

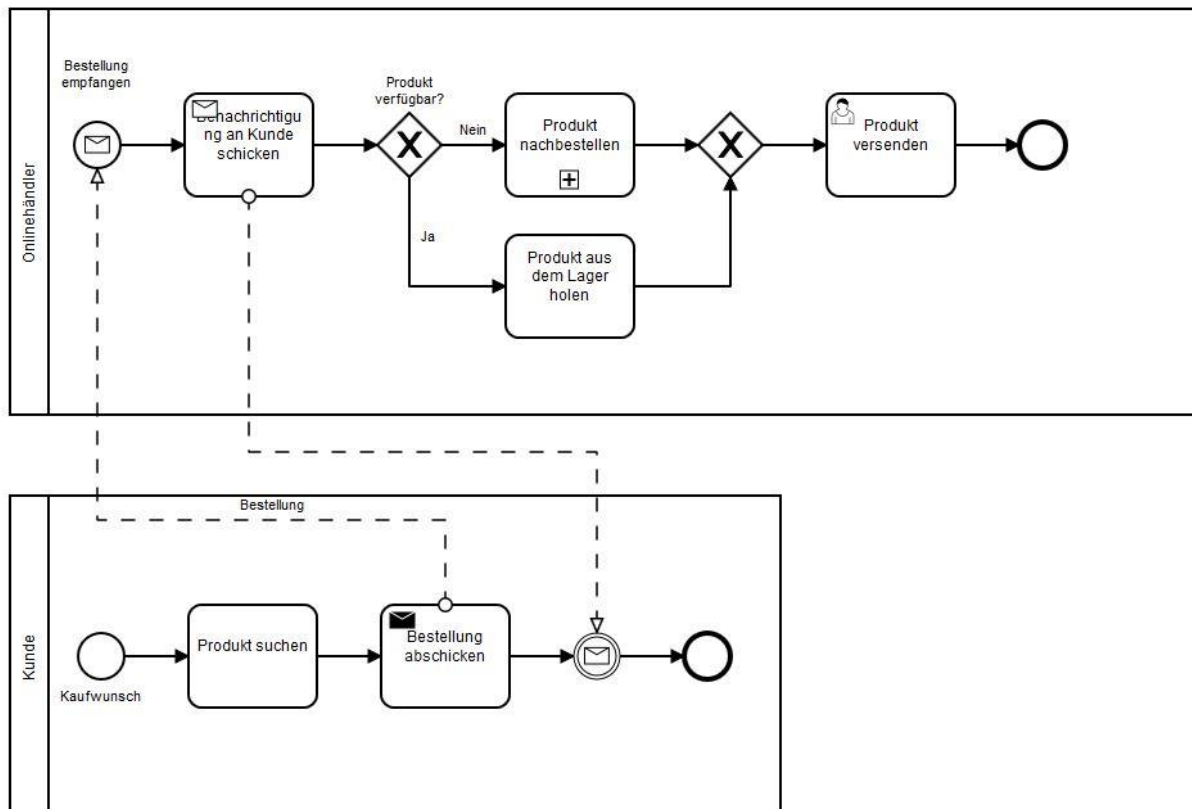
BPMN 2.0

In diesem Praktikum üben wir den Gebrauch verschiedener BPMN 2.0 Elemente.

Nutzen Sie dafür das folgende Online-Tool <https://demo.bpmn.io/>

Aufgabe 1 – Ein einfaches BPMN 2.0 Diagramm

Bauen Sie das abgebildete BPMN 2.0 Diagramm in dem Tool nach und speichern Sie es als .bpmn Datei.



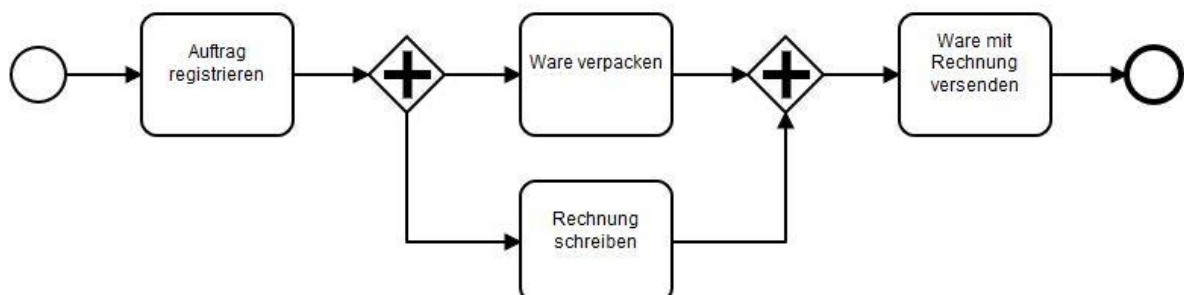
Öffnen Sie anschließend die Datei mit einem Editor.

