



MATHEUS ROCHA SILVA PIMENTA

Modelagem de processos de possíveis clientes em uma empresa de desenvolvimento de software

**LAVRAS – MG
2023**

MATHEUS ROCHA SILVA PIMENTA

Modelagem de processos de possíveis clientes em uma empresa de desenvolvimento de software

Relatório de Estágio Supervisionado apresentado à Universidade Federal de Lavras como parte das exigências do curso de Sistemas de Informação, para obtenção do título de Bacharel.

Profª. Renata Teles Moreira
Orientadora

LAVRAS - MG
2023

MATHEUS ROCHA SILVA PIMENTA

Modelagem de processos de possíveis clientes em uma empresa de desenvolvimento de software

Relatório de Estágio Supervisionado apresentado à Universidade Federal de Lavras como parte das exigências do curso de Sistemas de Informação, para obtenção do título de Bacharel.

Aprovado em 13 de Julho de 2023

Renata Teles Moreira
Paulo Afonso Parreira Júnior
José Monserrat Neto

DCC/UFLA
DCC/UFLA
DCC/UFLA

Profa. Renata Teles Moreira
Orientadora

**LAVRAS - MG
2023**

AGRADECIMENTOS

Agradeço primeiramente à Universidade Federal de Lavras, pelo suporte e oportunidade de realização deste trabalho. Meus sinceros agradecimentos aos professores, em especial à Prof. Renata Teles Moreira, pela orientação precisa, dedicação e valiosas contribuições ao longo do processo de construção do TCC. Agradeço também aos meus colegas de turma, pela troca de conhecimentos e amizades formadas. Agradeço aos meus amigos e familiares, pelo incentivo, compreensão e apoio incondicional durante toda a trajetória acadêmica. Sem a presença e suporte de todos vocês, este trabalho não teria sido possível.

LISTA DE FIGURAS

Figura 1. Principais serviços ofertados pela empresa	11
Figura 2. Estrutura da empresa	13
Figura 3. Estrutura da Diretoria de Delivery	14
Figura 4. Processo de Captação de Novos Clientes	16
Figura 5. Exemplo Molde-Registro	18
Figura 6. Referência comum	21
Figura 7. Referência embutida	22
Figura 8. Classe e objeto	23
Figura 9. Pacotes	24
Figura 10. Diagrama de Versão de Processo	25
Figura 11. Tipos de Eventos	27
Figura 12. Eventos interruptivos e não-interruptivos	30
Figura 13. <i>Gateway</i> Exclusivo Divergente	33
Figura 14. <i>Gateway</i> Inclusivo Divergente	34
Figura 15. <i>Gateway</i> Paralelo Divergente e Convergente	34
Figura 16. <i>Gateway</i> Complexo	35
Figura 17. <i>Gateway</i> Baseado em Evento	35
Figura 18. Página de criação de classe	42
Figura 19. Diagrama de processo	43
Figura 20. Visualização de <i>chat</i>	47

LISTA DE QUADROS

Quadro 1. Menu de elementos	26
Quadro 2. Ferramentas de elemento	27
Quadro 3. Eventos de Início	27
Quadro 4. Evento final	28
Quadro 5. Eventos intermediários	28
Quadro 6. Tipos de Atividades	32

SUMÁRIO

1. INTRODUÇÃO.....	8
2. SOBRE A ORGANIZAÇÃO.....	10
2.1. Empresa Alpha.....	10
2.2. Processo Organizacional.....	12
2.3. Produto.....	17
2.3.1. Modelagem de dados.....	17
2.3.1.1. Classe.....	18
2.3.1.2. Objetos.....	23
2.3.1.3. Pacotes.....	23
2.3.2. Modelagem de processos.....	24
2.3.2.1. Elementos de processo.....	26
2.3.2.1.1. Eventos.....	26
2.3.2.1.2. Atividades.....	31
2.3.2.1.3. Gateways.....	33
2.3.2.1.4. Conexão.....	36
2.3.2.2. Versão de processo.....	36
2.3.2.2.1. Campo.....	36
2.3.2.2.2. Métodos.....	37
2.3.2.2.3. Papéis.....	37
3. FUNDAMENTAÇÃO TEÓRICA.....	38
3.1. BPM.....	38
3.2. BPMN.....	39
3.3. Automação de Processos.....	40
4. ATIVIDADES DESENVOLVIDAS.....	41
4.1. Treinamento.....	41
4.2. Mapeamento de classes e processos.....	41
4.3. Automação de processos.....	43
4.4. Portal de Relacionamentos.....	44
4.5. Relatórios Analytics.....	45
4.6. Chatbot Whatsapp.....	45
4.7. Integrações.....	47
5. CONSIDERAÇÕES FINAIS.....	49
REFERÊNCIAS.....	50

1. INTRODUÇÃO

Conforme o BPM CBOOK, o *Business Process Management (BPM)* é uma disciplina que promove a transformação de ações isoladas em projetos de melhoria que contemplam toda uma organização que o implementa, por meio de uma cadeia produtiva processual. Sua adoção promove a automação de processos produtivos e o mapeamento de processos a partir de um padrão utilizado internacionalmente - conhecido por *Business Process Modeling and Notation (BPMN)* - visando, dentre seus objetivos, eliminar a ocorrência de falhas na transferência de dados, informações e responsabilidades entre indivíduos ou setores e gerar ganhos em produtividade e desempenho.

Uma organização orientada por processos é uma organização consciente de sua missão, que, conhecendo sua sequência de trabalho, analisa e busca aperfeiçoar seu desempenho. Isto posto, o *BPM* é uma iniciativa de modernização que possibilita o avanço da qualidade e da melhoria contínua dos processos de maneira integrada.

A Empresa Alpha é uma organização do ramo da Tecnologia da Informação, com sede em Belo Horizonte, e com modelo de negócio Software como Serviço (SaaS do inglês Software as a Service). Ela desenvolve e oferece uma plataforma digital *all-in-one* orientada a processos e melhoria contínua. A plataforma concentra várias soluções dentro da mesma tecnologia, tais como *CRM (Customer Relationship Management)*, *Billing (Faturamento)*, *E-commerce*, *ECM (Enterprise Content Management)*, Recursos Humanos, *BPM* e outros, buscando unificar todos os processos financeiros, de marketing, recursos humanos, administrativos, vendas e demais áreas administrativas.

É dentro desse contexto que as atividades do estagiário se encontram. Sua atuação se deu na área técnica do setor Comercial, atuando no desenvolvimento de provas de conceito para os clientes potenciais. A partir dos requisitos de um processo ou serviço do cliente potencial, as ações desempenhadas buscavam automatizá-lo dentro da plataforma por meio de diferentes funcionalidades e facilidades. Assim, o estagiário atuou no mapeamento e automação de processos, criação de classes, atributos e métodos e na aplicação de regras de negócio por meio do desenvolvimento de *scripts* utilizando *Javascript*.

Além deste capítulo introdutório, este relatório de estágio está organizado em cinco capítulos, dos quais o Capítulo 2 apresenta a organização onde o estagiário atuou; o Capítulo 3 aborda a fundamentação teórica dos assuntos os quais o estagiário vivenciou em sua rotina de trabalho; o Capítulo 4, que trata das atividades desenvolvidas; e, por fim, o Capítulo 5, que

apresenta as considerações finais acerca do aprendizado adquirido e das aplicações práticas da teoria aprendida no curso de Sistemas de Informação sobre as atividades desempenhadas durante o estágio.

2. SOBRE A ORGANIZAÇÃO

Este Capítulo discorre sobre a organização ao qual os trabalhos foram realizados.

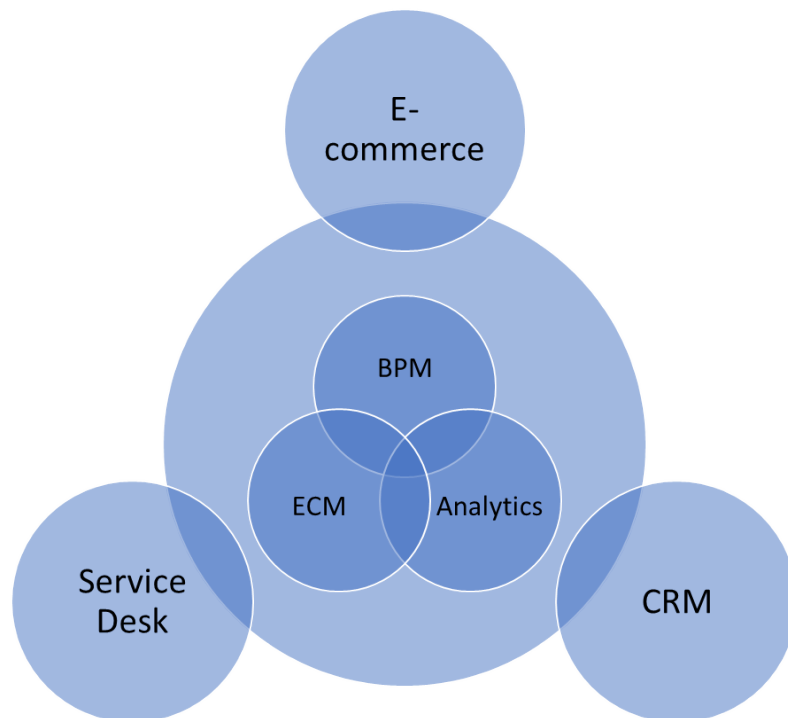
2.1. Empresa Alpha

A Empresa Alpha oferece uma plataforma digital *all-in-one* contendo três pilares gerenciais que formam o *core* de sua solução:

- Gestão por processos (BPM) - é uma disciplina gerencial que propõe a otimização dos processos de um negócio, de ponta a ponta, para gerar mais valor ao cliente e aprimorar o desempenho da empresa rumo à conquista de seus objetivos estratégicos. Assim, permite-se modelar, acompanhar e gerenciar toda a cadeia de processos de negócio da companhia de forma automatizada e integrada;
- Gestão de conteúdo organizacional, por meio do ECM (*Enterprise Content Management*) - é um sistema estratégico de coleta e organização de informações que gerencia todo o ciclo de vida de diferentes dados do negócio. Graças a esse tipo de tecnologia, é possível garantir mais produtividade, já que os dados são coletados de acordo com as especificidades de cada público. Na plataforma, é possível modelar e automatizar vários tipos de dados do seu negócio; e
- Gestão à vista, com o *Analytics* - é a ferramenta tecnológica utilizada para destrinchar os dados armazenados, tanto pelo ECM quanto pelo BPM, visando transformá-los em gráficos e modelos visuais, possibilitando a análise e comparação em tempo real e a extração de *insights* para apoiar as tomadas de decisões da empresa.

