



Requisitos do *Process Designer*

Link Consulting – Tecnologias de Informação, S.A.

Janeiro 2025





Índice

1	Introdução	4
2	Descrição Geral	5
2.1	Descrição do Process Designer	5
3	Definição de Processo	6
3.1	Definição de Processo de Negócio	6
4	Templates de Processo/Ready2Use	7
4.1	Copiar Template de Processo para outro Repositório Atlas	7
4.2	Apagar cópia de Template de Processo	7
4.3	Criar Template “Ready2Use (Fast-Setup)”	7
4.4	Criar Instância de Processo a partir de um Template “Fast-Setup”	8
5	Algoritmo de Partição de BPMN	9
5.1	Processar Elementos do BPMN	9
5.2	Agregar Ramos	10
5.3	Criar Fluxos de Sequência entre Ramos	10
5.4	Registar Estado do Algoritmo	11
6	Registo de SLAs sobre o Processo	12
6.1	Definir SLA para um Processo	12
7	Registo de Business Objects Utilizados no Processo	13
7.1	Definir Business Objects para uma Task	13
7.2	Passar Business Objects Entre Engines	13
8	Visualização de Tasks do Processo/Worklist	14
8.1	Ver Tasks na Worklist	14
8.2	Automatizar Execução de Script Task	14
9	eProcess Connector - Comunicação com eProcess	15
9.1	Implementar Processo em Múltiplos Engines	15
9.2	Enviar Atualização de Estado da Task para o Monitor	15
10	Visualização de Estatísticas e Mining do Processo	16
10.1	Ver Estatísticas do Processo	16
10.2	Realizar Process Mining	16
11	Integração com o GPT	17
11.1	Execução de Tarefas de Script com Prompts Textuais	17
11.2	Manuseamento de Entrada	17
11.3	Integração com a API do ChatGPT	18
11.4	Configuração do Batch	19
11.5	Análise de BPMN	19
11.6	Especificação do Script	19
11.7	Manuseamento da Saída	20
11.8	Estrutura do Batch – Configuração ATLAS	20
11.9	Configuração do Trabalho com Base no Tipo de Entrada	21
11.10	Manuseamento e Erros e Registos	21





1 Introdução

Este documento contém os requisitos do módulo “Process Designer” do projeto eProcess, sob a forma de casos de uso.



2 Descrição Geral

2.1 Descrição do Process Designer

O componente Process Designer do eProcess tem como principal função permitir ao utilizador a definição e configuração de Processos, assim como definir Service Level Agreements (SLAs), representando tempos limites que podem passar entre Tarefas de um Processo.

Internamente, este componente tem ainda a função de dividir o Processo, preparando-o para ser executado nos diferentes motores de execução e de comunicar toda esta informação ao componente "Process Monitor".



3 Definição de Processo

3.1 Definição de Processo de Negócio

- **Atores:** Utilizador Atlas
- **Pré-condições:** Utilizador autenticado
- **Fluxo:**
 1. Preencher formulário de criação de Business Process.
 2. Abrir editor de BPMN.
 3. Desenhar o Processo de acordo com os standards de BPMN.
 4. Guardar o desenho.
- **Pós-condições:** Um novo “Business Process” foi criado no Atlas com o respetivo diagrama BPMN.



4 Templates de Processo/Ready2Use

4.1 Copiar Template de Processo para outro Repositório Atlas

- **Atores:** Utilizador Atlas
- **Pré-condições:** Utilizador autenticado; Utilizador com acesso a pelo menos 2 repositórios
- **Fluxo:**
 1. Aceder ao Repositório Atlas que contém os templates de Processo.
 2. Aceder ao Catálogo de Processos e escolher um Processo para copiar.
 3. Selecionar a opção “Copiar Processo”.
 4. Definir o repositório destino para a Cópia.
 5. Selecionar “Save”.
- **Pós-condições:** No repositório destino é criado um Processo igual ao que está no Catálogo de Processos.