Sie sollten erkennen, dass das Diagramm als XML repräsentiert wird.

Aufgabe 2 - Gateways

Gateways dienen der Prozesssteuerung. Mit Ihnen können Entscheidungen modelliert werden. Sie führen zu Verzweigungen oder Zusammenführungen.

Parallel Gateway



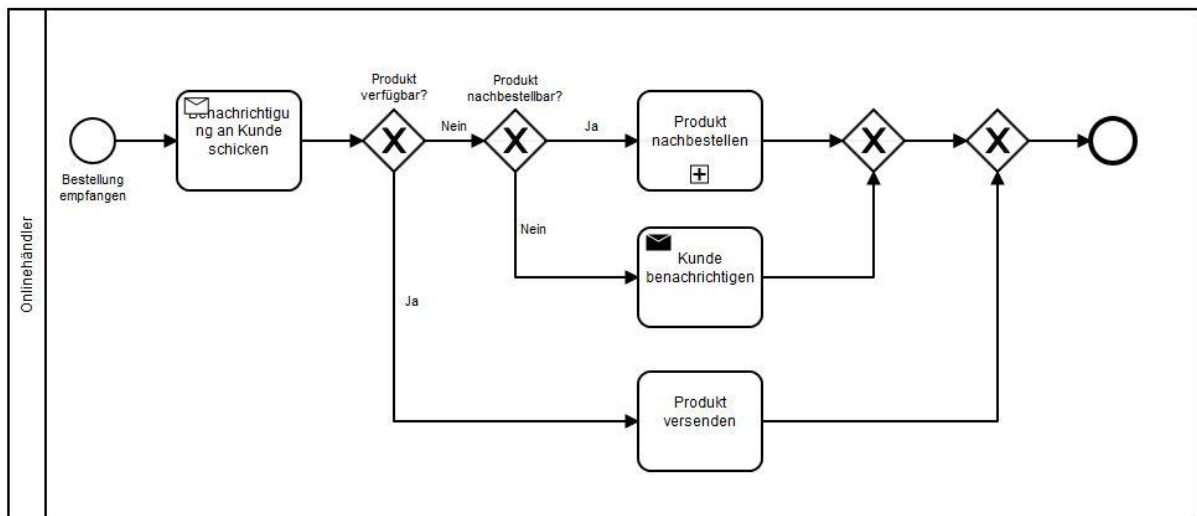
Beschreiben Sie den Ablauf mit eigenen Worten.

Wie steuern die beiden Parallel Gateways den Prozess?

Antwort

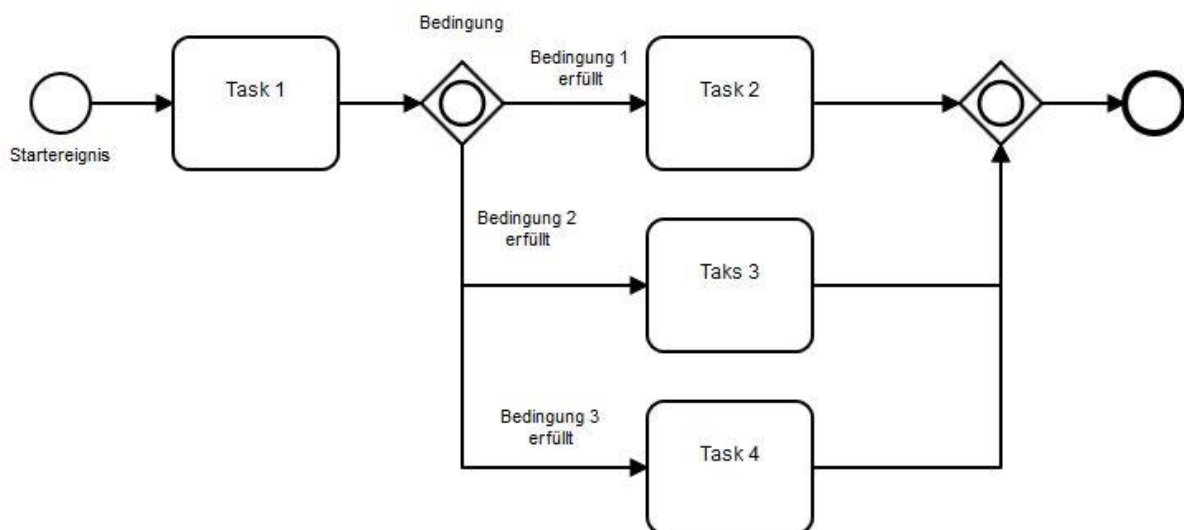
Das nach dem linken Parallel Gateway werden die Task sowohl Ware verpacken und Rechnung schreiben ausgeführt. Erst wenn beide Tasks erfüllt wurden, wird mit der Task Ware und Rechnung versenden fortgefahren.