Além dos pilares acima, a empresa também oferta vários módulos adicionais incluindo *Customer Relationship Management* (CRM), Central de serviços compartilhados (Service Desk), E-commerce, dentre outros. A Figura 1 ilustra os serviços ofertados pela empresa.

Figura 1. Principais serviços ofertados pela empresa



Fonte: Criado pelo autor

Dentro da empresa, a atuação do estagiário se deu na equipe de Delivery, considerado o departamento Comercial. Esse departamento é subdividido em Mundo Privado, com foco em prospectar clientes do setor privado, Mundo Público, com foco em prospectar clientes do setor público, Educação, com foco em prospectar clientes do ramo da Educação, e Global, com foco em prospectar clientes internacionais. Os trabalhos realizados permeiam todos esses subsetores, atuando em provas de conceito onde quer que houvesse necessidade. Entretanto, o estagiário respondia para a liderança do Mundo Privado, formada por uma equipe de oito pessoas.

O principal papel do estagiário foi o desenvolvimento de provas de conceito, ou seja, modelagem e automação de dados e processos a partir de requisitos de serviços/processos reais dos clientes potenciais desenvolvidos dentro da plataforma. Assim, o estagiário atuava em uma gama de serviços de diversas empresas com diferentes segmentos de mercado. Essa

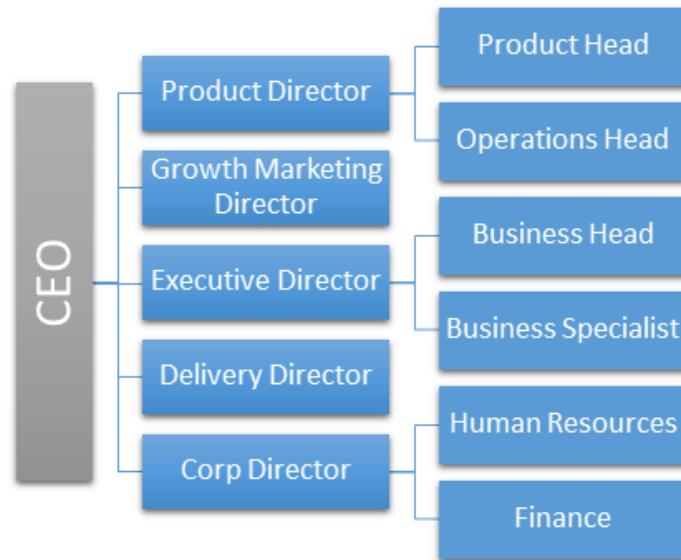
atuação não se limitava somente ao *core* da solução. Também utilizava-se dos outros módulos da ferramenta para criar uma prova de conceito robusta, próxima da realidade do cliente e que demonstrasse o potencial de modelagem e automação da solução.

2.2. Processo Organizacional

A empresa se encontra no mercado de tecnologia da informação há 18 anos. Devido ao seu foco na transformação digital ao longo dos anos, a organização de sua estrutura se tornou horizontalizada com equipes multidisciplinares, sem departamentalização das diversas áreas da empresa. Com uma estrutura horizontal, o trabalho está organizado ao redor de processos de negócio, tendo sua gestão desempenhada por equipes, ao invés de indivíduos (OSTROFF; SMITH, 1992, p. 149). Esse modelo de negócios permite uma série de benefícios, sendo eles: profissionais de áreas distintas trabalhando mais próximos uns dos outros; facilidade no *handoff* de atividades e na troca de informação; e maior proximidade entre direção e os times, visto que a direção está presente no dia a dia das operações.

A estrutura da empresa está apresentada na Figura 2 em função de seus macroprocessos. A interdisciplinaridade das funções fica evidente, por exemplo, nas equipes de Desenvolvimento de Software, que estão alocadas junto às diretorias de Product, Growth Marketing, Executive e Delivery e nas equipes de Design que estão alocados junto às diretorias de Product, Growth Marketing e Delivery.

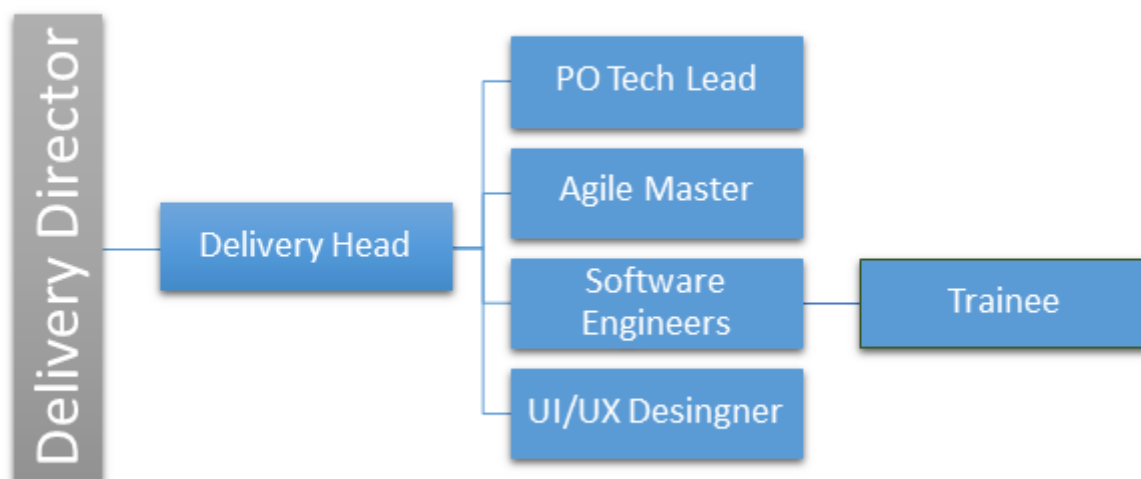
Figura 2. Estrutura da empresa



Fonte: Criado pelo autor

O estagiário atuou nos processos vinculados à Diretoria de Delivery, alocado na equipe de Software Engineering no macroprocesso de captação de novos clientes (Figura 3).

Figura 3. Estrutura da Diretoria de Delivery



Fonte: Criado pelo autor

Os processos da empresa estão organizados no desenvolvimento, sustentação e manutenção da sua principal solução. Os processos buscam entregar uma solução completa e robusta para atender às necessidades e expectativas dos clientes. O estagiário atuou especificamente no processo de Captação de novos clientes junto ao time técnico comercial. Conforme Figura 4, apresenta-se o fluxograma que ilustra as atividades executadas nesse processo.

O processo pode ser iniciado pelo próprio time Comercial ao buscar ativamente oportunidades no mercado, contatando potenciais clientes, ou por empresas que buscam conhecer a plataforma e têm interesse em implementá-la em seus processos. Quando o contato é bem sucedido, o time Comercial realiza várias reuniões para entender a necessidade do cliente, alinhar expectativas e apresentar a solução. Dessas reuniões, pode surgir a necessidade de demonstrar o potencial da solução no contexto do cliente e, então, é oferecido que seja feita uma prova de conceito.

O objetivo da prova de conceito é testar a viabilidade técnica da solução com o desenvolvimento resumido dos requisitos apresentados.

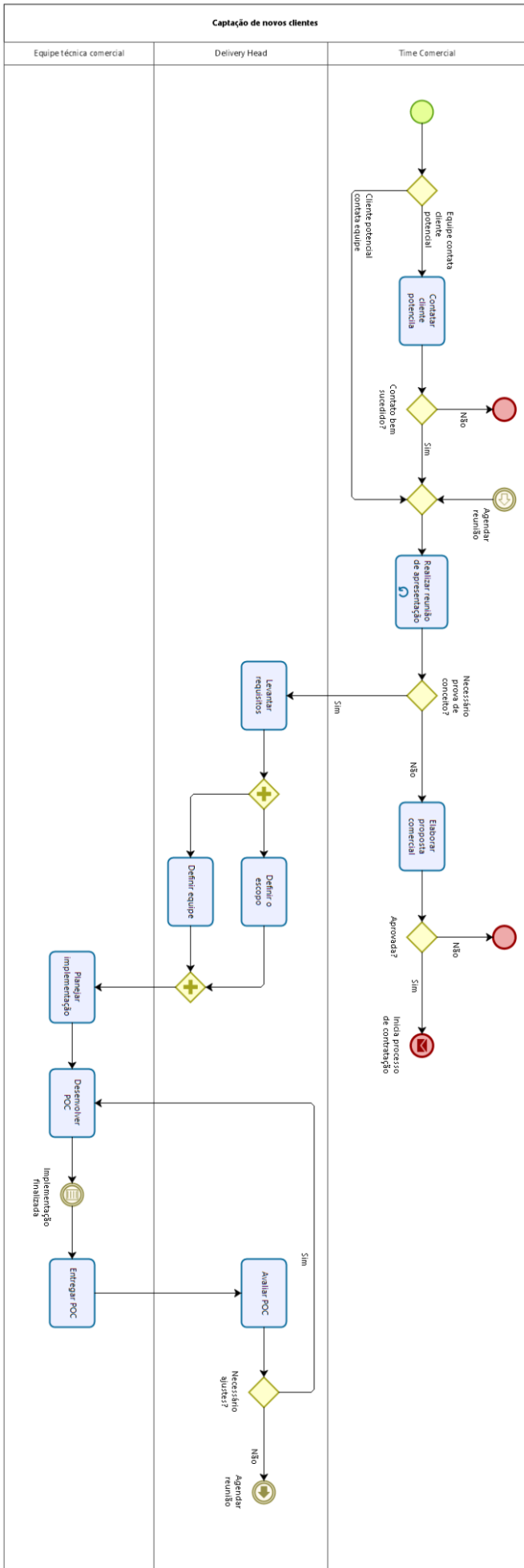
Caso a prova de conceito seja necessária, o *Delivery Head*, também membro da equipe Comercial e líder da equipe técnica Comercial, realiza o levantamento de requisitos junto ao responsável pelo processo do cliente, podendo ser processos ou descrições de serviços. Dessa forma, o *Delivery Head* realiza a definição do escopo limitando os requisitos a serem implementados de acordo com as funcionalidades que possivelmente serão mais interessantes para a empresa. Outro motivo para a delimitação do escopo está no fato da equipe técnica Comercial trabalhar em várias provas de conceito simultaneamente. Assim, o esforço de trabalho deve ser equilibrado de forma a produzir com qualidade e entregar as provas de conceito dentro do prazo acordado com a empresa. Levando em consideração a carga de trabalho dos membros da equipe, os responsáveis pelo desenvolvimento da prova de conceito são definidos.