4.2 Apagar cópia de Template de Processo

- **Atores:** Utilizador Atlas
- **Pré-condições:** Utilizador autenticado
- **Fluxo:**
 1. Aceder ao Repositório Atlas que contém o template de Processo copiado.
 2. Aceder ao mapa de Processos.
 3. Selecionar a opção “Delete”.
- **Pós-condições:** O Processo é apagado no repositório atual, mas mantém-se disponível no Catálogo de Processos do repositório de templates.

4.3 Criar Template “Ready2Use (Fast-Setup)”

- **Atores:** Utilizador Atlas
- **Pré-condições:** Utilizador autenticado
- **Fluxo:**
 1. Aceder ao Repositório Atlas que contém o Processo.
 2. Abrir a vista BPMN do Processo.
 3. Selecionar “Save”.
 4. Carregar em “Advanced options” e “Save as Fast-Setup template”.



5. Carregar em “Save and close”

- **Pós-condições:** No repositório é criado um template “Fast-Setup” que pode ser instanciado para criar Processos.

4.4 Criar Instância de Processo a partir de um Template “Fast-Setup”

- **Atores:** Utilizador Atlas
- **Pré-condições:** Utilizador autenticado
- **Fluxo:**
 1. Aceder ao Repositório Atlas que contém o Processo.
 2. Abrir a vista BPMN do Processo.
 3. Selecionar “Save”.
 4. Carregar em “Advanced options” e “Save as Fast-Setup template”.
 5. Carregar em “Save and close”
- **Pós-condições:** No repositório é criado um template “Fast-Setup” que pode ser instanciado para criar Processos.



5 Algoritmo de Partição de BPMN

O Algoritmo de Particionamento de BPMN está dividido em três etapas principais: descoberta, partição e agregação. O objetivo é processar um modelo BPMN (Business Process Model and Notation) que contém atividades com a especificação do sistema alvo e dividi-lo em diferentes modelos BPMN funcionais, que representam os processos presentes em cada sistema distinto.

Dados de entrada:

- **Modelo BPMN:** Representado no formato XML.
- **Ficheiro de parâmetros:** Contém todas as informações necessárias para identificar e associar cada atividade ao seu sistema-alvo.

Resultados:

- **Conjunto de diagramas BPMN:** Representados por ficheiros XML, cada um correspondendo a um sistema específico. representados por ficheiros XML.

5.1 Processar Elementos do BPMN

- **Atores:** Sistema de Processamento BPMN
- **Pré-condições:**
 - O modelo BPMN está disponível em formato XML.
 - O ficheiro de parâmetros necessário está configurado.
 - O modelo BPMN é funcional e corresponde com as restrições definidas.
- **Fluxo:**
 1. O sistema lê o modelo BPMN em formato XML.
 2. Cria uma estrutura com uma lista de processos, cada um com seus elementos e fluxos associados.
 3. Identifica o elemento inicial do processo e inicia a criação de um ramo.
 4. Processa cada elemento:
 - Verifica se pertence a um novo sistema.
 - Cria um ramo em caso de gateway ou evento de fim.
 - Atualiza o fluxo de sequência com os identificadores dos elementos conectados.
- **Pós-condições:**
 - Um mapa de ramos é criado, associando identificadores únicos a cada ramo.
 - Cada elemento é corretamente atribuído ao ramo correspondente.



5.2 Agregar Ramos

- **Atores:** Sistema de Agregação BPMN
- **Pré-condições:**
 - Os ramos identificados na etapa de partição estão disponíveis no mapa de ramos.
- **Fluxo:**
 1. Itera sobre cada ramo do mapa.
 2. Identifica fluxos de entrada para determinar o ramo de origem.
 3. Verifica se os ramos podem ser fundidos com base nos sistemas envolvidos.
 4. Realiza a fusão:
 - Ramos adjacentes: Conexão direta é criada.
 - Ramos obrigatórios: Tarefas de ligação são adicionadas.
 - Ramos alternativos: Gateways baseados em eventos são adicionados.
 - Ramos opcionais: Estrutura é modificada para acomodar múltiplos fluxos.
 5. Remove ramos que foram completamente integrados.
- **Pós-condições:**
 - Um diagrama BPMN funcional e estruturado é gerado para cada sistema.
 - Os fluxos de sequência refletem as conexões corretas entre ramos.

