

**UNIVERSIDADE FEDERAL DE SANTA CATARINA  
CENTRO TECNOLÓGICO  
DEPARTAMENTO DE AUTOMAÇÃO E SISTEMAS**

**Jhonata Vinicius Tridapalli**

**Relatório de Estágio**

**Integração de *dashboards* ao Portal de Processos da UFSC para monitoramento do desempenho dos processos de negócio automatizados pelo EAPn**

Florianópolis  
2020

**Jhonata Vinicius Tridapalli**

**Integração de *dashboards* ao Portal de Processos da UFSC para monitoramento do desempenho dos processos de negócio automatizados pelo EAPn**

Relatório submetido à Universidade Federal de Santa Catarina como requisito para a aprovação na disciplina **DAS 5501: Estágio em Controle e Automação Industrial** do curso de Graduação em Engenharia de Controle e Automação.

Orientador(a): Prof. Maurício Floriano Galimberti

Florianópolis  
2020

**Jhonata Vinicius Tridapalli**

**Integração de *dashboards* ao Portal de Processos da UFSC  
para monitoramento do desempenho dos processos de  
negócio automatizados pelo EAPn**

Este relatório de estágio foi julgado no contexto da disciplina DAS5501: Estágio em Controle e Automação Industrial e **APROVADO** na sua forma final pelo Curso de Engenharia de Controle e Automação.

Florianópolis, 04 de Dezembro de 2020

---

Prof. Maurício Floriano Galimberti  
Orientador  
Universidade Federal de Santa Catarina

## RESUMO

O Escritório de Automação de Processos de Negócio (EAPn) é resultado de um Projeto de Pesquisa institucionalizado no Laboratório de Sistemas de Conhecimento (LSC) junto ao Departamento de Informática e Estatística (INE) do Centro Tecnológico (CTC) da Universidade Federal de Santa Catarina (UFSC). Este projeto é destinado a automação dos processos de negócios acadêmicos, com o objetivo de implantar na universidade a cultura de Gestão por Processos segundo a disciplina gerencial BPM. As práticas previstas pela disciplina BPM vêm se tornando cada vez mais utilizadas pelas organizações mundiais, afinal, a eficiência de uma empresa está diretamente ligada à qualidade de seus processos. Motivado pela competitividade atual dos negócios e necessidade de aperfeiçoamento constante, além de aplicar práticas de gerenciamento BPM, é necessário identificar e avaliar constantemente o desempenho desses processos, sendo esta, umas das práticas prevista pela disciplina, considerada fator crucial para a melhoria dos processos. Dessa forma, ferramentas de monitoramento tornam-se indispensáveis, promovendo um ambiente que possibilite aos gestores uma visão geral do andamento dos processos, permitindo identificar requisitos de desempenho e assim, promover análises e melhorias. Para realizar o monitoramento dos processos automatizados pelo EAPn, tem-se por objetivo neste trabalho propor um método de integração com os atuais sistemas que possibilite a visualização gráfica de indicadores e métricas. Os resultados obtidos compreendem a integração da plataforma Metabase para análise de dados através de *dashboards* com o Bonita BPM, gerando uma página de aplicação para incorporação ao Portal de Processos automatizados da UFSC.

**Palavras-chave:** Gerenciamento de processos de negócios. BPM. Monitoramento. *Dashboard*. Metabase. Bonita BPM.

## ABSTRACT

The Escritório de Automação de Processos de Negócio (EAPn) is the result of an Institutional Research and Extension Project at the Laboratório de Sistemas de Conhecimento(LSC) with the Departamento de Informática e Estatística (INE) of the Centro Tecnológico (CTC) of the Universidade Federal de Santa Catarina (UFSC). This project is intended for the automation of academic business processes, with the objective of implementing a culture of process management in the university according to the BPM management discipline. As the practices approved by the BPM discipline, will become increasingly used by world organizations, in the end, a company's economy is directly linked to the quality of its processes. Motivated by the current business competitiveness and the need for constant improvement, in addition to applying BPM management practices, it is necessary to constantly identify and evaluate the performance of these processes, which is one of the practices provided by the discipline, considered a crucial factor for the improvement of processes. In this way, the monitoring tools become indispensable, promoting an environment that allows managers an overview of the progress of the processes, allowing to identify performance requirements and thus, promoting statistics and improvements. To carry out the monitoring of the automated processes by the EAPn, use this objective as a method of integration with the current systems that allows the graphical responses of the indicators and metrics. The results obtained include the integration of the Metabase platform for data analysis through panels with Bonita BPM, generating an application page for incorporation into the UFSC Automated Processes Portal.

**Key-words:** Business process management. BPM. Monitoring. Dashboard. Metabase. Bonita BPM.

## LISTA DE ILUSTRAÇÕES

Figura 1 – Fases do ciclo de vida BPM .....	19
Figura 2 - Peso e classificação de cada índice de indicadores de maturidade .....	23
Figura 3 - Principais funções dos sistemas BPM .....	25
Figura 4 – Arquitetura de comunicação entre Bonita BPM e Metabase.....	27
Figura 5 – Dashboard genérico desenvolvido no Metabase .....	30
Figura 6 - Ambiente UI Designer .....	34
Figura 7 - Diagrama de sequência UML da página de monitoramento .....	37
Figura 8 – Diagrama de classe UML .....	38
Figura 9 - Bibliotecas para a Extensão de API .....	40
Figura 10 - Requisição HTTP POST para autenticação de usuário .....	41
Figura 11 - Requisição HTTP GET para retorno de informações do dashboard .....	42
Figura 12 - Arquivo de propriedades da Extensão de API .....	43
Figura 13 - Layout da página web de monitoramento .....	44
Figura 14 - Elementos da página no ambiente de desenvolvimento UI Designer ....	47
Figura 15 - Relação entre as variáveis utilizadas no escopo da página .....	48
Figura 16 - Estrutura de navegação do Portal de Processos da UFSC .....	51
Figura 17 - Menu do Portal de Processos da UFSC .....	51
Figura 18 - Exemplo de consulta SQL e resultado gráfico .....	55
Figura 19 - Resultado gráfico do <i>dashboard</i> .....	56
Figura 20 - Página de monitoramento no Portal de Processos da UFSC .....	57

## LISTA DE TABELAS

Tabela 1 – Estrutura de requisição HTTP das APIs do Metabase .....	41
Tabela 2 - Variáveis definidas no escopo da página de monitoramento .....	49

## SUMÁRIO

<b>1 – INTRODUÇÃO .....</b>	<b>10</b>
1.1 – Problema .....	11
1.2 – Objetivos .....	11
1.3 – Justificativa .....	12
1.4 – Delimitação .....	12
1.5 – Estrutura .....	13
<b>2 – ESCRITÓRIO DE AUTOMAÇÃO DE PROCESSOS DE NEGÓCIO (EAPN) ....</b>	<b>14</b>
2.1 – Automação de Processos .....	14
2.1.1 – Bonita BPM .....	15
2.1.2 – Portal de Processos UFSC .....	16
2.1.3 – Monitoramento: motivação sob a perspectiva de automação .....	17
<b>3 – GERENCIAMENTO DE PROCESSOS DE NEGÓCIO .....</b>	<b>18</b>
3.1 – Ciclo de vida BPM .....	19
3.2 - Monitoramento .....	21
3.2.1 – Importância do monitoramento para os modelos BPMM .....	21
3.2.2 – O papel do monitoramento em sistemas BPM .....	24
3.2.3 – Ferramentas BI ( <i>Business Intelligence</i> ) .....	26
<b>4 – TEORIAS GERAIS E FERRAMENTAS UTILIZADAS .....</b>	<b>27</b>
4.1 – Metabase: Recursos de desenvolvimento .....	28
4.1.1 – Perguntas e painéis .....	29
4.1.2 – Consultas SQL .....	30
4.1.3 – Links públicos .....	31
4.2 – Bonita BPM: Recursos de desenvolvimento .....	31
4.2.1 – Estrutura de dados .....	32
4.2.2 – Páginas e aplicativos .....	33
4.2.3 – Extensão de API REST .....	34
4.2.4 – Parâmetros .....	35
<b>5 – DESENVOLVIMENTO .....</b>	<b>37</b>
5.1 – Configuração dos <i>dashboards</i> .....	38
5.2 – Extensão de API REST: Requisição de dados dos <i>dashboards</i> .....	39
5.2.1 – Arquivo Groovy .....	39
5.2.1 – Arquivo de propriedades .....	42
5.3 – Estruturação da página de monitoramento .....	43