Por outro lado, caso a prova de conceito não seja necessária, a proposta comercial é elaborada e apresentada à empresa e, sendo aprovada, inicia-se o processo de contratação.

Com o escopo e equipe definidos, a equipe técnica Comercial inicia o planejamento da implementação, definindo classes, seus atributos e métodos, as funcionalidades nativas que podem ser utilizadas e o protótipo do processo.

Após o planejamento, é iniciado o desenvolvimento da prova de conceito dentro da plataforma. O prazo para implementação varia de acordo com a complexidade e agenda do cliente. Ao finalizá-la, a equipe entrega a prova de conceito ao *Delivery Head* e, caso sejam identificadas necessidades de mudanças, a equipe técnica fica encarregada de implementá-las. Com o fim da implementação, o *Delivery Head* apresenta a prova de conceito à empresa. Não é incomum uma prova de conceito gerar outras provas, buscando demonstrar outras funcionalidades da solução. Com isso, as atividades relacionadas à prova de conceito podem ser desempenhadas quantas vezes forem necessárias antes de seguir para elaboração da proposta comercial.

Figura 4. Processo de Captação de novos clientes



Fonte: Criado pelo autor

2.3. Produto

A plataforma *all-in-one* trabalha com duas definições: a modelagem de dados e a modelagem de processos.

2.3.1. Modelagem de dados

Primeiramente, é importante explicar que dados são uma coleção de fatos que podem incluir números, imagens, vídeos, palavras, medidas, observações e muito mais. Os dados estão em toda parte e são uma fonte valiosa de informação para as empresas.

Na modelagem de dados, utilizamos dos fatos ou atributos que descrevem o dado para moldá-los, criando representações computacionais. Assim, a modelagem envolve uma análise para compreender o elemento que está sendo observado e, então, representá-lo computacionalmente.

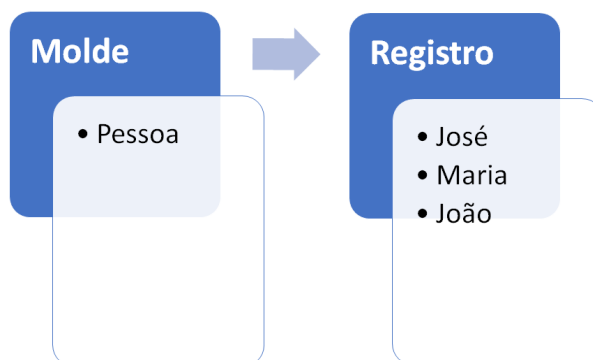
O objetivo de se modelar dados está no fato de podermos representar suas estruturas e comportamentos que ocorrem durante a rotina do negócio. Isso permite a digitalização das operações e possibilita a transformação digital do negócio. Consequentemente, quanto mais preciso, reflexivo e detalhista for este processo, mais eficiente se tornam as operações e menos frequente se torna o retrabalho.

A modelagem de dados na solução é baseada no Paradigma Molde-Registro. Ele define a representação de um padrão a ser seguido na construção do dado dentro da plataforma. Nela, temos dois contextos de criação de dados:

- O dado estruturante, representando o Molde no paradigma, que é o modelo final do dado que queremos; e
- O dado estruturado, representando o Registro no paradigma, que é o registro em si do dado estruturante.

Conforme é mostrado no exemplo da Figura 5, podemos criar o molde “Pessoa” - um dado estruturante - e depois efetuar o registro de pessoas - este sendo um dado estruturado. De um lado, se cria a entidade modeladora, onde está configurada a forma como registrar os dados, e, do outro lado, criam-se os registros desse modelador.

Figura 5. Exemplo Molde-Registro



Fonte: Criado pelo autor

A solução segue esse paradigma, mas o molde é chamado de Classe, e os registros de Objetos. Seguindo o exemplo da Figura 5, a Classe “Pessoa” se refere ao modelador de pessoas, e o Objeto “Pessoa” se refere aos registros de uma pessoa, por exemplo José, Maria e João.

2.3.1.1. Classe

Classes são as definições da estrutura dos Objetos (informações) armazenados. Toda classe criada deve ter as seguintes informações especificadas:

- Nome da classe;
- Identificador da classe, utilizado para referenciar a classe em *scripts* automatizados;
- Tipo da classe, podendo ser tipo “Padrão”, utilizado para criar classes que definem o conteúdo e comportamento de Objetos, ou tipo “Interface”, utilizado para criar definições para outras classes;
- Campo Segue as interfaces, tipo booleano, utilizado para selecionar as Interfaces implementadas pela classe atual;
- Campo Objetos buscáveis, tipo booleano, utilizado para definir a possibilidade de se buscar os objetos da classe. Geralmente é desabilitado em Classes criadas especificamente para definir formulários embutidos reaproveitáveis;

- Campos que definem as informações armazenadas pelos objetos da classe;
- Métodos que definem as operações (comportamento) que podem ser realizadas sobre os objetos das classes.
- Expiração do histórico, o tempo em segundos que os itens de histórico gerados para objetos da classe devem ser armazenados. Por padrão, esse tempo é de 31.536.000 segundos, ou seja, um ano;
- Ajuda para edição, o texto de ajuda exibido ao se criar/editar os objetos da classe.

A entidade Classe possui quatro características principais: Identificação, Conteúdo, Relacionamento e Comportamento.

Identificação

Toda classe é identificada a partir de seu nome e identificador. A diferença entre eles é que no Nome, geralmente se atribui um substantivo que resume o objetivo daquela entidade. Uma boa prática adotada é sempre evitar o uso de verbos como Nome. Por exemplo, a classe “Pessoa” possui todas as informações necessárias para uma pessoa, a classe Contrato, para um contrato, e assim por diante.

Já o identificador é a forma como o sistema identifica a classe e deve ser único. Por conta disso, o identificador possui restrições como ser alfanumérico, começando com uma letra ou com “_”. É recomendado o uso do padrão *PascalCase* para os identificadores.

Conteúdo

O conteúdo da classe é chamado de Campos. Os campos são os atributos de uma classe. Assim, sempre que se cria uma classe, é importante preenchê-la com os seus respectivos campos. Esses campos podem ser do tipo texto, número inteiro, decimal, arquivo, booleano, coordenadas geográficas, ou pode ser de um tipo especial, chamado Referência. Esse tipo Referência se refere a característica Relacionamento.

Os campos de uma Classe especificam como deverá ser exibido o Formulário de edição e visualização dos objetos. Assim, os campos possuem a seguinte estrutura padrão:

- Nome - rótulo do campo exibido no formulário do objeto;
- Identificador - utilizado para referenciar o campo em scripts automatizados. Definido automaticamente pela plataforma, mas podendo alterá-lo;

- Relevância com que o campo deve ser indexado para pesquisa e exibido nos cards, podendo ser Identidade, que define o campo como prioritário, Destaque, que o define como secundário, Comum, caracterizando-o como sem relevância, e Avançado, que oculta o campo inicialmente;
- Tipo do valor armazenado pelo campo;
- Valor padrão - objetos da classe que terão o campo com este valor caso não seja alterado;
- Embutido - que define um relacionamento do tipo Referência Embutida;
- Obrigatório - que define preenchimento obrigatório no momento da criação do formulário;
- Somente leitura - o valor do campo não pode ser alterado via formulário;
- Múltiplo - o campo permite armazenar múltiplos itens;
- Oculto - o campo não será visível nos formulários.

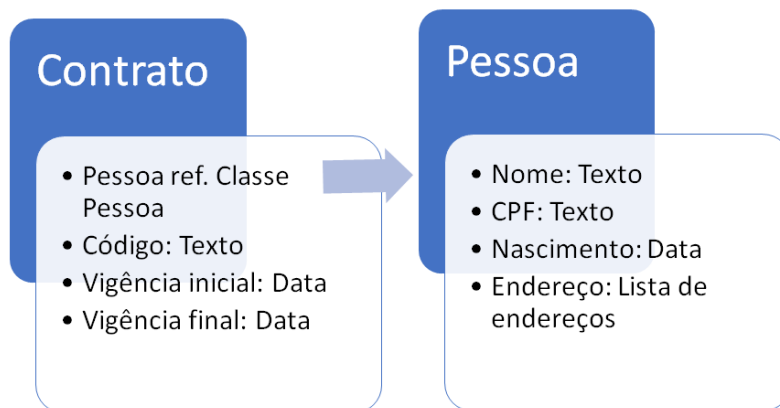
Relacionamento

É possível criar relacionamentos entre classes. Para isso, deve-se dizer que o campo é do tipo referência. Assim, criam-se vínculos entre as entidades agregando a capacidade de se reutilizar as classes e seus campos. As referências otimizam o tempo ao reaproveitar estruturas já modeladas, ao estruturar dados de forma consistente e não repetitiva, e ao padronizar os dados. Por exemplo, a entidade Contrato pode ter um vínculo com a entidade Pessoa. Então, quando registrar um contrato, poderá selecionar a pessoa à qual aquele contrato está vinculado, ao invés de recadastrar todos os dados da pessoa na classe Contrato. Existem dois tipos de relacionamentos: a Referência Comum e a Referência Embutida.

A referência comum é utilizada para representar referências entre objetos que existem de forma independente. Assim, a necessidade de recriar os campos semelhantes nas duas classes não seria necessária, porque essas informações já estariam contidas no registro de uma das classes.

Conforme exemplo demonstrado na Figura 6, o registro de contrato faz uma referência ao registro de pessoa. Com isso, aproveita-se os atributos de pessoa para a classe contrato.

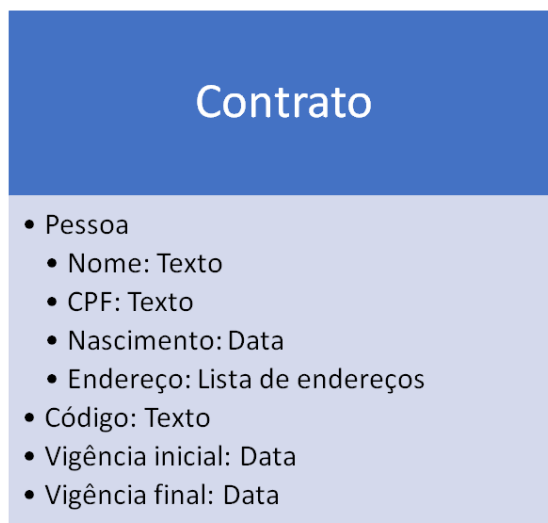
Figura 6. Referência comum



Fonte: Criado pelo autor

A referência Embutida é utilizada quando o objeto referenciado é parte integrante do objeto origem. Ou seja, o objeto referenciado não existe sem a sua origem, de tal forma que, quando é utilizado, o formulário do objeto referenciado é exibido dentro do objeto origem. Assim, passa-se a existir apenas um registro na plataforma (Figura 7).