5.3 Criar Fluxos de Sequência entre Ramos

- **Atores:** Sistema de Gestão de Fluxos
- **Pré-condições:**
 - Um ramo é identificado como concluído.
 - Um novo sistema é detetado no fluxo.
- **Fluxo:**
 1. Identifica o último elemento do ramo concluído.
 2. Cria um fluxo de sequência ligando o último elemento do ramo inicial ao primeiro elemento do novo ramo.
 3. Transmite os objetos de dados necessários entre ramos.
 4. Filtra os objetos de dados para incluir apenas aqueles acedidos pelo próximo sistema.
- **Pós-condições:**
 - Os fluxos de sequência entre ramos são estabelecidos e registados.



5.4 Registrar Estado do Algoritmo

- **Atores:** Sistema de Gerenciamento de Dados
- **Pré-condições:**
 - O algoritmo de partição está em execução.
- **Fluxo:**
 1. Atualiza:
 - Mapa de ramos com identificadores e conteúdos.
 - Lista de elementos processados.
 - Lista de elementos não explorados.
 - Identificador do último elemento processado.
 - Sistema ativo.
 2. Mantém a consistência entre as estruturas de dados durante o processamento.
 - **Pós-condições:**
 - O estado do algoritmo está corretamente registado e disponível para consulta.



6 Registo de SLAs sobre o Processo

6.1 Definir SLA para um Processo



- **Atores:** Utilizador Atlas
- **Pré-condições:** Utilizador autenticado; Processo de Negócio criado
- **Fluxo:**
 1. Abrir formulário de criação de SLA.
 2. Preencher com informação do Processo, Tarefas, tempo limite do SLA, e informação de alerta.
 3. Guardar o formulário.
- **Pós-condições:** Um novo “SLA” foi criado no Atlas associado a um Processo existente.
- Monitorizar Violação de SLA
- **Atores:** Monitor, Process Designer.
- **Pré-condições:**
 - Um SLA foi definido para um processo.
 - O processo está em execução.
- **Fluxo:**
 1. O Monitor acompanha a duração das tasks associadas ao SLA.
 2. Se uma task exceder a duração do SLA, o Monitor envia um alerta.
 3. O Process Designer recebe o alerta e toma medidas corretivas.
- **Pós-condições:**
 - A violação do SLA é registada.
 - O Process Designer é notificado da violação.



7 Registo de Business Objects Utilizados no Processo

7.1 Definir Business Objects para uma Task

- **Atores:** Utilizador Atlas.
- **Pré-condições:**
 - Um Business Process foi criado.
 - As tasks foram definidas no diagrama BPMN.
- **Fluxo:**
 1. O Process Designer define Business Objects para uma task.
 2. O Designer especifica se os Business Objects são inputs ou outputs.
 3. Os Business Objects são associados à task.
 4. O diagrama BPMN é guardado.
- **Pós-condições:**
 - Os Business Objects estão associados à task.
 - A task está pronta para usar os Business Objects durante a execução.

7.2 Passar Business Objects Entre Engines

- **Atores:** Workflow Engines (ex: edoc, Camunda), Message Queue.
- **Pré-condições:**
 - Business Objects foram definidos para tasks.
 - As tasks estão a ser executadas em diferentes Workflow Engines.
- **Fluxo:**
 1. Uma task no Engine A é concluída e envia os seus Business Objects de output para a Message Queue.
 2. A Message Queue entrega os Business Objects ao Engine B.
 3. O Engine B recebe os Business Objects e usa-os como inputs para a próxima task.
- **Pós-condições:**
 - Os Business Objects são passados com sucesso entre os engines.
 - A próxima task no Engine B pode prosseguir com os inputs recebidos.