Exclusive Gateway



Beschreiben Sie den Prozessablauf des oben gezeigten Diagramms.

Inclusive Gateway



Wie wird der Prozess nach dem ersten Inclusive Gateway fortgesetzt?

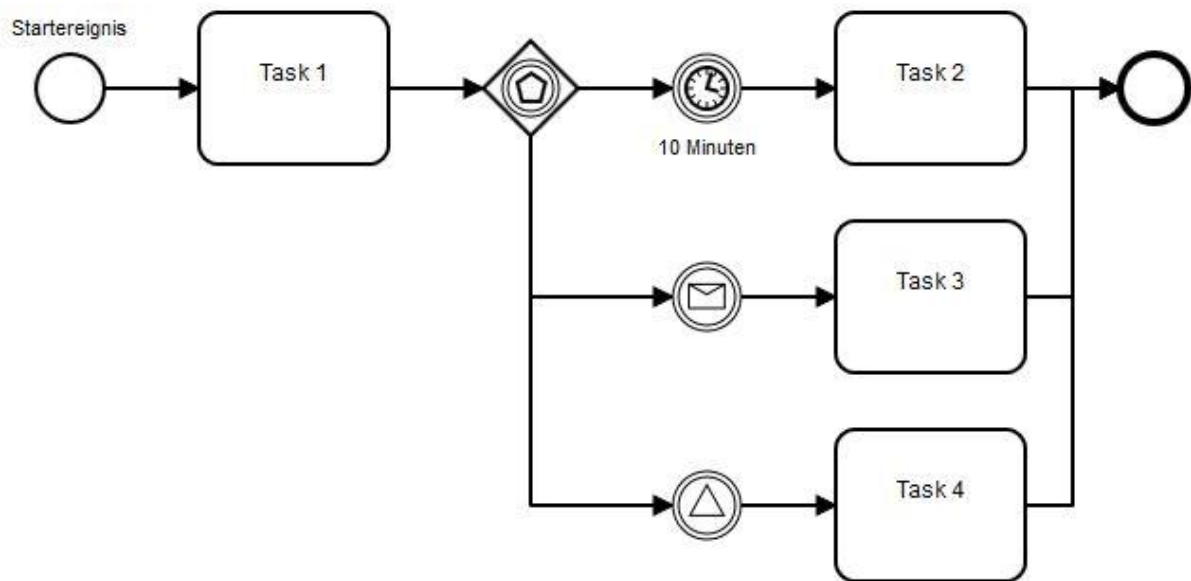
- Es kann mit einer, zwei oder drei Tasks weitergehen. Alle ausgehenden Pfade werden geprüft.

Wann wird der Prozess nach dem zusammenführenden Inclusive Gateway fortgesetzt?

- Das Gateway wartet auf alle aktiven Pfade.

Event Based Gateway

Modellieren Sie das folgende BPMN 2.0 Diagramm.



Was fällt Ihnen auf, wenn Sie die nachfolgenden Elemente des Event based Gateway auswählen?

Beschreiben Sie den Prozessablauf.

Lösung

- Es sind nur intermediate Catch Events möglich.
- Es wird die Task gewählt, dessen Event zuerst eintritt
- Tritt das Signal-Event zuerst ein, werden Timer Event und Message Catch Event gecancelt. Es wird mit Task 4 fortgefahren.
- Tritt das Message Catch Event zuerst ein, wird das Signal Event und der Timer gecancelt. Es wird mit Task 3 fortgefahren
- Wird innerhalb von 10 Minuten weder Message Event noch Signal Event ausgelöst, wird mit Task 2 fortgefahren.