Figura 7. Referência embutida



Fonte: Criado pelo autor

Comportamento

As classes possuem o que chamamos de métodos, que nada mais são do que os comportamentos ou ações que uma classe pode executar. Essas ações podem ser programadas para executar regras específicas, das mais simples até as mais complexas. Por exemplo, a classe Contrato pode ter uma ação chamada “Encerrar contrato” que, ao ser executada, obriga o usuário a indicar a data de encerramento, atualizando assim o campo data fim da vigência.

Os métodos são compostos pelas seguintes informações:

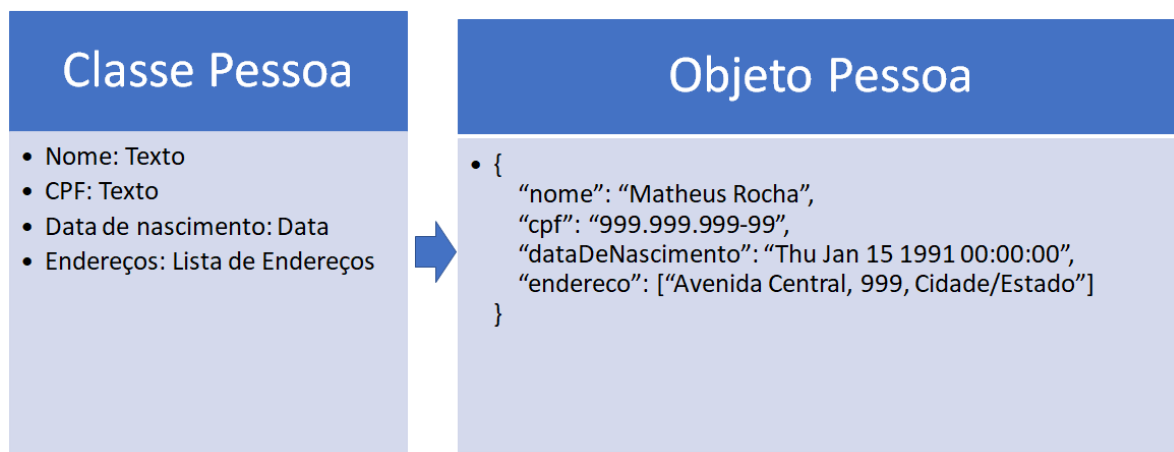
- Nome do método;
- Identificador, utilizado para referenciar o método em *scripts* automatizados;
- *Scripts*, campo onde o código deve ser escrito em *JavaScript* - pode ser criado mais de um *script* para um método, sendo executados na ordem de criação; e
- Nível de acesso, que define as permissões de acesso para chamar o método.

2.3.1.2. Objetos

Uma vez que a classe está criada e configurada, os seus objetos podem ser registrados. Para o usuário final, esse dado é apresentado como um formulário que ele preenche e salva na plataforma. Porém, para o sistema, esse dado é representado utilizando o formato JSON.

O objeto é considerado como a entidade básica da solução. Cada objeto criado pertence a uma Classe, conforme suas definições de campos e métodos, e é organizado dentro do Pacote dessa Classe. Eles representam os dados armazenados de forma estruturada. Pode ser qualquer tipo de registro, tal como um cliente, um produto, um contrato, uma operação financeira, dentre outras.

Figura 8. Classe e objeto

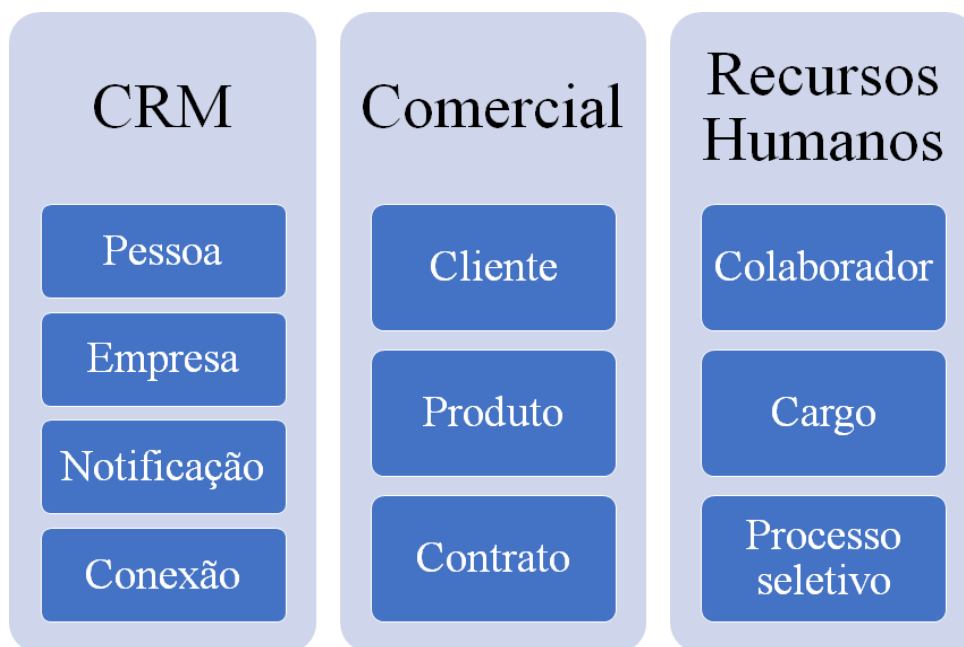


Fonte: Criado pelo autor

2.3.1.3. Pacotes

As classes e seus objetos podem ser agrupados por meio da estrutura chamada Pacote. O Pacote é responsável por agrupar classes comuns organizando suas informações.

Figura 9. Pacotes



Fonte: Criado pelo autor

Essa entidade é criada informando apenas seu nome e um identificador. No campo Nome se insere um texto que representa um denominador comum para o conjunto de classes agrupadas. Esse denominador comum é chamado de domínio. Por exemplo, as classes Pessoa, Empresa, Notificação e Conexão estão no domínio CRM (*Customer Relationship Management*), ou seja, todas envolvidas com a gestão de clientes.

O identificador do pacote é utilizado como caminho para acessar as classes pertencentes a ele e possui regras semelhantes ao identificador da classe, sendo também alfanumérico e gerado utilizando *CammelCase*.

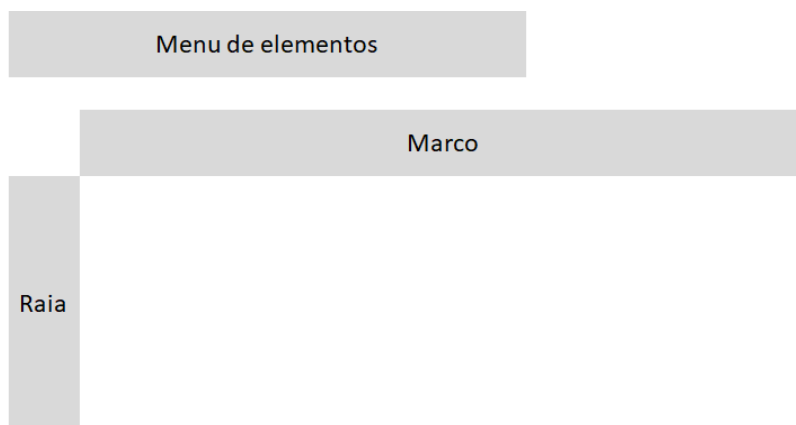
2.3.2. Modelagem de processos

A representação de processos de negócios é feita através da notação BPMN. *Business Process Model and Notation* (BPMN) é uma notação gráfica que tem como objetivo prover uma gramática de símbolos para mapear, de maneira padrão, todos os processos de negócio de uma organização.

O processo também é uma classe, chamada de Processo, e, portanto, é possível criar objetos a partir dela. Os objetos dessa classe representam os registros do processo, tais como seu nome, identificador, pacote e a referência para a classe de diagramação do processo, chamada Versão de Processo. Esta classe representa as versões do processo, sendo que um processo pode possuir várias versões distintas vinculadas a ele. Entretanto, apenas uma versão - normalmente a versão mais recente - é permitida ser referenciada para cada objeto da classe Processo.

O mapeamento e automação são feitos na classe Versão de Processo, classe de diagramação (Figura 10), e é enumerada na estrutura x.xx, iniciando sempre em 1.00. Ao abrir uma Versão de Processo, seu diagrama será apresentado com apenas uma raia (horizontal), um marco (vertical) e o menu de elementos.

Figura 10. Diagrama de Versão de Processo



Fonte: Criado pelo autor

O menu de elementos possui as opções de construção do processo mostradas no Quadro 1, explicado mais abaixo no texto.

Quadro 1. Menu de elementos

1. Seleção	2. Eventos Intermediários
3. Conexão	4. Eventos Finais
5. Atividades	6. Eventos Acopláveis
7. Gateways	8. Anotação de Texto
9. Eventos iniciais	10. Outras ferramantes

Fonte: Criado pelo autor

A barra da raia possui um menu próprio com opção para inserir acima, inserir abaixo, configurar raia, alterar texto e deletar raia. A barra de marco contém a opção de inserir marco, alterar texto e deletar marco.

Ao selecionar qualquer elemento em um diagrama, é exibida a barra de ferramentas do elemento. Para todo elemento, sua barra de ferramentas possui as opções mostradas no Quadro 2.

Quadro 2. Ferramentas de elemento

1. Criar conexão	2. Configurações
3. Remover	4. Criar elemento à frente
5. Renomear	6. Recolorir

Fonte: Criado pelo autor

2.3.2.1. Elementos de processo

Os elementos disponíveis para uso do menu de elementos seguem a notação BPMN. São elementos de processo Eventos, Atividades, Gateways e Conexões.

2.3.2.1.1. Eventos

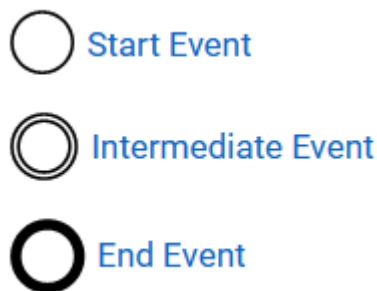
Eventos são componentes usados para representar a ocorrência de atividades em um processo. Essas atividades podem ocorrer no início de um processo (evento inicial), durante a execução de um processo (evento intermediário) ou no final de um processo (evento final).

