یک فرآیند یا شرکت کننده دیگر است.
فعالیت	<b>Activity</b>	کاری که یک شرکت یا سازمان در فرآیندهای کسبوکاری انجام می دهد. یک فعالیت می تواند اتمی (غیرقابل تفکیک) یا غیر اتمی (ترکیبی) باشد. انواع فعالیت هایی که بخشی از یک مدل فرآیند هستند عبارتند از: فرآیند، زیرفرآیند و وظیفه.
مصنوع	<b>Artifact</b>	یک شیء گرافیکی که اطلاعاتی در مورد فرآیند یا عناصر درون فرآیند ارائه می دهد. با این حال، مستقیماً بر جریان فرآیند تأثیر نمی گذارد.
ارتباط	<b>Association</b>	یک شیء اتصال برای پیوند دادن اطلاعات و مصنوعات با عناصر جریان استفاده می شود. یک ارتباط به صورت یک خط منقطع با یک، دو و یا هیچ سر فلش گونه که برای نشان دادن جهت جریان مفهومی استفاده می شود، نشان داده می شود.
فعالیت اتمی	<b>Atomic Activity</b>	فعالیتی که قابل تجزیه به جزئیات دقیق تری در مدل فرآیند نیست. این نوع فعالیت ها در سلسله مراتب ساختار فعالیت های فرآیند در جایگاه برگ هستند. از نظر گرافیکی به شکل یک نماد وظیفه در BPMN نشان داده می شود.
سیستم BPM	<b>BPM System</b>	فناوری که BPM را پشتیبانی کرده و در سازمان، فعال می کند.
تحلیلگر کسبوکار	<b>Business Analyst</b>	متخصصی که نیازها و مشکلات کسب کار را تجزیه و تحلیل می کند، با کاربران و ذینفعان مشورت می کند تا راه کارهایی را برای بهبود کسبوکار از طریق فناوری اطلاعات شناسایی کند و الزامات را در فرآیندهای کسبوکاری تعریف، مدیریت و نظارت می کند.
فرآیند کسبوکار	<b>Business Process</b>	مجموعه تعریف شده ای از فعالیت های کسبوکاری که بیانگر مراحل مورد نیاز برای دستیابی به یک هدف کسبوکاری است.
مدیریت فرآیند کسبوکار	<b>Business Process Management</b>	سرویس ها و ابزارهایی که از مدیریت فرآیند پشتیبانی می کنند (به عنوان مثال، تجزیه و تحلیل فرآیند، تعریف، پردازش، نظارت و مدیریت)، حتی از تعامل در سطح انسانی و برنامه نیز پشتیبانی می کنند. ابزارهای BPM می توانند فرآیندهای دستی را حذف کرده و مسیریابی درخواست ها را بین بخش ها و برنامه ها به طور خودکار انجام دهند.
هارمونی و هماهنگی	<b>Choreography</b>	دنباله ای پیوسته از تبادل پیام B2B بین دو یا چند شرکت کننده. در یک هارمونی و هماهنگی هیچ کنترل کننده مرکزی، نهاد مسئول، یا ناظر فرآیند وجود ندارد.
همکاری یا تعامل	<b>Collaboration</b>	همکاری عمل ارسال پیام بین دو شرکت کننده در یک مدل BPMN است. در این حالت دو شرکت کننده دو فرآیند BPML مجزا را نشان می دهند.
زیرفرآیند پنهان	<b>Collapsed Sub-Process</b>	یک زیرفرآیند که جزئیات جریان خود را پنهان می کند. شیء زیرفرآیند پنهان یک نشانگر به شکل یک مربع کوچک با علامت {+} در داخل خود دارد.
جریان جبران	<b>Compensation Flow</b>	جربانی که مجموعه فعالیت هایی را تعریف می کند که در حین بازگرداندن تراکنش انجام می شوند تا فعالیت هایی را که در جریان عادی فرآیند انجام شده اند، با حالت قبل بازگردانند و یا جبران کنند. یک جریان جبرانی را می توان از یک رویداد جبران یا رویداد میانی نیز فراخوانی کرد.
فعالیت مرکب	<b>Compound Activity</b>	فعالیتی است که حداقل از دو فعالیت تشکیل شده باشد که به واسطه حداقل یک جربانی توالی شرطی و یا یک گذرگاه به هم وابسته و یا مشروط شده باشند.