8 Visualização de Tasks do Processo/Worklist

8.1 Ver Tasks na Worklist

- **Atores:** Utilizador, Worklist.
- **Pré-condições:**
 - Um processo está em execução.
 - As tasks foram atribuídas ao utilizador.
- **Fluxo:**
 1. O utilizador abre a Worklist no Atlas.
 2. A Worklist exibe as tasks atribuídas ao utilizador.
 3. O utilizador seleciona uma task para ver os detalhes.
- **Pós-condições:**
 - O utilizador pode ver as tasks atribuídas a si.
 - O utilizador pode começar a trabalhar nas tasks.

8.2 Automatizar Execução de Script Task

- **Atores:** Worklist, Script Task, API do ChatGPT.
 - **Pré-condições:**
 - Uma Script Task foi configurada com um prompt e Business Objects de input/output.
 - O processo está em execução.
 - **Fluxo:**
 1. A Script Task é acionada após a conclusão da task anterior.
 2. A Worklist executa automaticamente a Script Task usando a API do ChatGPT.
 3. Os Business Objects de output são atualizados com os resultados.
 - **Pós-condições:**
 - A Script Task é executada sem intervenção do utilizador.
 - Os Business Objects de output são atualizados e prontos para a próxima task.
-



9 eProcess Connector - Comunicação com eProcess

9.1 Implementar Processo em Múltiplos Engines

- **Atores:** Process Designer, Workflow Engines, Monitor.
- **Pré-condições:**
 - Um diagrama BPMN foi criado e fragmentado para diferentes engines.
- **Fluxo:**
 1. O Process Designer implementa o processo em múltiplos engines (ex: edoc, Camunda).
 2. O Designer envia os detalhes do processo para o Monitor.
 3. O Monitor acompanha a execução do processo nos diferentes engines.
- **Pós-condições:**
 - O processo foi implementado em todos os engines especificados.
 - O Monitor está pronto para acompanhar o processo.

9.2 Enviar Atualização de Estado da Task para o Monitor

- **Atores:** Workflow Engine, Monitor.
 - **Pré-condições:**
 - Uma task está a ser executada num Workflow Engine.
 - **Fluxo:**
 1. O Workflow Engine envia um evento de "Início da Task" para o Monitor.
 2. A task é concluída, e o Engine envia um evento de "Fim da Task" para o Monitor.
 3. O Monitor atualiza o estado da task.
 - **Pós-condições:**
 - O Monitor tem o estado mais recente da task.
 - O estado da task está disponível para relatórios e análise.
-



10 Visualização de Estatísticas e Mining do Processo

10.1 Ver Estatísticas do Processo

- **Atores:** Process Designer, Monitor.
- **Pré-condições:**
 - Um processo foi executado.
 - As estatísticas foram recolhidas pelo Monitor.
- **Fluxo:**
 1. O Process Designer solicita as estatísticas do processo ao Monitor.
 2. O Monitor fornece estatísticas como durações das tasks e violações de SLA.
 3. O Designer visualiza as estatísticas na interface do Atlas.
- **Pós-condições:**
 - O Designer pode analisar o desempenho do processo.
 - As estatísticas estão disponíveis para análise adicional.

10.2 Realizar Process Mining

- **Atores:** Process Designer, Monitor.
- **Pré-condições:**
 - Um processo foi executado.
 - Os dados de mining foram recolhidos pelo Monitor.
- **Fluxo:**
 1. O Process Designer solicita os dados de mining do processo ao Monitor.
 2. O Monitor fornece dados de mining, incluindo sequências de tasks e durações.
 3. O Designer analisa os dados de mining para identificar gargalos.
- **Pós-condições:**
 - O Designer pode identificar ineficiências no processo.
 - Os dados de mining estão disponíveis para otimização do processo.