Os tipos de eventos na plataforma são: Simples, de Mensagens, de Sinal, de Conexão, Temporais e de Erro. Além dos eventos mencionados, há também os eventos acoplados, que representam ações que ocorrem durante a execução de atividades. Essas ações podem ou não interromper a atividade em andamento. Eventos interruptivos são representados por uma borda sólida, enquanto eventos não interruptivos são representados por uma borda tracejada.

De acordo com o BPM, um evento simboliza algo que ocorre durante o curso do processo. Existem três tipos principais de eventos (Figura 11):

- Evento de início (*Start Event*) - indica onde o processo se inicia;
- Evento final (*End Event*) - indica onde um caminho do processo se encerra;
- Evento intermediário (*Intermediate Event*) - indica que algo acontece entre o início e fim do processo.

Figura 11. Tipos de Eventos



Fonte: BPMN Quick Guide










Cada tipo de evento descrito pode ser subdividido em subtipos. Os subtipos mais utilizados em fluxos normais estão descritos nos Quadro 3, Quadro 4 e Quadro 5.

Quadro 3. Eventos de Início

Marcador	Gatilho	Descrição
	Nenhum	Não há nenhum gatilho definido.
	Mensagem	O processo se inicia com o recebimento de uma mensagem.
	Temporizador	Uma data, ou tempo, ou ciclo específico pode ser definido como gatilho que inicia o processo.
	Condicional	O processo inicia quando alguma condição se torna verdadeira.
	Sinal	Um sinal vindo de outro processo inicia o processo com este evento de início.
	Múltiplo	Indica que há múltiplas formas de se iniciar o processo e apenas o gatilho de uma delas é necessário.
	Múltiplo paralelo	Indica que há múltiplas formas de se iniciar o processo e o gatilho de todas é necessário.


Fonte: BPMN Quick Guide














Quadro 4. Evento final

Marcador	Gatilho	Descrição
	Nenhum	Não há nenhum resultado definido.
	Mensagem	Uma mensagem é enviada ao fim do processo.
	Sinal	Um sinal é enviado a outro processo ao encerrar o processo com este evento final.
	Múltiplo	Indica que há múltiplas consequências ao encerrar o processo e todas elas ocorrem.
	Agravamento	Indica que um agravamento é disparado e outro evento de agravamento intermediário fará a captura desse gatilho. Outros fluxos não são afetados por este evento e continuam a ser executados.
	Erro	Indica que um erro deve ser disparado e todos os fluxos do processo são interrompidos. O erro é capturado por um evento intermediário de erro de algum outro processo.
	Compensação	Indica que uma compensação, como um reembolso, é acionada ao encerrar o processo.
	Cancelamento	Indica que uma transação deve ser cancelada dentro de um subprocesso. O evento é capturado por um evento intermediário de cancelamento e representa o cancelamento de um processo.
	Terminativo	Todas as atividades do processo devem ser encerradas imediatamente.

Fonte: BPMN Quick Guide

Quadro 5. Eventos intermediários

Marcador	Gatilho	Descrição
	Nenhum	Indica alguma mudança de estado no processo.
Throw	Mensagem	Utilizado para envio de mensagem (<i>throw</i>), com

 Catch 		preenchimento, ou recebimento de mensagem (<i>catch</i>), sem preenchimento.
	Temporizador	Utilizado como mecanismo de atraso baseado em data, ou tempo, ou ciclo específico.
	Condicional	É acionado quando uma condição se torna verdadeira.
Throw  Catch 	Sinal	Utilizado para envio de sinal(<i>throw</i>), com preenchimento, ou recebimento de sinal (<i>catch</i>), sem preenchimento. Um sinal é uma comunicação dentro e entre níveis de processo, entre raias, e entre processos.
Throw  Catch 	Compensação	Utilizado para disparar uma compensação(<i>throw</i>), com preenchimento, ou capturar uma compensação (<i>catch</i>), sem preenchimento.
Throw  Catch 	Link	Utilizado para conectar dois elementos de um mesmo processo em que não seja possível ou viável visivelmente conectá-las diretamente no fluxo. Quando utilizado para “ir” (<i>throw</i>) ao evento alvo, o símbolo é preenchido. Quando é usado para "capturar" (<i>catch</i>) do evento de origem, o símbolo é sem preenchimento.
Throw  Catch 	Múltiplo	Significa que há múltiplos gatilhos designados ao evento. O evento pode ser de captura (<i>catch</i>) ou de acionamento (<i>throw</i>) de um ou mais gatilhos.
	Múltiplo paralelo	Significa que há múltiplos gatilhos designados ao evento. O evento pode apenas ser de captura (<i>catch</i>) e requer que todos os gatilhos sejam acionados.

Fonte: BPMN Quick Guide

Alguns subtipos dos eventos de início e intermediário ainda podem se dividir em interruptivo e não-interruptivo. Eventos interruptivos possuem um símbolo com borda sólida e interrompem o fluxo do processo que os aciona. Eventos não-interruptivos são representados por uma borda tracejada e não interrompem o fluxo do processo que os aciona. Os eventos interruptivos e não-interruptivos são ilustrados na Figura 12.

Figura 12. Eventos interruptivos e não-interruptivos

Interrupting



Non-Interrupting



Fonte: BPMN Quick Guide

2.3.2.1.2. Atividades

No contexto de um processo de negócio, as atividades são responsáveis pela execução de tarefas específicas. Cada atividade tem um tempo de execução definido, requer recursos da organização, recebe *inputs* e produz *outputs*. Sua estrutura é a mesma de uma classe, contendo as mesmas opções.

Os tipos de atividades e seus objetivos são:

- Atividade Humana - é realizada por um usuário do sistema designado na sua respectiva raia;
- Atividade Genérica - é usada para representar uma ação não totalmente definida, tendo utilidade apenas visual;
- Tarefa de Execução de Script - é automática e executada pelo sistema;
- Tarefa de Invocação de Serviço - tem a mesma utilidade que a tarefa acima, mas é usada para representar atividades de integração externa.
- Subprocesso - permite referenciar um processo já criado e assim ser usado no processo;
- Subprocesso Embutido - é usado para modelar um processo dentro de outro para alcançar uma maior legibilidade das regras de negócio.

A atividade humana é uma das configurações mais importantes do processo, pois impacta diretamente na experiência do usuário. Ela é altamente personalizável e possui diversas opções de configuração para fornecer a maior flexibilidade possível.

Atividades humanas são tarefas executadas por usuários dentro do processo. Elas são configuradas com um formulário que é exibido durante a execução. A atividade humana é uma classe e também possui campos e métodos.




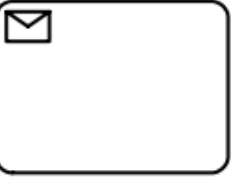
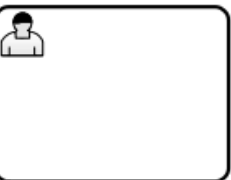

As regras de preenchimento são definidas de acordo com a configuração feita na atividade. Ela pode conter campos próprios criados especificamente para o formulário ou campos referenciados da versão do processo. Também possuem métodos padrões para automatização de lógicas e possibilidade de criação de métodos personalizados. Por fim, existem dois campos booleanos, opção para devolver a atividade para o status “Não atendida” e opção para exclusão da instância do processo que gerou aquela atividade.


Já conforme o BPM, a atividade, também conhecida como tarefa, é um elemento indivisível dentro do fluxo de um processo. Ela é utilizada quando o trabalho no processo não

pode ser subdividido em tarefas menores. Normalmente, a execução da tarefa é realizada por usuários finais e/ou aplicativos.

Há vários tipos de atividades no BPMN representadas de acordo com suas características. O Quadro 6 mostra os principais tipos.

Quadro 6. Tipos de Atividades

Representação	Tipo	Descrição
	Tarefa Abstrata	É um tipo de Tarefa não especificada utilizada para representar atividades genéricas que ainda não foram definidas e detalhadas.
	Tarefa de Serviço	É uma atividade que é realizada por um sistema ou aplicativo automatizado, em vez de ser executada manualmente por um usuário. Ela representa um trabalho que é executado por meio de um serviço, que pode ser acessado por um processo automatizado ou por outros serviços.
	Tarefa de Envio	É um tipo de Tarefa que representa a emissão de uma mensagem, sinal ou evento para uma entidade externa ao processo. Ela não espera por uma resposta e não precisa de um participante humano para ser concluída. Ao ser executada, a tarefa de envio emite a mensagem e segue adiante no processo.
	Tarefa de Recebimento	É um tipo de Tarefa que espera a recepção de uma mensagem ou evento antes de continuar o fluxo de trabalho do processo. Ela aguarda até receber uma mensagem ou evento específico para avançar para a próxima atividade.
	Tarefa de Usuário	É um tipo de Tarefa que representa uma atividade que deve ser realizada por um usuário ou um grupo de usuários em um processo. Geralmente, ela requer alguma forma de interação humana para ser concluída.
	Tarefa Manual	É um tipo de Tarefa que representa um trabalho que deve ser realizado manualmente por um ser humano. Ao contrário da Tarefa de Serviço, a Tarefa Manual não pode ser automatizada e requer a intervenção humana para ser concluída.

	Tarefa de <i>Script</i>	É um tipo de Tarefa que executa um script em alguma linguagem de programação como parte de uma atividade de processo. A Tarefa de Script é uma tarefa automática, o que significa que não requer interação do usuário e é executada pelo sistema.
---	-------------------------	---

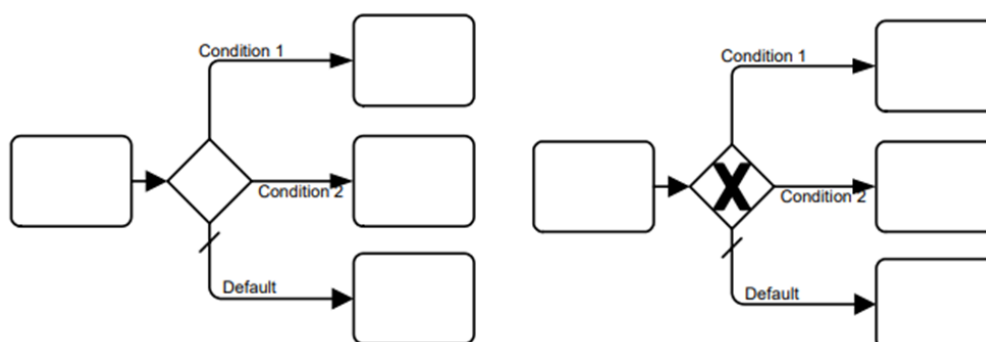
Fonte: BPMN Quick Guide

2.3.2.1.3. Gateways

Os *Gateways* são usados para controlar o fluxo de sequências em um processo e podem mesclar ou dividir tal fluxo com base em determinadas condições. Eles não representam trabalho real sendo feito e não têm impacto em medidas operacionais. Os *Gateways* podem definir vários tipos de comportamento, incluindo decisões exclusivas, inclusivas e complexas, mesclagem, ramificação e junção. Cada tipo de *Gateway* é indicado por um marcador no diagrama.