تصمیم گیری	<b>Decision</b>	

گذرگاهی در یک فرآیند کسب و کار که در آن جریان توالی می تواند یکی از چندین مسیر جایگزین را طی کند. همچنین این گذرگاه با عنوان "Or-Split" نیز شناخته می شود.	
رویداد پایانی	<b>End Event</b>
رویدادی که نشان می دهد یک مسیر فرآیند به کجا ختم می شود. رویداد پایانی به جریان های توالی مرتبط با خود در فرآیند پایان می دهد و بنابراین، هیچ جریان توالی خروجی نخواهد داشت. یک رویداد پایانی می تواند یک نتیجه خاص داشته باشد. این نتایج به شکل یک نماد در مرکز نماد رویداد پایانی ظاهر می شود. نتایج رویداد پایانی عبارتند از پیام، خطا، جبران، سیگنال، پیوند و چندگانه. رویداد پایانی همان شکل اصلی رویداد آغازین و رویداد میانی را دارد، یک دایره، اما با یک خط منفرد ضخیم رسم شده است.	
محتوای رویداد	<b>Event Context</b>
محتوای رویداد که فعال کننده رویداد نیز نامیده می شود، فعالیتی است که یا مورد انتظار است که وقوع یابد یا بر اساس واقع رسیدن توکن به رویداد مورد نظر، اجرا می شود.	
استثنا	<b>Exception</b>
رویدادی که در حین اجرای فرآیند رخ می دهد و باعث انحراف از جریان عادی فرآیند می شود. استثنایها می توانند توسط رویدادهای میانی مانند زمان، خطا یا پیام، ایجاد شوند.	
جریان استثنایی	<b>Exception Flow</b>
یک جریان توالی که از یک رویداد میانی پیوسته شده به فعالیت سرچشمه می گیرد. فرآیند از این مسیر عبور نمی کند مگر اینکه فعالیت با فعال سازی یک رویداد میانی پیوسته شده، قطع شود (یک استثنا - به بیرون از مرز فعالیت در حال اجرا مراجعه کنید).	
زیر فرآیند گسترده	<b>Expanded Sub-Process</b>
یک زیر فرآیند که جزئیات جریان خود را در چارچوب فرآیند اصلی خود نشان می دهد. یک زیر فرآیند گسترده به شکل یک مستطیل گوشه گرد نمایش داده می شود که برای نمایش اشیاء جریان در داخل آن بزرگ شده است.	
جریان	<b>Flow</b>
یک رابط جهت دار بین عناصر نمودار فرآیند، نمودار همکاری و یا نمودار هارمونی و هماهنگی، می باشد. یک جریان توالی نشان دهنده دنباله جریان اشیاء در یک نمودار فرآیند یا نمودار هارمونی و هماهنگی است. یک جریان پیام نشان دهنده انتقال یک پیام بین شرکت کنندگان در نمودار همکاری است. اصطلاح جریان اغلب برای نشان دادن پیشرفت کلی نحوه انجام یک فرآیند یا بخشی از فرآیند استفاده می شود.	
شیء جریان	<b>Flow Object</b>
یک شیء گرافیکی که می تواند به یک جریان توالی متصل شود. اشیاء جریان در یک نمودار فرآیند یا نمودار هارمونی و هماهنگی، رویدادها، فعالیتها و گذرگاهها هستند.	
چندشاخه	<b>Fork</b>
نقطه ای در فرآیند که در آن یک جریان توالی به دو یا چند جریان تقسیم می شود که به صورت موازی در فرایند، اجرا می شوند و به چندین فعالیت اجازه می دهد به طور همزمان و نه به صورت متوالی اجرا شوند. BPMN از چندین جریان توالی خروجی از فعالیتها یا رویدادها یا یک گذرگاه موازی برای پیاده سازی حالت چندشاخه استفاده می کند. همچنین به عنوان "AND-Split" نیز شناخته می شود.	
رویداد میانی	<b>Intermediate Event</b>
رویدادی که پس از شروع یک فرآیند رخ می دهد. یک رویداد میانی با نشان دادن مکان هایی که پیامها و تاخیرها مورد انتظار هستند، ایجاد جریان های جایگزین از طریق مدیریت استثنا، یا ترسیم جریان جبران، بر جریان فرآیند تأثیر می گذارد. با این حال، یک رویداد میانی یک فرآیند را شروع یا مستقیماً خاتمه نمی دهد. یک رویداد میانی به صورت دایره ای نمایش داده می شود که با یک خط دوتایی نازک ترسیم شده است.	