5.3.1 – Layout .....	44
5.3.2 – Funcionalidades operacionais .....	47
5.3.2.1 – Requisição da Extensão de API REST .....	49
5.3.2.2 – Controle de acessibilidade .....	50
5.4 – Incorporação da página de monitoramento ao Portal de Processos .....	50
5.4.1 – Configuração do menu de navegação .....	51
5.5 – Construção gráfica dos <i>dashboards</i> .....	52
5.5.1 – Indicadores de desempenho implementados .....	53
5.5.2 – Programação SQL para a consulta de dados .....	54
<b>6 – RESULTADOS .....</b>	<b>57</b>
<b>7 – CONCLUSÃO .....</b>	<b>58</b>
<b>REFERÊNCIAS .....</b>	<b>60</b>

## 1. INTRODUÇÃO

Para lidar com as mudanças cada vez mais rápidas de um mercado em crescente globalização, inovação e competitividade é imprescindível às organizações e empresas aumentarem seus desempenhos e agregarem mais valor aos seus produtos e serviços. A necessidade de melhoria na gestão do negócio desencadeado pelo emaranhado de processos ineficientes torna inevitável a busca por novas práticas de gerenciamento.

Neste contexto, surgiu o conceito de gerenciamento por processo, também chamado de BPM (*Business Process Management*). A abordagem gerencial BPM foi instituída com o objetivo de sistematizar e estruturar a gestão por processos nas organizações através de práticas e princípios. Segundo a *Association of Business Process Management Professionals* (ABPMP) através do Guia BPM CBOK (*Common Book of Knowledge*), BPM é compreendido pela prática das atividades de identificar, projetar, executar, documentar, medir, monitorar, controlar e promover melhorias nos processos de uma organização (CBOK, 2013).

Para sistematizar todas as atividades pretendidas, garantindo a melhoria contínua dos processos, é definido uma ordem de atividades sequenciais denominada Ciclo de Vida BPM. Este classifica a prática gerencial do BPM em seis etapas: planejamento, análise, design e modelagem, implementação, monitoramento e controle e por último, refino (CBOK, 2013).

Para oferecer os serviços mencionados no ambiente acadêmico da UFSC, foi estruturado o Escritório de Automação de Processos de Negócio (EAPn). Atualmente seus esforços estão direcionados na modelagem e automação de processos, sendo estes, disponíveis aos usuários da universidade através de um Portal de Processos.

Para monitorar o desempenho dos processos automatizados e oferecer serviços cada vez mais qualificados para os usuários e eficientes para a universidade, originou-se a proposição deste trabalho. O foco de interesse das atividades desenvolvidas é respectivo ao monitoramento, compreendido na quinta fase do ciclo BPM, que tem por objetivo determinar informações de desempenho que possibilitem a tomada de ações para refinamento dos processos.

## 1.1. Problema

Até o momento não existe uma visão clara e acessível das informações sobre o andamento dos processos automatizados da UFSC impossibilitando a análise e a avaliação de parâmetros de desempenho. Assim, não é possível fundamentar tomadas de decisões e analisar se os requisitos propostos são atendidos.

A ausência de uma ferramenta para monitoramento impossibilita aos gestores a análise minuciosa dos dados provenientes dos processos. Por consequência, desponta a falta de praticidade em mensurar e analisar resultados, por exemplo, dentro de um determinado período de tempo, índices de erros e acertos, a evolução do desempenho de colaboradores.

## 1.2. Objetivos

Com base no problema de pesquisa exposto, o objetivo geral do trabalho é desenvolver uma solução que possibilite o monitoramento dos processos de negócios através do Portal de Processos da UFSC, operacionalizando um ambiente preliminar para futuro acompanhamento de indicadores e métricas através de *dashboards*.

Adicionalmente, tem-se como objetivos específicos:

- Implementar recursos visando sua utilização no EAPn para demais aplicações.
- Documentar funcionalidades utilizadas para integração do atual sistema BPM com recursos de softwares de terceiros.

Os objetivos específicos destinam-se a gerar conhecimento que seja útil ao EAPn em projetos futuros, isto é, quando os *dashboards* virem a ser implementados e colocados em prática e projetos que necessitem o acesso de recursos em diferentes domínios. Dessa forma, além do caráter prático de implementação, muitos dos recursos serão apresentados em formato explicativo, guiando o leitor a compreender os motivos intrínsecos à implementação.

### 1.3. Justificativa

Dentre as práticas BPM está previsto a utilização de ferramentas que possibilitem a organização monitorar os processos, permitindo o acompanhamento de indicadores e métricas que quantifiquem resultados que possam refletir o desempenho do processo. Dessa forma, é possível conduzir avaliações e análises que proporcionam aos gestores a tomada de decisões gerando melhorias que auxiliem a instituição a atingir os seus objetivos.

Em projetos de implementação BPM, geralmente o foco principal é a automatização de processos, devido aos benefícios entregues a curto prazo. Porém, para validar e demonstrar que as ações de automação tomadas representam um cenário ótimo de desempenho para a organização é inevitável a utilização de mecanismos de monitoramento. Estes mecanismos lidam com o ajuste de recursos para garantir os objetivos do processo através de medições e avaliação de desempenho (Morais et al., 2014).

Neste contexto, empregar o monitoramento por meio de *dashboards*, possibilita simplificar e promover aos gestores um ambiente de visualização dinâmico e objetivo. Esta ferramenta é amplamente utilizada para exibir em formato visual e de maneira centralizada o conjunto de informações de interesse: indicadores e suas métricas. A exibição das informações em uma interface gráfica fornece visualização rápidas dos principais fatores, tornando possível ter acesso a aspectos pontuais de forma clara e concisa.

### 1.4. Delimitação

Os aspectos englobados pelo trabalho são relativos à infraestrutura utilizada para integração da plataforma de monitoramento e desenvolvimento de *dashboards* para utilização no Portal de Processos desenvolvido pelo EAPn. São abordadas as tecnologias empregadas e o formato utilizado para integração com os atuais sistemas.

Serão definidos apenas requisitos funcionais sobre *dashboards*, ressaltando principalmente seu papel e benefícios, não cobrindo requisitos mais avançados que devem ser considerados na construção dos painéis e das informações a serem disponibilizadas aos usuários.

## 1.5. Estrutura

O trabalho está organizado em 8 capítulos, as informações apresentadas são estruturadas da seguinte forma:

No capítulo 2 são apresentadas as informações do escritório na qual este trabalho está incluso, relatando os serviços prestados à universidade e a infraestrutura na qual está inserido.

No capítulo 3 é definido os conceitos da disciplina gerencial BPM e como estes se associam às atividades de monitoramento, descrevendo a sua importância perante os serviços do EAPn.

No capítulo 4 é descrito as teorias gerais que permearam o desenvolvimento do trabalho, explicitando as ferramentas e seus recursos que foram utilizadas para desenvolvimento.

No capítulo 5 é abordado as etapas de desenvolvimento, descrevendo os aspectos do sistemas, requisitos, arquitetura e funcionalidades implementadas.

No capítulo 6 é apresentado o resultado do trabalho, demonstrando a página de monitoramento para inserção ao Portal de Processos da UFSC.

No capítulo 7 são apresentadas as conclusões decorrentes do processo de integração, descrevendo as futuras implementação para completar o atual trabalho dentro do projeto do EAPn.

No capítulo 8 são apresentadas as bibliografias utilizadas como referencial teórico para o desenvolvimento do trabalho.