O Gateway Exclusivo divergente é utilizado para criar caminhos alternativos em um processo, podendo seguir apenas um caminho. A decisão a ser tomada deve possuir um conjunto de respostas alternativas, e cada resposta está associada a uma condição representada pelo fluxo de saída. Esse Gateway é representado por um losango com ou sem um indicador interno (X), como mostrado na Figura 13.

Figura 13. *Gateway* Exclusivo Divergente



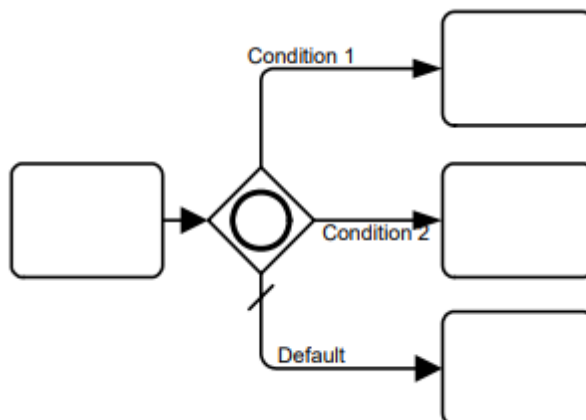
Fonte: BPMN 2.0 by Example

Caso nenhuma das expressões condicionais sejam verdadeiras, um caminho padrão pode ser definido. Por outro lado, se nenhuma das expressões condicionais for verdadeira, e nenhum caminho padrão for especificado, uma exceção de tempo de execução ocorrerá.

O Gateway Inclusivo divergente (Figura 14), da mesma forma que o Exclusivo, é utilizado para criar alternativas em um processo. Entretanto, diferente do Exclusivo, uma ou

mais das condições podem ser avaliadas como verdadeiras e todos os caminhos alternativos verdadeiros são acionados. Assim, a avaliação verdadeira de uma condição não exclui a avaliação de outra condição. Nesse Gateway, pelo menos uma condição deve ser verdadeira e um caminho padrão deve estar definido.

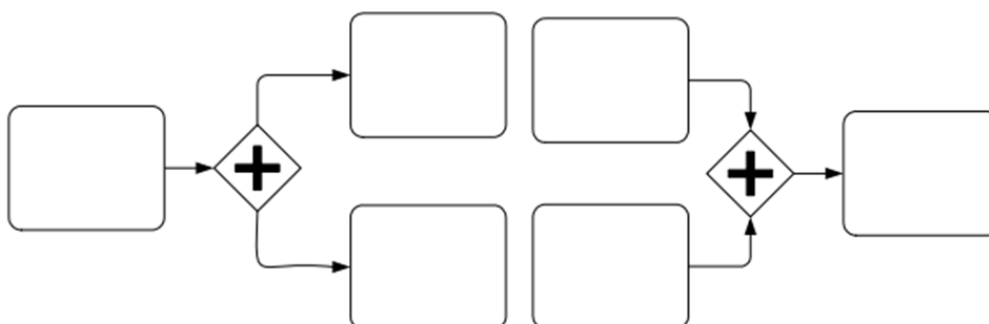
Figura 14. *Gateway* Inclusivo Divergente



Fonte: BPMN 2.0 by Example

O *Gateway* Paralelo possui duas funções: o *Gateway* Paralelo divergente é utilizado para criar fluxos paralelos; já o *Gateway* Paralelo convergente é utilizado para uni-los. Condições não são definidas para criar esse tipo de *Gateway*. Os fluxos saindo do *Gateway* são acionados e seguidos de forma independente e simultânea. Dois caminhos diferentes são combinados em um único caminho utilizando esse tipo de *Gateway*. O *Gateway* Paralelo divergente e o *Gateway* Paralelo convergente são mostrados na Figura 15.

Figura 15. *Gateway* Paralelo Divergente e Convergente

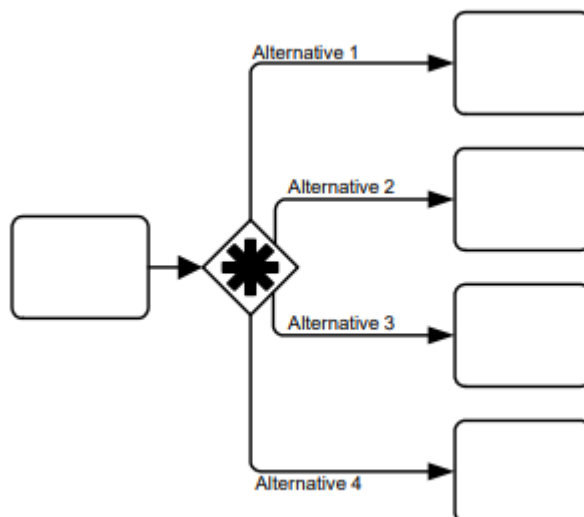


Fonte: BPMN 2.0 by Example

O *Gateway* Complexo é utilizado para modelar comportamentos de sincronização de fluxo complexos. Ele permite que múltiplas condições sejam avaliadas simultaneamente e, com base nas avaliações, o fluxo de processo seja dividido em caminhos diferentes. Assim,

ele torna o processo mais flexível e adaptável. O *Gateway* Complexo é mostrado na Figura 16.

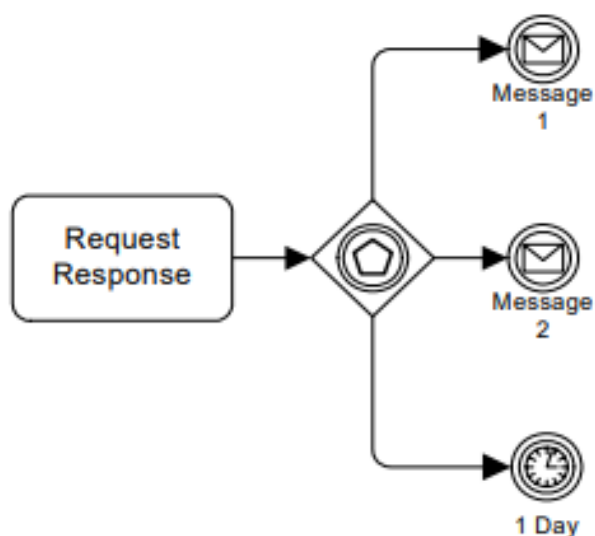
Figura 16. *Gateway* Complexo



Fonte: BPMN 2.0 by Example

Por fim, o *Gateway* Baseado em Evento (Figura 17) é utilizado para representar um ponto no processo em que os caminhos alternativos que seguem o *Gateway* são baseados em eventos ao invés de condições. Ele espera até que um evento ocorra, normalmente o recebimento de uma mensagem, e, em seguida, decide qual caminho de sequência deve ser seguido com base em uma regra de saída definida previamente.

Figura 17. *Gateway* Baseado em Evento



Fonte: BPMN 2.0 by Example

2.3.2.1.4. Conexão

As conexões definem a ordem do fluxo, que é representado pelas setas. Podem assumir três tipos distintos: Fluxo Padrão, Fluxo Condicional e Fluxo Normal.

2.3.2.2. Versão de processo

A Versão de Processo é um campo do objeto Processo que serve para configurar as definições de um processo. A configuração permite definir quais campos representarão as instâncias, os papéis dos usuários e muito mais. A Versão de Processo contém:

- Processo ao qual a versão faz referência;
- Rótulo com o nome da versão do processo;
- Descrição do processo;
- Campos do processo;
- Métodos padrões e opção para criação de métodos personalizados;
- Papéis com definição dos usuários e grupos de usuários que atuarão no processo.
- Referência para processos embutidos utilizados na modelagem;
- Status ativo ou inativo da versão;
- Opção avançado, que possibilita documentar detalhadamente o processo e também acessar o arquivo de importação, caso a versão tenha sido criada dessa forma.

2.3.2.2.1. Campo

Os processos são automaticamente integrados aos dados da organização, permitindo que seus atributos referenciem as classes criadas no ECM. Os campos do processo podem ter duas funções distintas: capturar informações que serão usadas no decorrer do processo ou exibir informações que foram coletadas anteriormente.

Na Versão de Processo, os campos desempenham um papel importante na aparência dos *cards* das instâncias, assim como acontece com os objetos das Classes. Campos com relevância Identidade e Destaque são responsáveis por definir as informações visuais exibidas para uma instância, influenciando na sua aparência.

Outra função importante dos campos do processo é a possibilidade de serem identificados pela ferramenta de busca da solução. Isso permite que tais campos sejam utilizados na criação de relatórios pela ferramenta Analytics.

2.3.2.2.2. Métodos

Os métodos da Versão de Processo possuem funcionalidade semelhante aos métodos das Classes, sendo utilizados para automatização do processo. Todo processo possui os seguintes métodos padrões:

- Iniciar: executado somente uma vez ao criar a instância;
- Obter Metadata: permite controlar as características internas dos campos;
- Finalizar: executado somente uma vez ao concluir a instância.

Além destes métodos, também é possível criar métodos personalizados para automatizar quaisquer regras de negócio.

2.3.2.2.3. Papéis

Os papéis são responsáveis por determinar quem tem permissão para executar as atividades humanas em cada Raia do processo. Cada papel tem as seguintes informações:

- Nome: Identifica o nome do papel;
- Identificador: registro único atribuído ao papel;
- Nível de acesso: define permissão de acesso;
- Autorização, sendo dividido em dois tipos:
 - Visualizar: Permite visualizar instâncias e executar atividades na sua raia;
 - Gerenciar: Além de visualizar e executar atividades, pode alterar os responsáveis pelas tarefas, apagar e intervir manualmente nas instâncias.
- Ver no fluxo: permite aos usuários visualizarem a evolução dos fluxos das instâncias. Isso inclui acompanhar o progresso das atividades, verificar o histórico de ações realizadas e identificar possíveis problemas ou gargalos no processo.

3. FUNDAMENTAÇÃO TEÓRICA

Para que seja observado em sua totalidade, é relevante que o fluxo de atividades do processo seja retratado por meio de esquematização gráfica. Essa esquematização é intitulada de mapeamento ou desenho de processos. O mapeamento esboça operacionalmente quais são as atividades inter-relacionadas e quais as instâncias responsáveis pela sua realização.