11 Integração com o GPT

11.1 Execução de Tarefas de Script com Prompts Textuais

- **Atores:** Designer de Processos, Sistema, API ChatGPT.
- **Pré-condições:**
 - Um processo BPMN com uma Tarefa de Script está definido.
 - Um prompt textual é configurado como propriedade da Tarefa de Negócio.
- **Fluxo:**
 1. O Designer de Processos cria uma Tarefa de Script no processo BPMN.
 2. A Tarefa de Script é configurada com um prompt textual.
 3. O Batch é executado, chamando a API do ChatGPT para processar o prompt.
 4. A saída (XML atualizado) é guardada no Objeto de Negócio de saída.
- **Pós-condições:**
 - O Objeto de Negócio é atualizado com a informação processada.
 - O registo da execução do Batch é armazenado no objeto da Tarefa de Script.

11.2 Manuseamento de Entrada

11.2.1 Entrada é um Objeto de Negócio com um Ficheiro ou Imagem

Atores: Designer de Processos, Sistema.

- **Pré-condições:**
 - Existe um Objeto de Negócio com uma propriedade de ficheiro ou imagem.
 - A Tarefa de Script está configurada para lidar com este tipo de entrada.
- **Fluxo:**
 1. O Designer de Processos seleciona o Objeto de Negócio como entrada.
 2. O sistema exporta o Objeto de Negócio e o ficheiro/imagem associado.



3. O Batch processa a entrada utilizando a API do ChatGPT.
 4. A saída é guardada no Objeto de Negócio de saída especificado.
- **Pós-condições:**
 - O Objeto de Negócio de saída contém o ficheiro ou imagem atualizado.
 - O registo da execução do Batch é armazenado.

11.2.2 Entrada é um ficheiro ATLAS XML

- **Atores:** Designer de Processos, Sistema.
- **Pré-condições:**
 - Um ficheiro ATLAS XML está disponível como entrada.
 - A Tarefa de Script está configurada para lidar com ficheiros ATLAS XML.
- **Fluxo:**
 1. O Designer de Processos seleciona o ficheiro ATLAS XML como entrada.
 2. O sistema exporta o ficheiro XML para o diretório de entrada.
 3. O Batch processa o ficheiro XML utilizando a API do ChatGPT.
 4. A saída é guardada no Objeto de Negócio de saída especificado.
- **Pós-condições:**
 - O Objeto de Negócio de saída contém o ficheiro XML atualizado.
 - O registo da execução do Batch é armazenado.

11.3 Integração com a API do ChatGPT

- **Atores:** Sistema, API ChatGPT.
- **Pré-condições:**
 - Uma chave de API válida e um endpoint do ChatGPT estão configurados.
 - A Tarefa de Script está configurada com um prompt textual.
- **Fluxo:**
 1. O Batch envia o prompt e os dados de entrada para a API do ChatGPT.
 2. A API processa o prompt e devolve o resultado.
 3. O sistema guarda a resposta da API no Objeto de Negócio de saída.
- **Pós-condições:**
 - O Objeto de Negócio de saída é atualizado com a resposta da API.
 - O registo da execução do Batch inclui os detalhes da chamada à API.



11.4 Configuração do Batch

- **Atores:** Designer de Processos, Sistema.
- **Pré-condições:**
 - Um processo BPMN é guardado como um "Modelo de Lista de Trabalho".
 - A Tarefa de Script está configurada com as propriedades necessárias.
- **Fluxo:**
 1. O Designer de Processos guarda o BPMN como um Modelo de Lista de Trabalho.
 2. O sistema cria automaticamente um Batch com os parâmetros necessários.
 3. A configuração do Batch é armazenada no objeto da Tarefa de Script.
- **Pós-condições:**
 - O Batch está pronto para execução.

11.5 Análise de BPMN

- **Atores:** Sistema.
- **Pré-condições:**
 - Um ficheiro BPMN com uma Tarefa de Script é carregado.
 - A Tarefa de Script está configurada com um prompt e um Objeto de Negócio.
- **Fluxo:**
 1. O sistema analisa o ficheiro BPMN.
 2. Quando encontra uma Tarefa de Script, o sistema cria um Batch com os parâmetros especificados.
 3. O Batch é configurado com o prompt e o Objeto de Negócio.
- **Pós-condições:**
 - O Batch é criado e está pronto para execução.
 - As propriedades da Tarefa de Script são validadas.