## **2. ESCRITÓRIO DE AUTOMAÇÃO DE PROCESSOS DE NEGÓCIO (EAPn)**

O EAPn é resultado de um Projeto de Pesquisa junto ao Centro Tecnológico – CTC/UFSC de estruturação e implantação de um núcleo de gerenciamento por processo centrado em automação. O projeto foi idealizado pelo professor doutor Maurício Floriano Galimberti com o intuito de pesquisar a viabilidade e os benefícios da prática por gestão de processos em um ambiente acadêmico, gerando um modelo de Escritório de Automação de Processos aplicável a Instituições de Ensino Superior.

Alinhado aos fundamentos da disciplina gerencial BPM, atualmente o EAPn oferece serviços de mapeamento, modelagem, documentação e automação de processos seguindo um método desenvolvido e validado pelo escritório. O método é resultado de desenvolvimento do Prof. Maurício desde 2010, e que foi adaptado para o Trabalho de Conclusão de Curso (TCC), feito pelo então graduando em Sistemas da Informação Hugo Piovesan Gonçalves, intitulado “Guia para Modelagem e Automação de Processos de Negócios Acadêmicos: estudos de caso com processos da UFSC” de 2016 (Gonçalves, 2016). Foi desenvolvido um guia para servir como base para orientar a modelagem e a automação dos processos de negócios acadêmicos da universidade.

O projeto de modelo do EAPn foi iniciado a partir desse guia, oferecendo serviços centrados na automação com o intuito de melhorar o gerenciamento de processos de negócios da UFSC. Com um núcleo específico que tenha o conhecimento adequado sobre a disciplina BPM é possível propiciar melhor gestão sobre os processos e disseminar de forma mais objetiva as práticas de gerenciamento propostas.

### **2.1. Automação de processos**

As tecnologias de automatização devem auxiliar no desenvolvimento de soluções caracterizadas pela substituição de atividades manuais por aplicações. Além disso, devem possibilitar a criação de interfaces sistêmicas, integrações entre diferentes bases de dados, fluxos de trabalho mais ágeis e utilização assertiva de informações. Com isso, os benefícios são diversos, aperfeiçoando os processos de negócios e contribuindo para a transformação digital das empresas.

A automação apresenta-se como um meio para a melhoria dos processos e extremamente vantajosa se for alinhada às práticas BPM. Idealmente, os projetos de implementação de BPM compreendem as etapas de análise (AS-IS), redesenho (TO-BE) e implementação (TO-DO), sendo a implementação geralmente caracterizada pela utilização de tecnologias que possibilitem a automação.

Para isso, são utilizadas ferramentas de gerenciamento de processos denominadas BPMS (*Business Process Management System*) para a automação do fluxo de informação e ações de um processo, por meio de:

- Ferramentas de modelagem e análise de processos;
- Mecanismos de regras de negócios e de fluxo de trabalho;
- Ferramentas de simulação e teste.

Através do BPMS é possível tornar processos de negócios mais eficientes e adaptáveis, porém seu conjunto de ferramentas condiciona o desenvolvimento da automação, exigindo que os BPMS se adequem constantemente às novas tecnologias do mercado.

### **2.1.1. Bonita BPM**

No EAPn é utilizado o software Bonita desenvolvido pela empresa Bonitasoft, oferecendo uma plataforma de gerenciamento de processos de negócio centrada na automação de processos. O software apresenta versões gratuitas e de código aberto mantendo seus recursos principais de modelagem e automação, justificando sua utilização.

Através de uma interface simples e intuitiva, auxilia na modelagem de processos utilizando-se da notação BPMN (*Business Process Model and Notation*), representando o processo através de diagramas e fluxos. Este tipo de representação auxilia na visualização, promovendo uma melhor compreensão do processo. Desta forma, mudanças no percurso do processo podem ser feitas de maneira mais simplificada.

O Bonita oferece um ambiente completo de desenvolvimento, apresentado em 3 módulos:

- **Bonita Studio:** Ambiente de desenvolvimento dos diagramas de fluxo, definição de bases de dados, regras de negócio, pessoas envolvidas, conexão com sistemas externos.
- **Bonita UI Designer:** Ambiente composto de recursos para construção de formulários digitais para preenchimento dos dados definidos no Studio. Além disso, disponibiliza ao usuário ferramentas gráficas, de estruturação e codificação para desenvolver interfaces personalizadas.
- **Bonita Portal:** É um Portal que permite ao usuário simular os processos e executar as tarefas envolvidas. Possibilita gerenciar os processos, disponibilizando um ambiente para administrador.

Uma ferramenta BPMS completa é constituída de ferramentas que possibilitam o monitoramento dos processos através de painéis de indicadores, neste caso, estas são disponíveis apenas nas versões pagas do Bonita BPM. Porém, através de recursos avançados disponibilizados na versão gratuita é possível efetuar a integração entre sistemas e assim, desenvolver um ambiente personalizado que possibilite atingir os fins propostos pelo trabalho.

### 2.1.2. Portal de Processos UFSC

O Portal de Processos é a plataforma de acesso aos processos automatizados no Bonita BPM. Através dele os usuários da universidade conseguem iniciar processos, executar tarefas e acompanhar o andamento das atividades. O ambiente foi desenvolvido utilizando o módulo UI Designer do Bonita através do recurso *Application Page* para criação de páginas personalizadas, possibilitando a criação de um ambiente que substitua o Bonita Portal e que atenda às necessidades específicas de cada negócio, constituindo um aplicativo.

As páginas são integradas aos dados de negócio dos processos, possibilitando a recuperação direta através de API (*Application Programming Interface*)<sup>1</sup>. No aplicativo são definidos fatores de interação, lógica e layout, totalmente customizáveis e rapidamente adaptáveis às necessidades do negócio.

---

<sup>1</sup> API é um conjunto de rotinas para acesso a um software, aplicação ou plataforma web para utilização de seus serviços e recursos.

Com o objetivo de tornar esse ambiente ainda mais completo, será desenvolvida uma página de aplicação que possibilite aos gestores dos processos o acompanhamento de informações através de *dashboards*, resolvendo os problemas das limitações de monitoramento do Bonita BPM. Aspectos técnicos e conceituais usados no desenvolvimento da solução, requisitos construtivos, integração de sistemas e desenvolvimento da página serão abordados no capítulo 4.

### **2.1.3. Monitoramento: motivação sob a perspectiva da automação**

Com a automação de processos é possível se obter muitos benefícios, principalmente quando aplicado a organizações que não possuem em sua cultura práticas gerenciais como o BPM. A automação alinhada ao gerenciamento orientado por processos reduz possíveis focos de ineficiência, que a longo prazo podem desencadear problemas maiores. Além disso, visa reduzir os custos de execução, recursos financeiros e humanos possibilitando que estes sejam distribuídos para fins mais críticos (Galimberti, 2019).

Motivado por todos os possíveis benefícios mencionados, espera-se que após a automação a empresa presencie um cenário de melhoria no desempenho de seus processos. Porém, como verificar e analisar de modo objetivo e quantitativo o desempenho associado à execução do processo? Neste contexto, é apresentada a proposta do trabalho, utilizando o monitoramento dos processos automatizados como uma ferramenta para medição de desempenho.

Automatizar processos não é uma demanda única que se encerra assim que os processos atuais são automatizados. É preciso acompanhá-los de forma contínua para que melhorias sejam implementadas sempre que novas necessidades sejam identificadas. Para isso, utilizam-se ferramentas de monitoramento que possibilitem análises sistemáticas.

### 3. GERENCIAMENTO DE PROCESSOS DE NEGÓCIO

Um processo de negócio é uma sequência de atividades ou tarefas estruturadas com o objetivo de entregar algum resultado de valor para o cliente, seja por um serviço ou produto. Com a evolução e modernização, as organizações se tornaram compostas por inúmeros processos de negócio, motivando transformações em suas práticas gerenciais. Dessa forma, a abordagem de gestão por processos de negócios BPM torna-se uma solução, compreendendo práticas que possibilitem o aumento da produtividade e redução de custos.