O Modelo e Notação de Processos de Negócios (BPMN), de acordo com o BPM CBOK (2013), é um padrão de notação que apresenta um conjunto robusto de símbolos para modelagem de diferentes aspectos de processos de negócio (BPM CBOK, 2013, p. 79). O padrão de notação define uma série de símbolos que descrevem relacionamentos e ordem de precedência. Um dos grandes ganhos da adoção desse padrão é a padronização dos desenhos de processo. Sobre esse benefício, o BPM CBOK (2013) traz que:

“[...] À medida que a organização amadurece em projetos de desenho de processos, o método se torna padronizado e apresentado às equipes como um padrão da organização a ser utilizado em esforços continuados” (BPM CBOK, 2013, p. 150).

A padronização dos desenhos de processo torna possível a definição de uma abordagem de gerenciamento comum a todos os processos em toda a organização e em todos os níveis de detalhamento do fluxo de trabalho.

3.1. BPM

BPM, ou Gestão de Processos de Negócio, é uma abordagem estratégica para o aprimoramento e otimização dos processos empresariais. Segundo Capote (2012, p. 41), em seu livro "BPM para todos", a gestão de processos é essencial para as organizações que desejam alcançar redução de custos, excelência operacional e melhor relacionamento com os clientes, chamado pelo autor de “tríplice coroa” organizacional.

O objetivo do BPM é mapear, modelar, analisar, otimizar e automatizar os processos de uma empresa, promovendo a melhoria contínua. Capote destaca que a modelagem de processos é uma etapa fundamental, pois permite “criar um mecanismo essencial para compreensão, documentação, análise, desenho, automatização e medição de atividades de negócio, bem como os seus recursos e as interações entre o negócio e o ambiente” (CAPOTE, 2015, p. 130).

Em seu livro "Guia para formação de analistas de processo", Capote (2015) destaca a importância dos analistas de processo na implementação do BPM. Esses profissionais são responsáveis por identificar oportunidades de melhoria, realizar a análise dos processos existentes e propor soluções eficientes.

Para tanto, os analistas de processo devem conhecer e utilizar ferramentas e técnicas específicas que auxiliam no mapeamento e automação de processos. A automação permite à organização adquirir “a capacidade de observar como as atividades estão sendo realizadas, se estão atendendo às regras e condições descritas e aplicadas e, principalmente, passam a contar com uma importantíssima capacidade de análise de desempenho operacional” (CAPOTE, 2015, p. 42).

Em suma, o BPM é uma abordagem que visa aprimorar os processos de negócio por meio da análise, modelagem e automação (CAPOTE, 2012). O BPM é uma prática essencial para as organizações modernas que desejam se manter competitivas, promovendo a eficiência, a qualidade e a inovação em seus processos.

3.2. BPMN

BPMN, ou Notação de Modelagem de Processos de Negócio, é uma linguagem gráfica amplamente utilizada para representar processos empresariais. Conforme Capote (2012, p. 127), o BPMN desempenha um papel fundamental na modelagem e documentação dos processos, permitindo que os processos sejam descritos em diferentes níveis de detalhamento e de forma padronizada .

O BPMN é composto por elementos gráficos, como atividades, eventos, gateways e fluxos, que permitem representar visualmente as diferentes etapas e interações do processo. Essa representação gráfica facilita a comunicação entre os envolvidos no processo, eliminando ambiguidades, estabelecendo uma compreensão comum, facilitando a identificação de gargalos, ineficiências e oportunidades de melhoria nos processos.

Diante disso, torna-se evidente que o BPMN desempenha um papel significativo na gestão de processos de negócio, oferecendo à organização e aos envolvidos uma representação visual clara e compreensível dos fluxos de trabalho. Além disso, essa ferramenta promove uma comunicação efetiva, estimula a colaboração e impulsiona a melhoria contínua dos processos organizacionais.

3.3. Automação de Processos

Capote (2012), em seu livro "BPM para todos", traz que a automação de processos consiste no uso de tecnologias e ferramentas para automatizar atividades e fluxos de trabalho, reduzindo a necessidade de interação humana e sendo mais utilizada em processos produtivos físicos.

A automação permite a substituição de tarefas manuais e repetitivas por sistemas automatizados, proporcionando ganhos de tempo e redução de custos. Capote (2015, p. 42) destaca que essa prática possibilita um maior controle e monitoramento dos processos, facilitando a tomada de decisão e o gerenciamento das atividades organizacionais.

Ao adotar a automação de processos, as organizações podem alcançar melhorias significativas em sua eficiência operacional. Capote (2015) ressalta que a automação contribui para a padronização e integração das atividades, reduzindo variações e aumentando a qualidade dos resultados, e também para um aprimoramento global de todo o processo.

4. ATIVIDADES DESENVOLVIDAS

Neste Capítulo são apresentadas as atividades desenvolvidas durante o estágio na Empresa Alpha.

4.1. Treinamento

Ao iniciar os trabalhos, a primeira atividade obrigatória exigida pela empresa foi a participação em um treinamento que abordou os seguintes assuntos: o uso da plataforma comercializada pela empresa, conceitos de gestão de processos e a utilização de queries para busca em banco de dados. O curso teve duração aproximada de 30 dias.

A empresa oferta em seu portal de serviço interno, chamado Universidade, um conjunto de materiais para estudos organizados por módulos e tópicos desde os conceitos mais simples até os mais complexos.

Os materiais são compostos de vídeos, textos, imagens e exercícios práticos. O participante recebeu acesso a um ambiente de treinamento para praticar e se familiarizar com a plataforma. Durante esse período, um colaborador experiente foi designado para esclarecer dúvidas, acompanhar o desenvolvimento e orientar o participante. Ele inclusive forneceu feedbacks sobre o andamento do treinamento ao time de Recursos Humanos e Liderança da área de atuação.

Vale ressaltar que esse ambiente de treinamento é o mesmo utilizado pelos clientes ao adquirirem licença de uso da plataforma.

4.2. Mapeamento de classes e processos

De forma a ilustrar brevemente a rotina de trabalho na etapa que antecede e gera uma demanda de trabalho do estagiário, é possível delinear o seguinte fluxo: (1) As demandas são geradas a partir de reuniões com o cliente conduzidas pelo time comercial; (2) Depois que a necessidade de uma prova de conceito é identificada, o *Delivery Head* entra em contato com o cliente para entender a necessidade e coletar informações; (3) Em alguns casos, o *Delivery Head* coleta os requisitos durante as conversas com o cliente, enquanto em outros casos, o cliente documenta os requisitos e os envia para o *Delivery Head*.

O *Delivery Head* recebe os requisitos do cliente e se reúne com a equipe técnica para alinhamento e apresentação da demanda. Nesse momento, um esboço em alto nível, tanto do diagrama de classes quanto dos processos, são desenvolvidos.

O estagiário então trabalha primeiramente na criação das classes (Figura 18), definição de seus campos, de seus métodos e relacionamentos. Caso alguma codificação nos métodos seja necessária, ela normalmente é feita após a conclusão da estrutura da prova de conceito, ou seja, quando a arquitetura das classes e processos está definida. Isso se dá devido a inter-relação das classes.

Figura 18. Página de criação de classe

← Criação de Classe

Nome*

Identificador*

Pacote*

Tipo*

Padrão Interface

Segue as interfaces

Objetos buscáveis?*

Campos*

Métodos*

[-] Avançado

Expiração do histórico

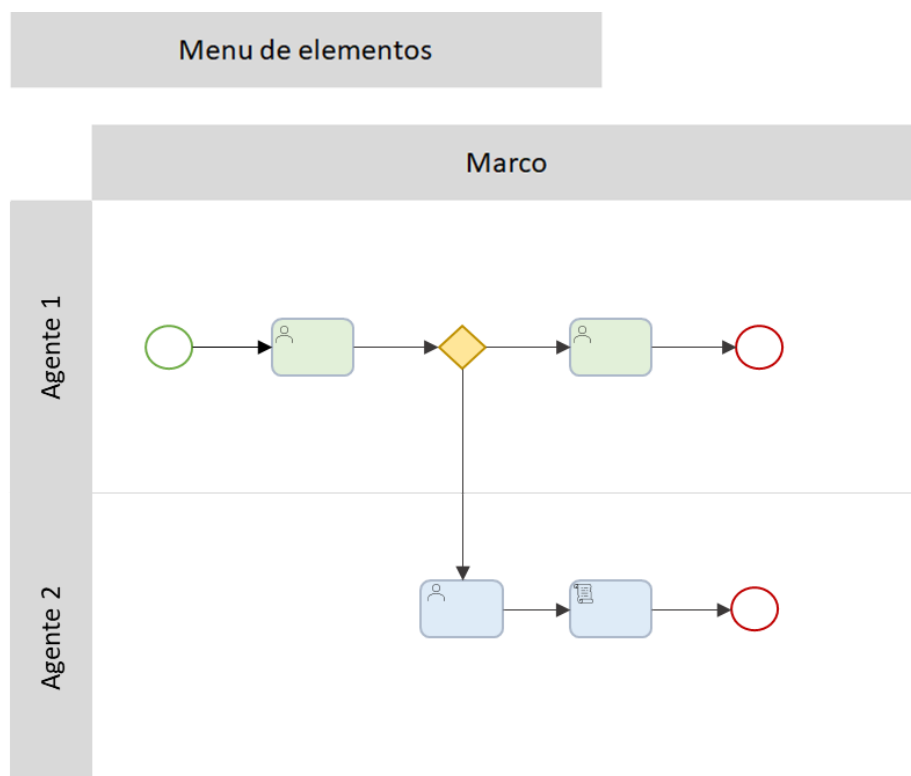
View de exibição

Fonte: Criado pelo autor

O próximo passo é o mapeamento dos processos necessários. As classes precisam já estarem concluídas pois, muitas vezes, os campos dos processos fazem referências às classes. Nessa etapa, é feito o desenho de cada processo de acordo com a Notação de Mapeamento de Processos adotada na plataforma. Os elementos são adicionados ao diagrama definindo:

- raias;
- atividades e seu tipo;
- eventos;
- gateways;
- subprocessos.

Figura 19. Diagrama de processo



Fonte: BPMN 2.0 by Example

Depois da conclusão do fluxograma, é necessário configurar os campos, métodos e papéis (responsáveis pela execução) do processo. Qualquer codificação necessária nos métodos das classes e dos processos é realizada após esse momento. Apenas com a conclusão do desenho e da configuração do processo é que se segue com a automação.