Aufgabe 3 – Business Rules

Aufgabe 3a

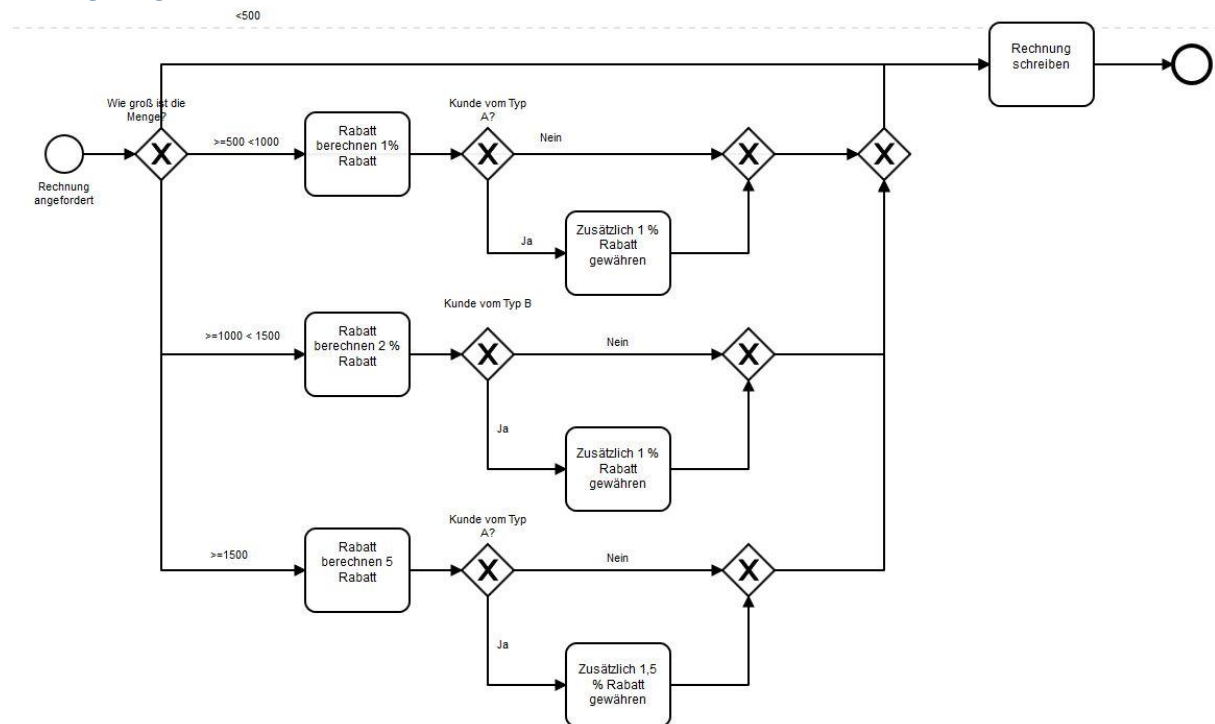
Modellieren Sie den folgenden Prozess. Ein Unternehmen erhält eine Bestellung. Abhängig von der Bestellmenge und dem Kundentyp wird ein Rabatt gewährt. Nach Berechnung des Rabattes wird die Rechnung erstellt.

- Menge < 500: kein Rabatt
- Menge ≥ 500 und < 1000: 1 % Rabatt. Sollte der Kunde vom Typ A sein, wird zusätzlich 1 % Rabatt gewährt.
- Menge ≥ 1000 und < 1500: 2 % Rabatt. Sollte der Kunde vom Typ B sein, wird zusätzlich 1 % Rabatt gewährt.
- Menge ≥ 1500 : 5 % Rabatt. Sollte der Kunde vom Typ A sein, wird zusätzlich 1,5 % Rabatt gewährt.

Nutzen Sie dafür ausschließlich folgende Elemente: Startevent, End Event, User Task (wird von einem menschlichen Nutzer ausgeführt), Exclusive Gateway.

Der Kunde bzw. Kommunikation mit ihm soll der Einfachheit halber nicht mit berücksichtigt werden.