O gerenciamento de processos de negócios visa o aperfeiçoamento contínuo dos processos, sempre através de uma visão sistêmica da organização fazendo uma gestão voltada inteiramente a processos e não a silos funcionais (departamentos). Essa nova visão permite um acompanhamento total do processo independentemente dos departamentos associados à produção do bem ou serviço, possibilitando enxergar os processos ponta-a-ponta.

Ao fazer um pedido de um produto em uma plataforma de venda online até que o mesmo seja entregue na sua residência, o mesmo percorre inúmeros setores da organização que o produz, cada um com suas respectivas tarefas, seja comercial, produção ou transporte. No momento que as tarefas do setor são realizadas, automaticamente a responsabilidade do produto se torna do próximo setor, sendo que geralmente, não há compartilhamento de conhecimento entre ambos. Desmistificando essa cultura de gestão, acompanhar um processo ponta-a-ponta, transversal a diferentes silos funcionais<sup>2</sup>, envolve o todo de uma organização, acompanhando o andamento do pedido até o mesmo ser entregue e assim, gerar valor tangível ao cliente.

Com a prática da disciplina gerencial BPM é possível conhecer os processos executados, medir e gerenciar, com a finalidade de sistematizar, facilitar e melhorar as operações envolvidas e, conseqüentemente, gerar benefícios para a organização com o aumento de produtividade e o melhor aproveitamento dos recursos.

---

<sup>2</sup> Os silos funcionais são relacionados aos setores individuais de uma empresa, responsáveis por um conjunto de atividades específicas.

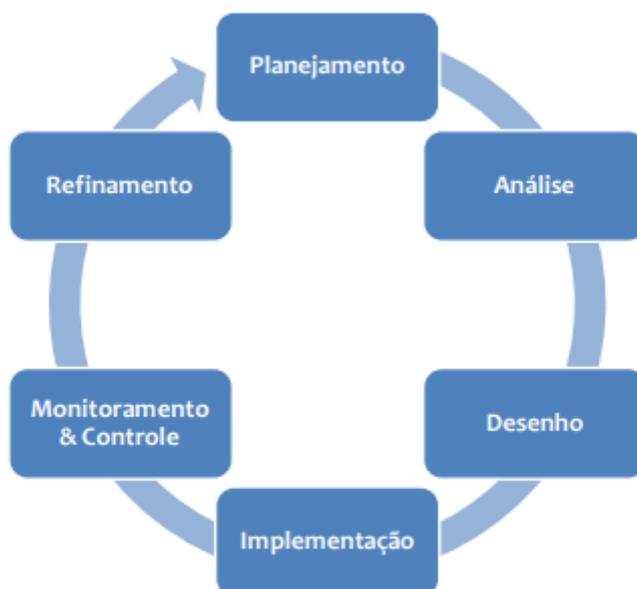
### 3.1. Ciclo de vida BPM

A gestão por processo BPM é fundamentada no princípio de melhoria contínua, implicando em um comprometimento permanente da organização. Dessa forma, deve ser garantido o alinhamento dos processos à estratégia organizacional, possibilitando alcançar os objetivos traçados.

O ciclo de vida BPM consiste em um conjunto de fases com uma sequência lógica que possibilitam o acompanhamento de todas as etapas de implementação do gerenciamento por processos, sendo semelhante às ações do ciclo PDCA: planejar (*plan*), fazer (*do*), checar (*check*) e agir (*act*). Com objetivos semelhantes, os ciclos têm o intuito de promover uma visão focada nas causas e não somente nas consequências, resultando em uma ferramenta de análise e mudança de processos de constante planejamento. Assim, busca-se identificar oportunidades de melhorias promovendo as ações necessárias para atingir os fins desejados.

Alinhado ao Guia CBOK (Guia para Gerenciamento de Processos – Corpo Comum de Conhecimento de ABPMP), a disciplina gerencial BPM propõe um ciclo de vida completo de 6 fases, orientado a necessidade de cada organização e variável de acordo com a maturidade do cenário de utilização de BPM. A Figura 1 apresenta as fases que, segundo o BPM CBOK (2013), exemplificam o modelo típico de ciclo BPM.

Figura 1 – Fases do ciclo de vida BPM



Fonte: (CBOK, 2013).

Resumidamente, cada fase pode ser descrita por:

1. **Planejamento:** Analisar informações dos processos disponíveis identificando de que maneira estão alinhados aos serviços prestados. Nesta etapa é realizado o diagrama do processo atual, conhecido também por modelo AS-IS (Como é).
2. **Análise:** É realizada uma análise do atual processo de negócio, compreendendo um estudo sobre todo o ambiente do negócio.
3. **Desenho:** É realizado o diagrama do processo, criando um modelo com o objetivo de representar a situação futura do processo, conhecido também por modelo TO-BE (Como será). Esse modelo é uma projeção concebida a partir da análise do processo AS-IS.
4. **Implementação:** Consiste na execução dos processos que foram identificados, mapeados e redesenhados nas fases anteriores, atribuídos de serviços de automação e possíveis mudanças organizacionais.
5. **Monitoramento e controle:** Obter informações resultantes da execução dos processos que possibilitem estabelecer medidas de desempenho e possíveis focos de melhorias.
6. **Refinamento:** Mediante os resultados da fase anterior associado às metas estabelecidas e melhorias propostas, são executadas as alterações nos processos buscando otimizações e benefícios à organização.

Considerando o processo de adequação da universidade às práticas de gerenciamento propostas, e devido ao pouco tempo transcorrido desde a implantação e disponibilização dos serviços do EAPn, o gerenciamento dos processos atualmente contempla, primordialmente, as fases 1, 2, 3 e 4. Para essas fases, foram desenvolvidos modelos com um conjunto de diretrizes sistematizando a metodologia

aplicada, que foram validados e que serviram como guia para a adequação de 6 processos modelados e automatizados conforme as práticas BPM.

As fases 5 e 6 ocorrem sem um modelo definido, sendo geralmente efetuadas através da troca de informações informal entre os gerentes e usuários que executam os processos com a equipe do EAPn. Dessa forma, o monitoramento abrange as necessidades percebidas pelos usuários finais e através delas são desenvolvidas soluções que resultem em melhorias e satisfaça as exigências impostas.

Sem implantar o monitoramento, os trabalhos realizados nas fases anteriores perderão utilidade com o transcorrer do tempo, visto a necessidade contínua de adequação dos processos com a realidade volátil das organizações. Por isso, a gestão BPM engloba ações de monitoramento, caracterizando o conceito de melhoria contínua.

A fim de compreender melhor o conceito de monitoramento e controle conforme a disciplina gerencial BPM, foco de interesse do trabalho, é apresentando o tópico a seguir, 3.2. Neste são definidos os conceitos e características intrínsecas ao tema, abordando formatos de implementação e seu impacto no refinamento para melhoria do desempenho dos processos e, por consequência, ao negócio como um todo.

### **3.2. Monitoramento**

Segundo o Guia CBOK, equiparando com a fase verificar (*Check*) do ciclo de vida PDCA, é atribuído ao monitoramento o objetivo de medir o desempenho real do processo em comparação ao desempenho esperado (CBOK, 2013). Para isso, monitorar se caracteriza por uma maneira de analisar, coletar e oferecer informações, as quais são relativas à estruturação dos dados das atividades de um processo.

#### **3.2.1. Importância do monitoramento para os modelos BPMM**

Modelos de Maturidade de processo de negócio, designados de BPMM (*Business Process Maturity Model*), possibilitam à organização avaliar a maturidade das práticas relacionadas à gestão por processos. Para isso, são definidos parâmetros que possibilitam identificar o quão avançadas as organizações estão na implementação de BPM.

Atualmente existe uma imensa quantidade de modelos BPMM. Segundo a ABPMP (*Association of Business Process Management Professional*) já foram identificados 150 modelos propostos que abordam diferentes pontos de vista (CBOOK, 2013). Uma deficiência desses modelos tem sido o foco simplificador de apenas uma dimensão para medir a maturidade do BPM e a falta de aplicações reais (Rosemann et al., 2004). Dessa forma, avaliar a maturidade e orientar as organizações para um cenário de processos mais estruturados e organizados se torna um desafio, porém, extremamente necessário.