4.3. Automação de processos

A automação de processos é a configuração de cada elemento do processo - ela depende do tipo de elemento que está sendo automatizado. A seguir é apresentada uma visão geral da automação para cada elemento básico de um processo.

- Vincular os papéis às respectivas raias traçadas no processo mapeado, definindo assim os responsáveis pelas atividades de cada raia;
- Definir o tipo de automação para o evento inicial, podendo adotar: um início padrão sem nenhum parâmetro de entrada; ou conforme um tempo definido; ou de modo automatizado, podendo ocorrer com recebimento de parâmetros de

entrada de formulário do Portal de Relacionamentos, ou do chatbot, ou de APIs, ou de outro processo;

- Para o caso de atividades, sua automação depende do seu tipo. Para atividades de *script* ou serviço, é definido o código. Para atividades de usuário, é definido um formulário que contenha os campos do processo e, se necessário, os campos específicos da atividade. Se houver necessidade de criar métodos, a codificação de seu *script* também é realizada neste momento. Cada atividade também pode conter *scripts* com a execução de lógicas nas etapas de sua criação, de seu atendimento por algum usuário e/ou de sua finalização;
- Definir as condições a serem avaliadas em cada *gateway*;
- Se houver subprocesso, deve-se estabelecer os parâmetros de entrada e saída, além de realizar o mapeamento e automação conforme descrito nos tópicos 4.2 e 4.3;
- O evento final é automatizado apenas nos casos quando há necessidade de configurar saídas de dados do processo;
- Setas de fluxo também podem receber scripts e podem ser automatizadas com alguma lógica.

Com a conclusão da automação, os usuários finais são criados e vinculados aos papéis do processo. Em seguida, para realização dos testes, o ambiente deve ser preenchido com dados, criando objetos para as classes utilizadas nos processos. Assim, os testes são iniciados e, se necessário, correções são feitas. Essas correções podem impactar a estrutura das classes e processos, exigindo alterações no que já foi criado.

Quando os testes forem concluídos com sucesso, é preciso gerar dados no ambiente com execução de várias instâncias dos processos para apresentação ao cliente.

4.4. Portal de Relacionamentos

Algumas provas de conceito exigem a criação e configuração de um portal de relacionamento *Web* para armazenar os serviços criados através de classes e processos, para que os usuários finais possam acessá-los. Essa é a principal interface de comunicação entre a plataforma e o usuário final.

Para criar o portal, é necessário criar um objeto da classe Portal, criar os serviços que acessam as classes e processos desenvolvidos na plataforma e associá-los ao portal, além de organizar a página *Web* definindo catálogos e categorias.

Qualquer funcionalidade da plataforma pode ser acessada via portal de relacionamentos. Cada portal e serviço possui uma URL específica podendo ser compartilhada via quaisquer meios de comunicação.

Outra demanda frequente é a documentação das classes e processos. Essa atividade também é realizada no portal de relacionamento, por meio da criação de textos utilizando *Markdown* ou HTML. Essa documentação é configurada de acordo com a necessidade, podendo ter menus, tópicos, imagens e vídeos.

4.5. Relatórios *Analytics*

A criação de relatórios a partir da funcionalidade *Analytics* também é uma demanda frequente nas provas de conceito. Essa funcionalidade é nativa da solução e é bastante flexível. Os relatórios podem ser criados a partir de dados de classes ou processos. Tais dados são gerados durante a execução de processos e criação de objetos de classe.

Ela permite diversas configurações e definições, tais como:

- Definição de tipos de visualização, tais como gráficos de pizza, de barra, tabelas, listas ou *pipeline*;
- Definição de atributos a partir dos campos dos processos e classes;
- Definição de métricas e funções de agregação como soma, média e contagem;
- Definição de filtros; e
- Definição de parâmetros.

Os relatórios podem também ser inseridos como serviços no portal com atualização em tempo real.

4.6. Chatbot *Whatsapp*

A atividade de desenvolvimento de *chatbot* envolve várias etapas de configuração. Inicialmente, é necessário criar e configurar uma aplicação no *Facebook Developer* para utilizar o *chatbot* no *Whatsapp*. O próprio *Facebook* disponibiliza um número para se

comunicar com o *chatbot* criado na plataforma. Cabe salientar que a conexão entre o *Facebook* e a plataforma é feita via API.

Essa funcionalidade possui uma ferramenta específica de mapeamento e automação de processos, mas apresenta algumas diferenças em relação à ferramenta principal. Os tipos de ferramentas possíveis são:

- Atividade de *chatbot*: são mensagens pré-definidas que o *chatbot* envia ao usuário durante sua execução;
- Atividade de *script*: são códigos com alguma lógica e permite acesso a qualquer informação dentro da solução;
- Atividade de formulário: são mensagens que requerem alguma interação com o usuário.

Em seguida, é realizada a automação dos elementos do processo de *chatbot*, definindo mensagens enviadas, parâmetros coletados do usuário e *scripts* com as lógicas necessárias. A plataforma possui uma classe chamada "*Chat*" do pacote "Comunicação", que cria um objeto do tipo *chat* toda vez que o *chatbot* é acionado. Assim, é possível enviar mensagens para o número disponibilizado pelo *Facebook* e iniciar a execução do processo para testes dentro do próprio objeto de *chat* gratuitamente.

Figura 20. Visualização de *chat*

Classes	Objetos	Visualização de objeto
<ul style="list-style-type: none"> ▶ Nome_pacote ▶ Nome_pacote2 ▼ Comunicação <ul style="list-style-type: none"> • Chat 	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> P Positivo Contato: (35)99999-9999 </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> M Maria Contato: (35)99999-9999 </div> <div style="border: 1px solid #ccc; padding: 5px;"> J José Contato: (35)99999-9999 </div>	<div style="border: 1px solid #ccc; padding: 5px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> P Positivo ⋮ </div> <div style="margin-top: 5px;"> Contato: (35)99999-9999 </div> <div style="background-color: #f1c40f; padding: 10px; margin-top: 10px;"> <div style="text-align: right; margin-bottom: 10px;"> HOJE </div> <div style="background-color: #f9e79f; padding: 10px;"> <!-- Chat content area --> </div> </div> <div style="border-top: 1px solid #ccc; padding-top: 5px; margin-top: 10px;"> <div style="display: flex; justify-content: space-between; align-items: center;"> 😊 <input type="text" value="Digite aqui..."/> 📎 📷 🎤 </div> </div> </div>

Fonte: Criado pelo autor

4.7. Integrações

É comum que os requisitos contenham integrações com outros sistemas do cliente. Nesses casos, antes de iniciar a implementação da prova de conceito, é preciso entender o funcionamento da API e testá-la.

Primeiramente, é necessário estudar a documentação da API para compreender como realizar as chamadas corretamente. Em seguida, faz-se uso do Postman para realizar a chamada API e testá-la. Obtendo sucesso, entende-se a estrutura da chamada e resposta da API para mapeá-las na plataforma.

O mapeamento é realizado desenvolvendo código em uma classe ou elemento de processo, como atividades de *script* ou serviços. O objetivo é obter e armazenar o resultado da chamada em campos de classe ou processo para uso na plataforma.

5. CONSIDERAÇÕES FINAIS

Durante o estágio realizado na empresa de software Alpha, foi possível vivenciar a aplicação prática dos conhecimentos adquiridos ao longo do curso de Sistemas de Informação. Com base nessa experiência, o estagiário teve a oportunidade de conhecer e vivenciar o cotidiano e desafios de um profissional de TI. Além disso, também o permitiu conhecer de forma holística uma plataforma robusta que realmente aplica a gestão de processos de negócio em seu *core*.

Com relação às sugestões para a empresa, é importante destacar que ela possui uma área muito recente e pequena relacionada à Inteligência Artificial (IA). Acredito que a IA será peça fundamental no futuro das plataformas SaaS e será requisito indispensável para os clientes que buscam essa tecnologia.

Quanto à sugestão para o curso de Sistema de Informação, recomenda-se abordar de forma mais prática o ensino de mapeamento de processos, empregando o uso de ferramentas gratuitas, tais como o Bizagi, em atividades voltadas para o desenho de processos.

Muitos dos conceitos vistos nas disciplinas do curso de Sistema de Informação foram pertinentes para executar as atividades do estágio. As disciplinas iniciais, tais como Introdução aos Algoritmos, Introdução a Computação e Estruturas de Dados, são fundamentais para qualquer trabalho na área de TI, pois fornecem uma base sólida para qualquer tecnologia que o estudante venha a trabalhar. Dentre todas as outras, há duas disciplinas que contribuíram de maneira direta para o bom desempenho no estágio: a disciplina Gestão de Tecnologia da Informação abordou a gestão de processos introduzindo o tema de forma detalhada e objetiva, oferecendo ao estagiário um conhecimento prévio fundamental para o início de seu trabalho na empresa Alpha; a outra disciplina foi Gerência de Projetos de Software, que abordou de forma mais aprofundada o ciclo de vida de projetos, tema premente na realidade do mercado de empresas de software.

Por fim, destaca-se que o estágio proporcionou não apenas a aplicação de conhecimentos adquiridos ao longo do curso, mas promoveu a geração de novos conhecimentos por meio da prática diária e da vivência na rotina de uma empresa do setor.

REFERÊNCIAS

ABPMP - ASSOCIATION OF BUSINESS PROCESS MANAGEMENT PROFESSIONALS. Guia para o Gerenciamento de Processos de Negócio Corpo Comum de Conhecimento (BPM CBOK) - Versão 3.0. 2013. 400 p.

OSTROFF, Frank; SMITH, Douglas. The horizontal organization. The McKinsey Quarterly, n. 1, p. 148-168, 1992.

DIFFEN. Data vs Information - Difference and Comparison | Diffen. Disponível em: <http://www.diffen.com>. Acesso em: 25 fev. 2023.

BPM. Guia BPMN 2.0 em Português. Disponível em: http://www.bpmb.de/images/BPMN2_0_Poster_PT.pdf. Acesso em: 13 abr. 2023.

BPMN Quick Guide. Disponível em: <https://www.bpmnquickguide.com/view-bpmn-quick-guide/>. Acesso em: 13 abr. 2023.

BPMN.org. Disponível em: <https://www.bpmn.org/>. Acesso em: 13 abr. 2023.

BRITTO, Gart Capote de. BPM Para Todos - Uma Visão Geral Abrangente, Objetiva e Esclarecedora sobre Gerenciamento de Processos de Negócio. 1. ed. Rio de Janeiro: Gart Capote, 2012.

BRITTO, Gart Capote de. Guia para Formação de Analistas de Processos - BPM Volume I. 2a ed. Rio de Janeiro: Gart Capote, 2015.