11.6 Especificação do Script

- **Atores:** Sistema, Designer de Processos.
- **Pré-condições:**
 - O script está configurado com os parâmetros necessários (param.txt, inputDir, outputDir).
 - O ficheiro de parâmetros (param.txt) está corretamente estruturado.



- **Fluxo:**
 1. O sistema lê o ficheiro de parâmetros e valida a sua estrutura.
 2. O script processa os dados de entrada utilizando a API do ChatGPT.
 3. A saída é guardada no diretório de saída especificado.
- **Pós-condições:**
 - Os ficheiros de saída são gerados no diretório de saída.
 - O registo da execução do script é armazenado.

11.7 Manuseamento da Saída

11.7.1 Formato de Saída: Ficheiro ATLAS

- **Atores:** Sistema.
- **Pré-condições:**
 - A propriedade "Formato de Saída" está definida como "Ficheiro ATLAS".
 - Os dados de entrada são válidos e processáveis.
- **Fluxo:**
 1. O script processa os dados de entrada utilizando a API do ChatGPT.
 2. A saída é guardada como um ficheiro ATLAS XML.
- **Pós-condições:**
 - O ficheiro ATLAS XML é gerado e armazenado no diretório de saída.

11.7.2 Formato de Saída: Texto

- **Atores:** Sistema.
- **Pré-condições:**
 - A propriedade "Formato de Saída" está definida como "Texto".
 - Os dados de entrada são válidos e processáveis.
- **Fluxo:**
 1. O script processa os dados de entrada utilizando a API do ChatGPT.
 2. A saída é guardada na propriedade "Texto de Saída" do Objeto de Negócio.
- **Pós-condições:**
 - A propriedade "Texto de Saída" é preenchida com a resposta da API.

11.8 Estrutura do Batch – Configuração ATLAS

- **Atores:** Sistema.



- **Pré-condições:**
 - O Batch está configurado com trabalhos de exportação, execução e importação.
 - Os dados de entrada são válidos e processáveis.
- **Fluxo:**
 1. O trabalho de exportação exporta os dados de entrada para o diretório de entrada.
 2. O trabalho de execução processa os dados utilizando a API do ChatGPT.
 3. O trabalho de importação importa os dados processados para o Objeto de Negócio de saída.
- **Pós-condições:**
 - O Objeto de Negócio de saída é atualizado com os dados processados.
 - O registo da execução do Batch é armazenado.

11.9 Configuração do Trabalho com Base no Tipo de Entrada

- **Atores:** Sistema.
- **Pré-condições:**
 - A entrada é um Objeto de Negócio com uma propriedade de ficheiro ou imagem.
 - O Batch está configurado com dois trabalhos de exportação, um de execução e um de importação.
- **Fluxo:**
 1. Os trabalhos de exportação exportam o Objeto de Negócio e o ficheiro/imagem.
 2. O trabalho de execução processa os dados utilizando a API do ChatGPT.
 3. O trabalho de importação importa os dados processados para o Objeto de Negócio de saída.
- **Pós-condições:**
 - O Objeto de Negócio de saída é atualizado com o ficheiro/imagem processado.

11.10 Manuseamento de Erros e Registos

- **Atores:** Sistema.
- **Pré-condições:**



- O ficheiro de parâmetros (param.txt) está corretamente estruturado.
- O sistema está configurado para registar erros e detalhes da execução.
- **Fluxo:**
 1. O sistema valida o ficheiro de parâmetros e os dados de entrada.
 2. Se ocorrer um erro, o sistema regista o erro e interrompe o processo.
 3. Se a execução for bem-sucedida, o sistema regista os detalhes da execução.
- **Pós-condições:**
 - Os erros são registados e comunicados.
 - As execuções bem-sucedidas são registadas para auditoria.