Vários fatores são cruciais para o sucesso do BPM que podem complicar ou impedir sua implementação. Fatores críticos de sucesso comumente mencionados são: organização e mudança cultural, alinhamento da abordagem BPM às metas e estratégias corporativas, foco no cliente e seus requisitos, medição e melhoria de processos, a necessidade de uma abordagem estruturada ao BPM, compromisso de gerenciamento (Rosemann & Bruin, 2005).

A complexidade de aplicação das práticas BPM, contento todos os fatores mencionados, resulta na dificuldade de implementação e progressão para estágios mais altos de maturidade. Por isso, vem se discutindo diferentes formatos e definições para modelos que encontrem o equilíbrio entre a complexidade e simplificação (Rosemann & Bruin, 2005). Nesse contexto, surge a importância do monitoramento, definido por alguns autores como fator crucial de maturidade.

O modelo BPMM-Lee (Lee et al., 2007) aplicável ao escopo de BPM e processos, define como nível máximo de maturidade, as organizações na qual os processos são monitorados pré-ativamente e controlados, utilizando os dados de desempenho dos processos sistematicamente para melhorá-los. Complementando, o modelo de Escada de Maturidade do Processo (PML) (Harmon, 2004, 2007) define como nível máximo de maturidade quando os processos são medidos e gerenciados, contanto com uma equipe de melhoria de processos.

O artigo "*Developing a business process management maturity model: A study of 300 Iranian superior companies*" (Shafiei & Hajiheydari, 2014) apresenta uma revisão da literatura em busca de indicadores de maturidades BPM considerando os principais fatores para o sucesso de sua implementação, resultando em um guia para as organizações.

Como resultado obteve-se uma lista de 10 indicadores identificados como essenciais, dentre eles, o monitoramento (ver Figura 2). Este foi categorizado como

uma ferramenta associada à medição de índices e no processo de melhorias. O estudo prosseguiu através de um questionário distribuído entre as 300 empresas superiores iranianas para identificar se as características definidas cobrem os conceitos de maturidade utilizados.

Na visão das organizações, em termos de importância, a arquitetura de processos é considerada o indicador mais importante, enquanto que o monitoramento está dentre os menos importantes, na oitava colocação, segundo Shafiei & Hajiheydari, 2014.

Figura 2 - Peso e classificação de cada indicador de maturidade

Category	process		operational		Tool			organizational		
Category weight	0.323		0.189		0.294			0.194		
Dimensions of the category	Scope of implementation	Process architecture	Stakeholder communication and management	BPM governance	methods	BPM technology	monitoring	culture	people	Strategic alignment
Dimension weight	0.111	0.0889	0.0394	0.606	0.43	0.38	0.19	0.346	0.47	0.184
Dimension weight in its category	0.036	0.287	0.074	0.114	0.126	0.112	0.056	0.067	0.091	0.035
Dimension rank	9	1	6	3	2	4	8	7	5	10

Fonte: (Shafiei & Hajiheydari, 2014).

Essa pesquisa revela que as organizações não identificam dentre as suas práticas de gerenciamento por processos, ações de monitoramento, e não a utilizam como parâmetro essencial de maturidade. É enaltecido o contraste entre os conceitos teóricos apresentados pelos autores na revisão da literatura, dos quais, 19 entre os 27 artigos utilizados citam o monitoramento como um dos principais domínios e indicadores de maturidade.

Dentre as categorias da Figura 2, processo e ferramentas (incluindo monitoramento, tecnologias e métodos) são os mais importantes entre todas as categorias. Isso sugere que as organizações devem dar mais atenção a essas duas categorias tentando melhorá-las (Shafiei & Hajiheydari, 2014). De acordo com os resultados desta pesquisa, foi identificado que as políticas organizacionais são o maior obstáculo à movimentação em direção a níveis mais altos de maturidade. Comportamento humano, motivações e estrutura organizacional foram estabelecidas como fatores mais importantes do que obstáculos à tecnologia.

Para Rossemann e Bruin (2005), o fator que coloca as organizações em níveis baixos de maturidade em práticas BPM são fatores culturais e de desempenho. Dessa forma, uma das proposições de valor que um modelo deve oferecer, é servir como uma ferramenta de monitoramento contínuo no processo passando do estado em que está para o futuro.

### **3.2.2. O papel do monitoramento em sistemas BPM**

O conjunto de funções associadas a um sistema BPM geralmente são limitadas por: modelagem, execução, controle e monitoramento. A aplicabilidade destes em áreas tão diversas ocasionou na existência de diferentes abordagens na definição de sistemas BPM e principalmente na falta de consenso sobre as suas principais características e funções (Neely et al., 2007).

A diversidade de definições fornece diferentes perspectivas provenientes de características próprias a cada sistema. Dentre os autores, as principais características são definidas através de sistemas que permitem à empresa planejar, medir e controlar seu desempenho. Dessa forma, deve garantir que as iniciativas de vendas e marketing, práticas operacionais, recursos de tecnologia da informação, decisão de negócios e atividades das pessoas estejam alinhadas com os negócios estratégicos (Maisel, 2001, p. 12).

O artigo "*Towards a definition of a business performance measurement system*" (Neely et al., 2007) apresenta uma revisão da literatura buscando os papéis e funções que definem as principais características de um sistema BPM e como estas se relacionam com a medição de desempenho. Através das definições foram identificados os papéis que o sistema BPM desempenha em uma organização. O resultado é demonstrado na Figura 3.

Figura 3 - Principais funções dos sistemas BPM

	Atkinson (1998)	Atkinson <i>et al.</i> (1997)	Bititci <i>et al.</i> (1997)	Bourne <i>et al.</i> (2002)	Forza and Salvador (2000)	Gates (1999)	Ittner <i>et al.</i> (2003)	Kaplan and Norton (1996)	Kerssens-Van Drongelen and Fisscher (2003)	Lebas (1995)	Lynch and Cross (1991)	Maisel (2001)	McGee (1992)	Neely (1998)	Neely <i>et al.</i> (1995)	Otley (1999)	Rogers (1990)	Total percentage
Strategy implementation/execution	X					X	X	X			X	X	X	X		X	X	59
Focus attention/provide alignment	X						X					X	X	X		X	X	41
Internal communication (communicating performance, and priorities/objectives)			X		X				X		X			X	X	X		41
Measure performance/performance evaluation								X	X	X		X	X	X	X			41
Monitor progress	X	X									X	X	X	X	X			35
Planning				X								X	X				X	29
External communication									X				X	X				24
Rewards	X												X	X				18
Performance improvement														X	X			18
Managing relationships	X	X																12
Feedback					X												X	12
Double-loop learning	X									X								12
Strategy formulation														X				6
Benchmarking														X				6
Compliance with regulations														X				6
Control												X						6
Influence behaviour																X		6

Note: "X" is included if the definition provided refers to the role stated in the first column

Fonte: (Adaptado de Neely et al., 2007).

O resultado apontado pelo autor revela o pouco consenso quanto às características, no entanto, denota que 53% dos autores mencionam em suas definições a função de “*performance measures* (medidas de desempenho)”. De encontro com a temática proposta, 35% dos autores mencionam “*monitor progress* (monitoramento de progresso)” como uma das funções essenciais.

A função de medição de desempenho é levada em consideração pois o autor elege 5 categorias que englobam as funções mencionadas, sugerindo como ponto de partida para a seleção dos papéis dos sistemas BPM. A categoria relativa à medição de desempenho, considerada a principal pela recorrência de citação, é definida pela abrangência dos papéis de monitoramento de progresso, medição de desempenho e avaliação de desempenho (Neely et al., 2007). Dessa forma, é atribuído ao monitoramento como meio de efetuar a medição e avaliação de desempenho.

Os resultados da revisão da literatura apresentada ressaltam os objetivos propostos, demonstrando que o monitoramento é uma característica imprescindível para qualquer BPMS. Através de ações de aquisição, agrupamento, classificação, análise, interpretação e disseminação de dados, as organizações podem eficientemente medir seu desempenho, monitorar e compelir progresso, e estabelecer prioridades (Neely, 1998, p.5-6).