پیوستن	<b>Join</b>
نقطه ای در فرآیند که در آن دو یا چند جریان توالی موازی در یک جریان توالی، ترکیب می شوند. BPMN از یک گذرگاه موازی برای انجام حالت پیوستن استفاده می کند. همچنین به عنوان "AND-Join" نیز شناخته می شود.	
مسیر	<b>Lane</b>
پارتیشنی که برای سازماندهی و دسته بندی فعالیتها در یک حوزه فرآیندی، استفاده می شود. یک مسیر فرآیند، تمام طول حوزه فرآیند را به صورت عمودی یا افقی در برمی گیرد. مسیرها اغلب برای مواردی مانند نقش های داخلی (به عنوان مثال، مدیر، همکار)، سیستمها (به عنوان مثال، یک برنامه سازمانی)، یا یک بخش داخلی (مانند حمل و نقل، مالی) استفاده می شوند.	

ادغام	<b>Merge</b>	نقطه‌ای در فرآیند که در آن دو یا چند جریان‌توالی مختلف در یک جریان‌توالی ترکیب می‌شوند. در این حالت هیچ هماهنگی لازم نیست زیرا هیچ فعالیت موازی در نقطه اتصال اجرا نمی‌شود. BPMN از چندین جریان‌توالی ورودی برای یک فعالیت یا یک گذرگاه انتخاب‌انحصاری یا گذرگاه انحصاری شرطی برای انجام ادغام، استفاده می‌کند. همچنین به عنوان "OR-Join" نیز شناخته می‌شود.
پیام	<b>Message</b>	پیام شیء است که محتوای ارتباط بین دو شرکت‌کننده را به تصویر می‌کشد. یک پیام از طریق یک جریان‌پیام منتقل می‌شود.
جریان‌پیام	<b>Message Flow</b>	یک نماد اتصال‌دهنده که جریان پیام‌ها را بین دو شرکت‌کننده، نشان می‌دهد. یک جریان‌پیام با یک خط منقطع نشان داده می‌شود.
جریان‌عادی	<b>Normal Flow</b>	جربانی که از یک رویداد آغازین سرچشمه می‌گیرد و از طریق فعالیت در مسیرهای جایگزین و موازی تا رسیدن به یک رویداد پایانی ادامه می‌یابد.
فرآیند والد	<b>Parent Process</b>	فرآیندی که یک زیرفرآیند را در محدوده خود دارد.
شرکت‌کننده	<b>Participant</b>	یک نهاد تجاری (به عنوان مثال، یک شرکت، بخش شرکت، یا یک مشتری) یا یک نقش کسب‌وکاری (به عنوان مثال، خریدار یا فروشنده) که کنترل‌کننده یا مسئول یک فرآیند کسب‌وکاری است. اگر از حوزه‌های فرآیند استفاده شود، یک شرکت‌کننده با یک حوزه فرآیند معادل می‌شود. در یک همکاری، شرکت‌کنندگان به طور غیر رسمی به عنوان "حوزه فرآیند" شناخته می‌شوند.
حوزه فرآیند	<b>Pool</b>	یک حوزه فرآیند نشان دهنده یک شرکت‌کننده در یک همکاری است. از نظر گرافیکی، یک حوزه فرآیند محفظه‌ای برای پارتیشن‌بندی یک فرآیند از سایر حوزه فرآیند یا شرکت‌کنندگان است. یک حوزه فرآیند برای داشتن یک فرآیند لازم نیست، به عنوان مثال، یک فرآیند می‌تواند یک کادر نامشخص از جزئیات فرآیند باشد.
فرآیند کسب‌وکار اختصاصی	<b>Private Business Process</b>	فرآیندی که فقط درون یک طبقه‌بندی خاص است و نوعی فرآیند است که به طور کلی گردش کار یا فرآیند BPM نامیده می‌شود.
فرآیند	<b>Process</b>	فرآیند، توالی یا جریانی از فعالیت‌ها در یک سازمان با هدف انجام کار است. در BPMN، یک فرآیند به عنوان نموداری از عناصر جریان نشان داده می‌شود که مجموعه‌ای از فعالیت‌ها، رویدادها، گذرگاه‌ها و جریان‌های توالی هستند.
نتیجه	<b>Result</b>	نتیجه، پیامد رسیدن به یک رویداد پایانی است. انواع نتایج عبارتند از پیام، خطا، جبران، سیگنال، پیوند و چندگانه.