Lösung Aufgabe 3a

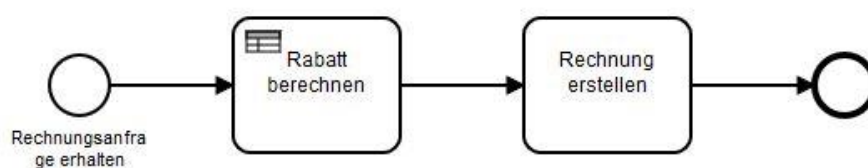


Aufgabe 3b

Das in Aufgabe 2a erstellte BPMN 2.0 Diagramm wird sehr unübersichtlich. Business Rules sollten nicht im BPMN-Diagramm abgebildet werden. Stattdessen sollte eine **Business Rule-Task** eingesetzt werden, die beschreibt, wie z. B. der Rabatt zu berechnen ist.

Stellen Sie den oberen Ablauf mithilfe einer Business Rule Task dar. Business Rule Tasks sollen eine oder mehrere Regeln synchron ausführen.

Lösung Aufgabe 3b



Kontrollfrage

Was ist der Unterschied zwischen einer User-Task und einer Business Rule Task?

Antwort


- User Task wird von einem Menschen ausgeführt.
- Eine Business Rule Task führt synchron eine oder mehrere Regeln aus.

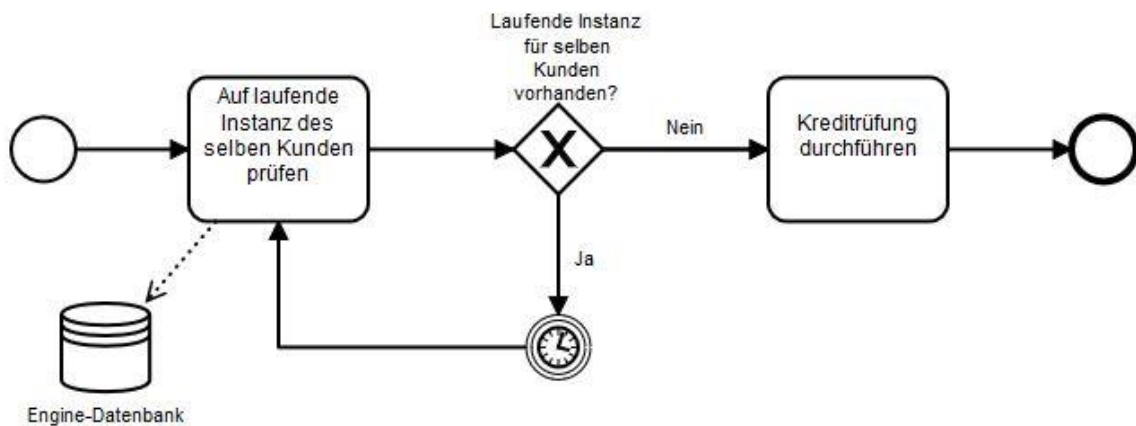
Aufgabe 4 - Events

Viele Prozesse können auf unterschiedliche Weise in BPMN 2.0 modelliert werden.



Es soll die Kreditwürdigkeit eines Kunden überprüft werden. Während die Überprüfung läuft, kann es sein, dass eine zweite Anfrage für denselben Kunden eintrifft. Diese soll jedoch erst dann bearbeitet werden, wenn die erste abgeschlossen ist.

Timer und Loop

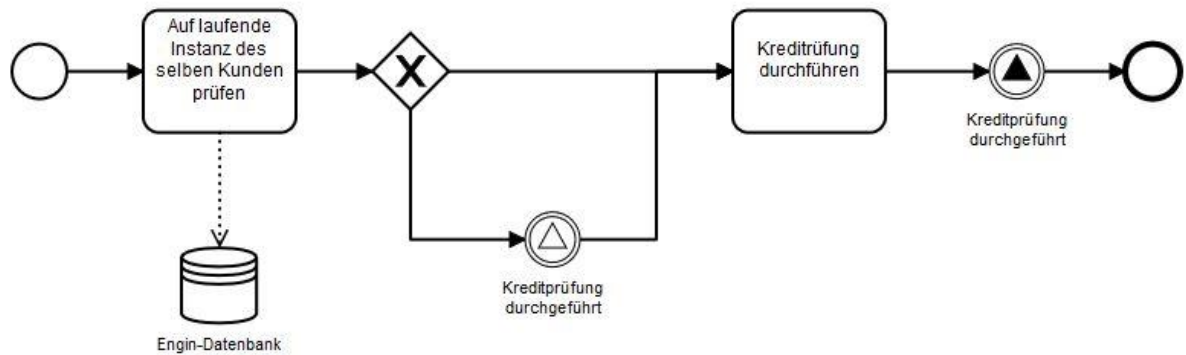
	<p>Intermediate Timer Event Erreicht ein Prozess ein Intermediate Timer Event, wird dieses nach Ablauf einer bestimmten Zeit ausgelöst.</p>
---	--