### 3.2.3. Ferramentas BI (*Business Intelligence*)

No contexto de sistemas BPM, encontra-se como componentes de software, recursos que permitem o monitoramento gráfico de indicadores de processo através de *dashboards*. Além da automação, a gestão por processos necessita de inteligência de negócio ampliando a capacidade analítica dos gestores. Neste cenário se inserem as ferramentas BI.

Essas ferramentas de inteligência de negócio são utilizadas para a análise de dados, gerando, através de consultas estruturadas, informações aos usuários. Os resultados apresentados em relatórios ou painéis possibilitam aos gerentes rapidez e assertividade na tomada de decisões visando, por exemplo, aumento da eficiência operacional.

Partindo de um conjunto grande de dados, torna-se um trabalho difícil encontrar informações relevantes com impactos táticos e estratégicos. Aliado ao desempenho do negócio, o BI proporciona um conjunto de ferramentas analíticas essenciais ao contraste de análise passado e futuro, verificando tendências, padrões e históricos de desempenho ao longo do tempo.

A associação entre BI e sistemas BPM possibilita estruturar e racionalizar as operações de negócios (Bach et al., 2007), de forma a complementar as tecnologias previstas em aplicações BPM. Associado ao ciclo de vida, as informações de desempenho geradas pelo BI tendem a melhorar o controle na medição de conformidade da operação, sendo um direcionador no programa de melhoria contínua da organização.

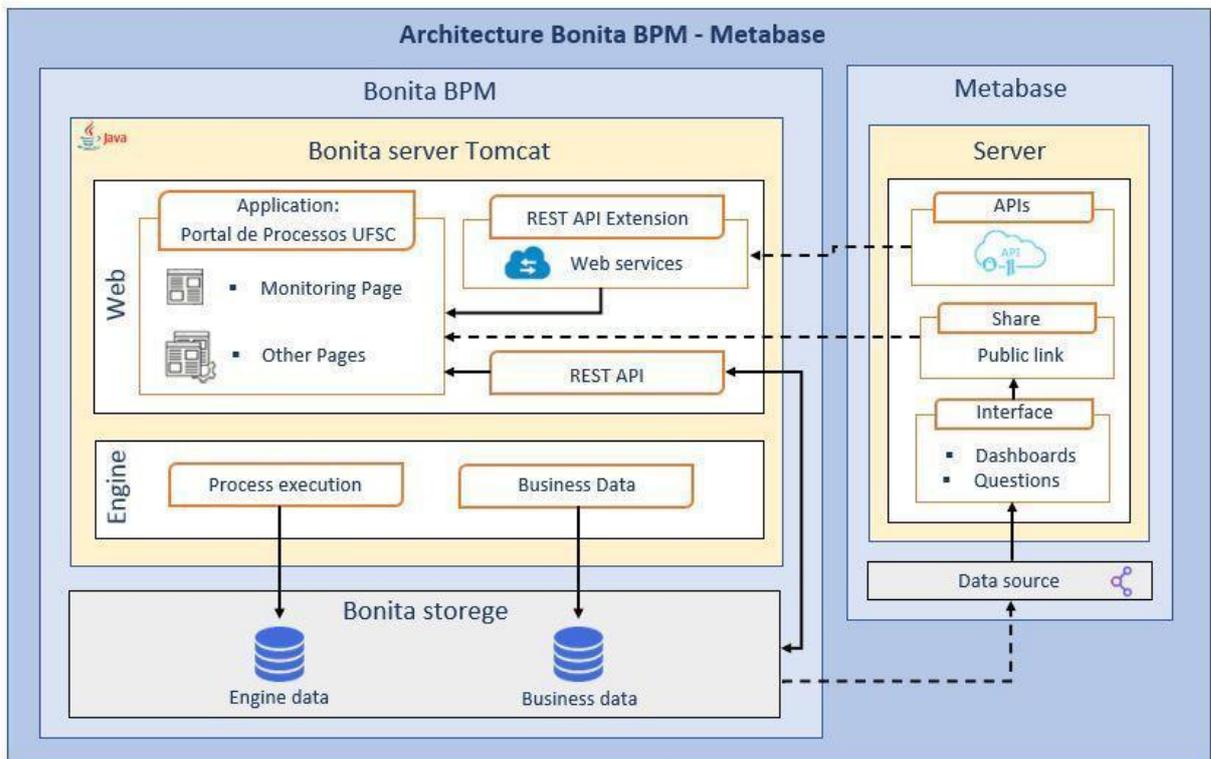
Informações de desempenho podem favorecer um novo tipo de dados para capacidade de reportes de BI, permitindo que gestores façam novos questionamentos sobre o desempenho operacional tanto histórico como atual (CBOOK,2013).

Este conceito demonstra como os recursos de monitoramento para análise gerencial estão associados ao BPM. Mesmo sendo recente, estas tecnologias estão cada vez mais integradas ao cenário de processos de negócios.

#### 4. TEORIAS GERAIS E FERRAMENTAS UTILIZADAS

Em decorrência das limitações do Bonita BPM, não oferecendo em sua versão gratuita ferramentas específicas de monitoramento, desenvolveu-se uma solução baseada na integração de *dashboards* construídos através do software BI (*Business Intelligence*) Metabase com páginas desenvolvidas com o software Bonita. A arquitetura de comunicação dos componentes elementares utilizados no desenvolvimento da solução é apresentada na Figura 4 e ambos são descritos no decorrer do capítulo.

Figura 4 – Arquitetura de comunicação entre Bonita BPM e Metabase



Fonte: Autoria própria.

O Portal de Processos da UFSC é um aplicativo *web* constituído por uma coleção de páginas, criadas através de dois recursos do Bonita: páginas e aplicativos. Para promover um ambiente único de gerenciamento sobre os processos, facilitando a operabilidade para os usuários envolvidos, a solução apresentada destina-se a expandir o Portal de Processos utilizando os recursos mencionados. Dessa forma, a arquitetura da página de monitoramento desenvolvida será restrita a utilização das ferramentas disponíveis no ambiente UI Designer.

Os mecanismos implementados são condicionados à utilização da estrutura do banco de dados fornecida pelo Bonita e do modelo de dados e propriedades definidas pelo EAPn na automação dos processos. Para a concepção da solução, informações definidas nos *dashboards* são correlacionadas às informações do respectivo processo automatizado que provém os dados para estruturação do painel. A interdependência entre as informações origina restrições que devem ser seguidas para funcionamento correto da aplicação, porém não resulta em nenhuma limitação de desenvolvimento.

Dentre todos os usuários com acesso ao Portal de Processos, a página de monitoramento irá se restringir ao uso dos quais detém o papel de dono de processo. Essa designação é destinada aos usuários que têm atribuído a si a responsabilidade geral pelo gerenciamento ponta-a-ponta do processo de negócio. Assim, estes são os responsáveis por garantir que o processo atenda às expectativas de desempenho estabelecidas.

As ferramentas e técnicas apresentadas destinam-se em proporcionar uma solução que demande implementações orientadas à utilização dos recursos disponíveis no Bonita BPM, fornecendo um ambiente simples para construção e gerenciamento dos *dashboards*.

#### **4.1. Metabase: Recursos para desenvolvimento**

O Metabase é uma ferramenta de inteligência de negócios destinada à coleta, processamento e análise, transformando grandes quantidade de dados em informações úteis aos interesses da organização. Com ele é possível fazer a leitura e interpretação de um banco de dados, possibilitando análises através de recursos visuais e interfaces interativas.

A utilização do Metabase originou-se de uma pesquisa de softwares *open source* para análise de dados disponíveis gratuitamente no mercado. O uso do software é baseado em aspectos específicos para utilização no EAPn e das ferramentas disponíveis em sua versão gratuita para desenvolvimento da solução proposta. As principais características consideradas são:

- **Integração com banco de dados:** Permite integração com 20 sistemas de gerenciamento de banco de dados. Entre eles, o h2, utilizado no

ambiente de desenvolvimento local no Bonita BPM e MySQL utilizado no ambiente de produção do EAPn.