جریان‌توالی	<b>Sequence Flow</b>	جریان‌توالی یک شیء متصل‌کننده است که ترتیب انجام فعالیت‌ها در یک فرآیند را نشان می‌دهد و با یک خط گرافیکی ثابت نشان داده می‌شود. هر جریان فقط یک محل آغاز و فقط یک محل خاتمه دارد. یک جریان‌توالی می‌تواند از مرزهای بین مسیرهای یک حوزه فرآیند عبور کند اما نمی‌تواند از مرزهای یک حوزه فرآیند عبور کند.
رویداد آغازین	<b>Start Event</b>	رویدادی که نشان دهنده محل آغاز فرآیند مورد نظر است. رویداد آغازین، جریان فرآیند را شروع می‌کند و هیچ جریان‌توالی ورودی ندارد، اما می‌تواند یک محتوای رویداد یا فعال‌کننده داشته باشد. رویداد آغازین به صورت دایره‌ای نمایش داده می‌شود که با یک خط نازک کشیده شده است.
زیرفرآیند	<b>Sub-Process</b>	فرآیندی که در یک فرآیند دیگر گنجانده شده است. زیرفرآیند می‌تواند به شکل پنهان‌شده باشد که جزئیات آن قابل مشاهده نیست یا می‌تواند در یک نمای گسترش‌یافته باشد که جزئیات آن را درون محدوده فرآیند می‌توان مشاهده نمود. یک زیرفرآیند شکلی مشابه به شکل ظاهری وظیفه دارد.
مسیر پیشرفت	<b>Swimlane</b>	مسیر پیشرفت یک کادر گرافیکی برای طبقه‌بندی یا تفکیک مجموعه‌ای از فعالیت‌ها از فعالیت‌های دیگر است. BPMN دو نوع مسیر پیشرفت مختلف دارد. برای اطلاع از این دو نوع به توضیحات محدوده فرآیند و مسیر فرآیند مراجعه کنید.
وظیفه	<b>Task</b>	

<p>یک فعالیت اتمی که در یک فرآیند گنجانده شده است. زمانی از وظیفه استفاده می‌شود که کار در فرآیند به سطح دقیق‌تری از جزئیات مدل فرآیند تجزیه نشود. به طور کلی، یک کاربر نهایی، یک برنامه کاربردی یا هر دو، می‌توانند وظیفه را انجام دهند. یک شیء جریان از نوع وظیفه از نظر ظاهر مانند یک زیرفرآیند است.</p>	
<p><b>Token</b></p> <p>توکن</p> <p>یک مفهوم نظری که به عنوان یک موضوع کمکی برای تعریف رفتار زمان اجرای یک فرآیند، استفاده می‌شود. رفتار عناصر فرآیند را می‌توان با توصیف نحوه تعامل آنها با یک توکن در حال عبور از ساختار داخلی فرآیند، تعریف کرد. به عنوان مثال، یک توکن از یک گذرگاه انتخاب انحصاری عبور می‌کند، اما تنها در امتداد یکی از جریان‌های توالی خروجی این گذرگاه، ادامه می‌یابد.</p>	
<p><b>Transaction</b></p> <p>تراکنش</p> <p>تراکنش زیرفرآیندی است که مجموعه‌ای از فعالیت‌های هماهنگ را نشان می‌دهد که توسط سیستم‌های مستقل و بدون پیوند، مبتنی بر ارتباطات کسب‌وکاری تعریف شده، انجام می‌شود. این هماهنگی منجر به یک نتیجه توافق شده، منسجم و قابل تأیید در همه شرکت کنندگان می‌شود.</p>	
<p><b>Trigger</b></p> <p>فعال کننده</p> <p>فعال کننده، مکانیزمی است که یک رخداد را تشخیص می‌دهد و می‌تواند باعث پردازش اضافی در پاسخ شود، مانند شروع یک فرآیند کسب‌وکاری. فعال کننده‌ها با رویدادهای آغازین و رویدادهای میانی مرتبط هستند و می‌توانند از انواع ذکر شده، باشند: پیام، زمان، شرط، سیگنال، پیوند، و چندگانه.</p>	
<p><b>Uncontrolled Flow</b></p> <p>جریان کنترل نشده</p> <p>جریانی که بدون وابستگی یا عبارات شرطی ادامه دارد. به طور معمول، یک جریان کنترل نشده، یک جریان توالی بین دو فعالیت است که یک نشانگر شرطی (لوزی کوچک) یا یک گذرگاه میانی (لوزی بزرگ) ندارند.</p>	