Signal Event

	<p>Intermediate Signal Catch Event: Hört auf ausgelöste Signal Throw Events. Sollte der Prozess an einem intermediate Signal Catch Event gelangen, wartet er, bis das entsprechende Signal Throw Event ausgelöst wird.</p>
	<p>Signal Throw Event: Ein Signal Event wird an alle aktiven Instanzen eines BPMN 2.0 Prozesses übermittelt.</p>

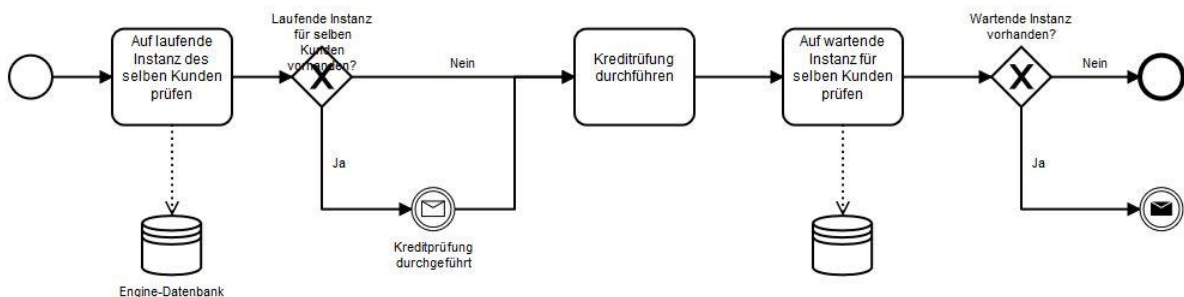
Lösung



Message Event

	Intermediate Message Catch Event: Hört auf ausgelöste Message Throw Events. Sollte der Prozess an einem intermediate Message Catch Event ankommen, wartet er, bis eine Message mit dem richtigen Namen eintrifft.
	Intermediate Message Throw Event: Ein Message Event wird an einen bestimmten Empfänger (Instanz). Messages haben einen Namen und eine Payload (Daten).

Lösung



Kontrollfrage

Wie unterscheiden sich Signal Events von Message Events?

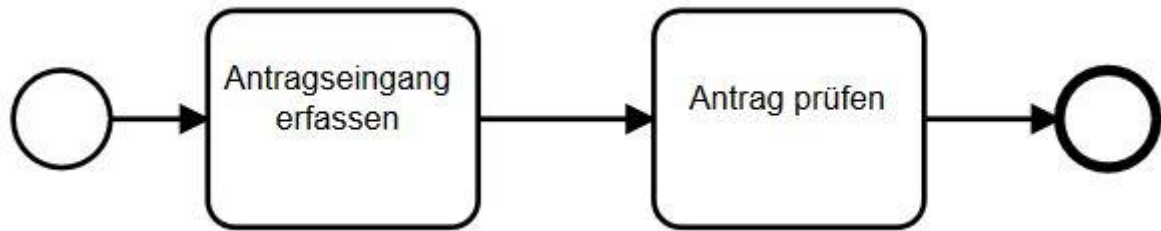
Antwort

- Signal Events werden an jede Instanz geschickt. Es gibt keinen festen Empfänger.
- Message Events haben einen genau angegebenen Empfänger.