- **Limites de usuários:** Não possui limites de usuários. Restrição recorrente encontrada em outros softwares do mesmo segmento, limitando os usuários para acesso.
- **Limites de painéis:** Não possui limites de painéis. Podem ser construídos quantos *dashboards* forem necessários.
- **Simplicidade:** Disponibiliza ao desenvolvedor um ambiente intuitivo e flexível com ferramentas gráficas de visualizações pré-configuradas, possibilitando a criação gráfica de *dashboards* sem o desenvolvimento de códigos. Além disso, disponibiliza um ambiente para programação utilizando a linguagem SQL (*Structured Query Language*) para consultas mais avançadas.
- **Suporte:** Principalmente por ser *open source*, possui grande quantidade de informação em comunidades, além de manuais e guias para desenvolvimento e utilização dos recursos disponibilizados.

O Metabase oferece um ambiente de desenvolvimento que permite a manipulação de tabelas integradas a diferentes bancos de dados, utilizando a estrutura padrão do local de origem. Para simplificar e promover análises rápidas de fluxos de dados complexos são fornecidas ferramentas de sumarização, filtros e ordenação, potencializando a eficiência de geração de resultados.

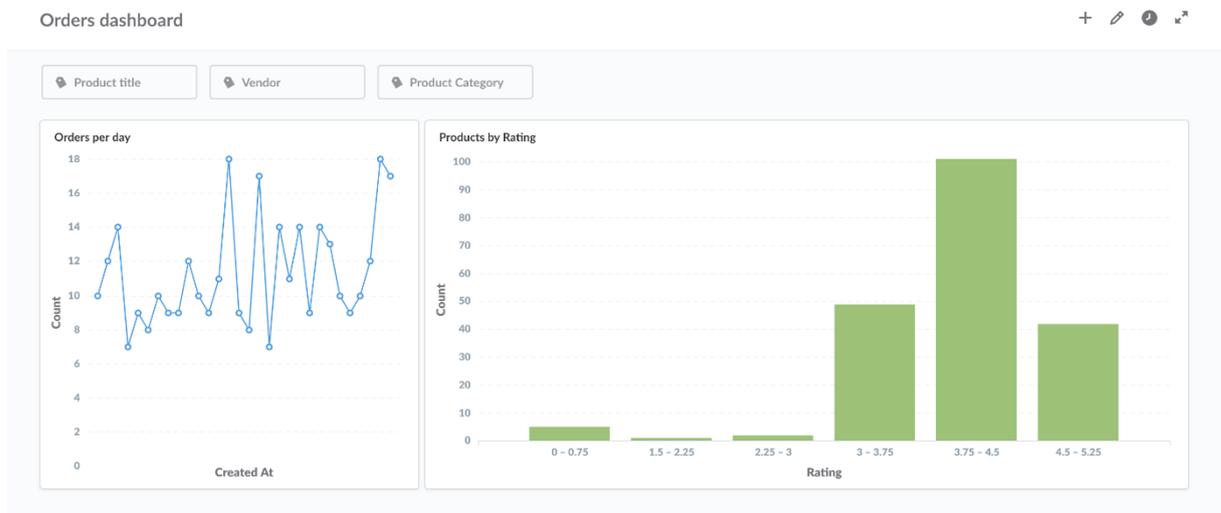
#### 4.1.1. Perguntas e painéis

Métricas e indicadores são informações quantitativas ou qualitativas provenientes de perguntas e suas respectivas respostas. Neste contexto, utilizar o recurso para criação de perguntas consiste em empregar técnicas para processamento de dados a fim de se obter um determinado resultado, isto é, uma resposta.

O Metabase possui 3 ambientes para a construção de perguntas: simples, personalizado e nativo. Estes são utilizados de acordo com o nível de complexidade da pergunta a ser elaborada, podendo ir de simples métricas de contagem até perguntas mais complexas que exigem a junção de tabelas, vários estágios de filtragem ou agregação e recursos gráficos avançados.

Os resultados podem ser configurados para serem exibidos graficamente, utilizando por exemplo, os recursos de visualização disponibilizados. Com a criação de uma coleção de perguntas, estas podem ser associadas a um painel (*dashboard*) possibilitando a visualização de diferentes resultados em uma única interface.

Figura 5 – Dashboard genérico desenvolvido no Metabase



Fonte: Autoria própria.

A disposição dos elementos pode ser configurada e dimensionada, resultando no layout mais adequado à aplicação. A atualização dos dados é ajustada em intervalos de tempo pré-definidos, além disso, dispõe da funcionalidade para a utilização de filtros que são executáveis a todas as perguntas.

#### 4.1.2. Consultas SQL

Para os objetivos definidos no trabalho, será empregado o ambiente para criação de perguntas através de consultas nativas em SQL (*Standard Query Language*) considerando o sistema de gerenciamento de bancos de dados MySQL. A

linguagem permite manipular e executar consultas sobre bases de dados relacionais, atuando como interface de acesso aos dados.

Partindo das tabelas associadas à base de dados, são aplicados *queries* de consultas para, por exemplo, selecionar os dados de uma coluna específica da tabela. Estas podem ser definidas por um comando, que além de selecionar possibilita: inserir, alterar, atualizar, filtrar, agrupar, ordenar, entre outras operações. De forma geral, qualquer comando estará relacionado a armazenar, manipular ou recuperar dados.

Dessa forma, as informações são geradas através do relacionamento entre as tabelas, linhas e colunas mediante as operações de consultas aplicadas, estruturando o resultado conforme a finalidade associada.

#### **4.1.3. Links públicos**

Este recurso permite o compartilhamento público dos painéis, permitindo o acesso à visualização por qualquer pessoa através de uma URL (*Uniform Resource Locator*) específica. A atribuição ou ocultação de filtros podem ser configurados via URL, permitindo ao usuário final maior controle sobre o painel que será integrado ao seu aplicativo.

O link é acessível a qualquer usuário, independentemente se este tiver autenticação no Metabase. Ao ser incorporado em um aplicativo ou ter acesso a URL, o usuário não possui automaticamente acesso ao próprio Metabase, impossibilitando-o de acessar demais informações do banco de dados. Dessa forma, a autenticação para visualização dos painéis e segurança é controlada pela aplicação que utilizará os links públicos.

A solução proposta no trabalho é fundamentada na utilização deste recurso, permitindo incorporar os painéis nas páginas *web* desenvolvidas no Bonita, inserindo o contexto de navegação que possibilite a sua abertura e visualização.

#### **4.2. Bonita BPM: Recursos de desenvolvimento**

O Bonita é uma plataforma DPA (*Digital Process Automation*) que permite criar aplicativos com base em processos, disponibilizando os recursos tradicionais de sistemas BPM. É caracterizado por alinhar práticas de modelagem, operação,

automação e melhorias de processos de negócios com tecnologias de inteligência operacional para otimizar a orquestração<sup>3</sup> e padronizar a experiência do usuário.

O conceito DPA é uma especificação das tecnologias de um BPMS. No caso do Bonita, é associado principalmente ao uso de aplicativos com foco em melhorar a eficiência e experiência do cliente (Clair, 2019). Independentemente da nomenclatura utilizada e do contexto associado às ferramentas destinadas a aspectos da automação, visam promover principalmente: fluxos de trabalhos automáticos, interfaces interativas e inteligentes, colaboração entre usuários, adaptação à mudança, rastreabilidade e transparência nas interações feitas entre usuário e sistema.

As ferramentas e aspectos do Bonita BPM são relativas à utilização da edição Community versão 7.6.3. Uma visão geral de cada recurso associado ao desenvolvimento da solução é apresentada nos subtópicos abaixo, especificando as particularidades de interesse.