Aufgabe 5 - Start Events, Boundary Events, Intermediate Events

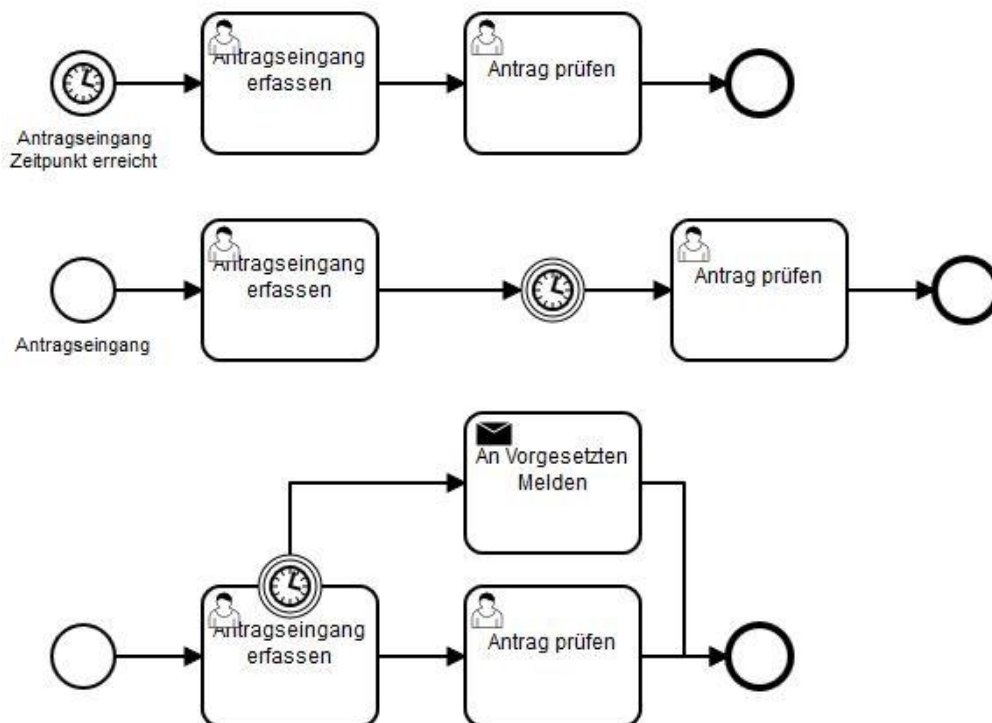
Events können an drei verschiedenen Stellen in einem BPMN 2.0 Diagramm vorkommen: Als Start Event, Boundary Event einer Task und als intermediate Event zwischen Tasks.

Modellieren Sie folgende Prozesse ausgehend von dem stark vereinfachten Prozess Urlaubsantrag bearbeiten. Antragseingang erfassen und Antrag prüfen sollen **User Tasks** sein.



- 1.) Der Prozess soll zu einem Bestimmten Zeitpunkt ausgelöst werden.
- 2.) Der Prozess Urlaubsantrag bearbeiten soll zwischen Erfassung des Eingang des Antrags und Bearbeitung mindestens 1 Tag unterbrochen werden.
- 3.) Dauer das Prüfen des Antrags zu lange, soll der Vorgesetzte informiert werden.

Lösung



Aufgabe 6 – Best Practices

Best Practices umfassen Modellierungsmethoden, die das Verstehen eines BPMN 2.0 Diagramms einfacher machen.

- Fluss-Kreuzungen vermeiden
- Übersichtliche Anordnung der Elemente
- Namensgebung:
 - Tasks: Objekt + Verb (Infinitiv)
 - Events: Objekt + Verb
 - Start Events sollten mit einem Auslöser beschriftet sein (Der Autor dieser Anleitung hat sich daran nicht gehalten!)
 - Pools: Sollten immer beschriftet sein
 - XOR Gateways sollten mit einer aussagekräftigen Frage beschriftet sein
- Symmetrische Modellierung: Jedes verzweigende Gateway hat sein eigenes zusammenführendes Gateway

- Task-Elemente sollten die gleiche Größe haben.

Finden Sie Fehler im folgenden BPMN 2.0 Diagramm hinsichtlich der Best Practices.

Siehe Datei: [diagram_best_practices_wrong.bpmn](#)

Korrigieren Sie die Fehler.

Lösung siehe Datei [diagram_best_practices_solution.bpmn](#)

- Oberer Pool nicht beschriftet
- Startereignis im oberen Pool nicht beschriftet
- G1: Exclusive Gateway nicht als Frage gestellt
- M1: Messageflow fehlt
- Zwischen G6 und G3: Symmetrische Modellierung missachtet. Es müsste noch ein zusammenführendes Exclusive Gateway rein. (Würde aber auch so funktionieren. Ist nur unübersichtlich)
- Ausrichtung der Elemente unordentlich
- G2: Bedingung für „Nein“ nicht angegeben
- G3: Bedingung für „Ja“ nicht angegeben
- Pfad zwischen T1 und G1 unklar.
- Es kreuzen sich verschiedene Pfade
- T7: Task nicht Objekt + Verb beschriftet
- T9: Task nicht Objekt + Verb beschriftet