#### **4.2.1. Estrutura de dados**

O Bonita possui uma aplicação de gerenciamento que utiliza 2 bancos de dados relacionais RDBMS (*Relational DataBase Management System*) para armazenar dados do Bonita Engine e do Modelo de Dados Corporativos. Para cada modelo de dados e suas respectivas tabelas são atribuídos os seguintes propósitos:

- O modelo de dados do Bonita Engine armazena informações sobre as definições, configurações e histórico de execução dos processos implementados, usuários e informações de configurações da plataforma Bonita.
- O modelo de dados corporativos ou BDM (*Business Data Model*) armazena os dados corporativos compartilhados por processos e aplicativos. É definido como um conjunto de objetos de negócio, tornando-se variáveis da instância de um processo.

---

<sup>3</sup> Orquestração refere-se à relação entre atividades e eventos do processo, promovendo a compreensão da sequência de trabalho.

Para conectar-se com o banco de dados o Bonita oferece os drivers JDBC para conectividade com os sistemas de gerenciamento H2, PostgreSQL e MySQL, executadas dentro de um servidor Apache Tomcat. Para as etapas de desenvolvimento e testes é utilizado o sistema H2, vinculado automaticamente pelo Bonita em suas configurações padrões de instalação. Para as aplicações em servidores de produção é utilizado o MySQL.

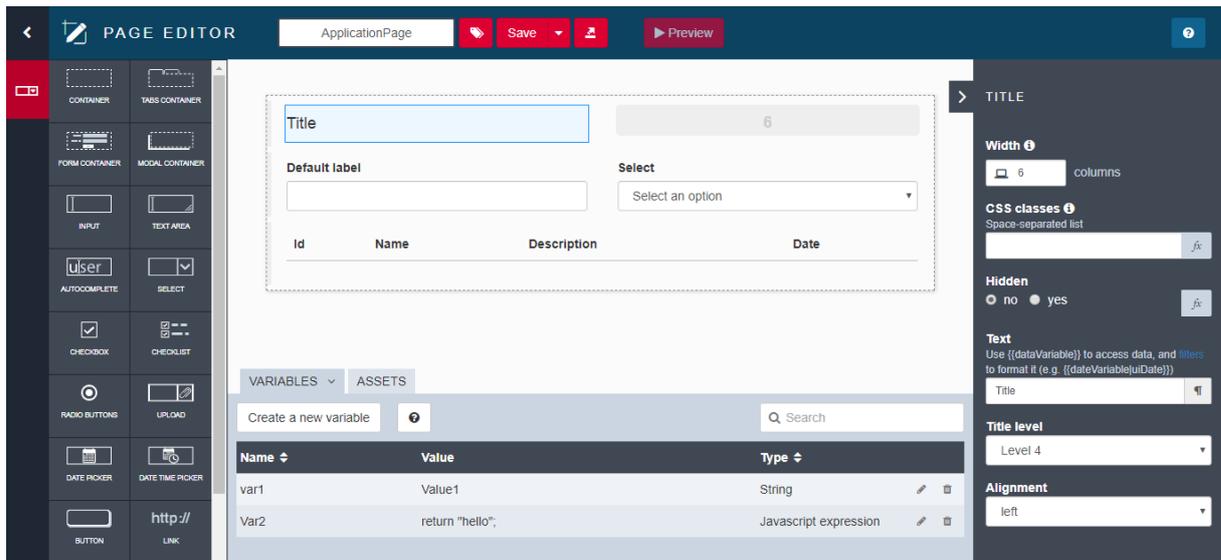
#### 4.2.2. Páginas e aplicativos

O UI designer permite que os desenvolvedores projetem páginas com interfaces de usuário responsivas, comumente chamadas de páginas de aplicação ou páginas personalizadas. Essas denominações são associadas à sua finalidade, referente ao uso em aplicativos ou à customização do Bonita Portal. Este recurso destina-se à programação *front-end* da aplicação, configurando a interface gráfica através de elementos visuais CSS, *bootstrap* e *AngularJS* e definindo as operações resultantes da interação com o usuário.

O conteúdo da página é definido com a utilização de elementos de interação (*widgets*) disponibilizados para uso através da ação pegar e soltar, tais como: botões, textos, tabelas, links e imagens. Os *widgets* são compostos de propriedades que são vinculados a variáveis de dados que definem suas operações e características gráficas, fornecendo funcionalidades específicas ao usuário. As variáveis utilizadas podem ser do tipo string, JSON, API externa, expressão Javascript ou um parâmetro URL, possibilitando a utilização de diferentes estruturas de dados e implementar funcionalidades complexas.

Caso seja necessário utilizar elementos diferentes dos disponibilizados, o desenvolvedor pode criar um *widget* personalizado definindo sua aparência e comportamento através das linguagens HTML, AngularJS e JavaScript. Após a criação, os *widgets* personalizados ficam disponíveis junto aos demais para utilização em qualquer página.

Figura 6 - Ambiente UI Designer



Fonte: Autoria própria.

As páginas *web* são importadas no Bonita Portal para que sejam incorporadas a um aplicativo, permitindo o acesso para navegação através de um endereço URL. Um aplicativo é um ambiente customizado para um perfil de usuário específico visando uma necessidade comercial, possibilitando a interação com dados e processos de negócios de maneira mais eficiente que o Bonita Portal padrão. Dessa forma, o aplicativo é responsável por definir os requisitos de navegação entre as suas páginas associadas.

O construtor de aplicativos é uma interface onde é especificado o conjunto de páginas para criar um mapeamento entre os recursos, a estrutura de menu a ser empregada, o contexto e navegação atribuída a cada página e informações gerais de descrição. A estrutura de navegação destina-se a especificar para quais páginas as seções dos menus devem apontar e a hierarquia a ser utilizada.

#### 4.2.3. Extensão de API REST

API REST é um modelo de arquitetura de comunicação que permite acessar dados com solicitações HTTP, seguindo um conjunto de restrições a serem usadas para fornecer APIs de acesso a um serviço Web.

Através das APIs é possível acessar todos os objetos do Bonita, como processos, tarefas e usuários, através de métodos de requisição para efetuar as ações de criar, recuperar, atualizar ou excluir. Utilizando as requisições e invocando os

métodos específicos a cada API pode-se obter informações para definir fluxos de trabalho, iniciar um processo, listar informações para uso em páginas e formulários ou até mesmo para recuperar resultados em aplicativos externos.

O Bonita disponibiliza uma biblioteca de APIs com categorias específicas a cada recurso, com chamadas e respostas estruturadas definidas em seu mecanismo, especificados por um conjunto de URL de solicitações. O conteúdo de retorno é configurado através de parâmetros e filtros de consulta e para cada objeto contido na resposta são associados todos os seus atributos.

Utilizar esses recursos com respostas previamente estruturadas torna-se um problema em aplicações mais complexas, quando se faz necessário o uso de informações correlacionadas que necessitem diferentes solicitações de APIs e quando não se utilizam todas as informações contidas no conteúdo da resposta. Dessa forma, para se obter as informações desejadas pode ser necessário desenvolvimentos em *front-end* (formulários, páginas e interfaces visíveis aos usuários).

Para possibilitar ao desenvolvedor a utilização de APIs com um conteúdo de resposta mais adequada à aplicação, é utilizada a ferramenta de criação de Extensão de API REST. Esse recurso é utilizado para consultar os dados de mecanismo ou de negócios do Bonita criando uma nova estrutura de requisição e resposta, com todas as operações em *back-end*. Os arquivos são programados através da linguagem Groovy e são inseridos para utilização no Bonita Portal.

Suas funcionalidades não se restringem à utilização dos dados provenientes do Bonita, mas também de dados de sistemas de terceiros (banco de dados externos, serviços da web, etc.). O processo de integração e utilização de informações do Metabase é proveniente de solicitações por APIs realizadas exclusivamente através dessa ferramenta, visto a necessidade de segurança no processo de autenticação para acesso ao Metabase.

#### **4.2.4. Parâmetros**

Os parâmetros são variáveis de no máximo 256 caracteres que podem ser do tipo texto, booleano, inteiro ou double. São utilizados no Bonita Studio para armazenar um valor associado à definição do processo e se aplicam a todos os casos, diferentemente das variáveis de negócio que são associadas aos casos de um processo, e podem assumir um valor diferente a cada caso iniciado.

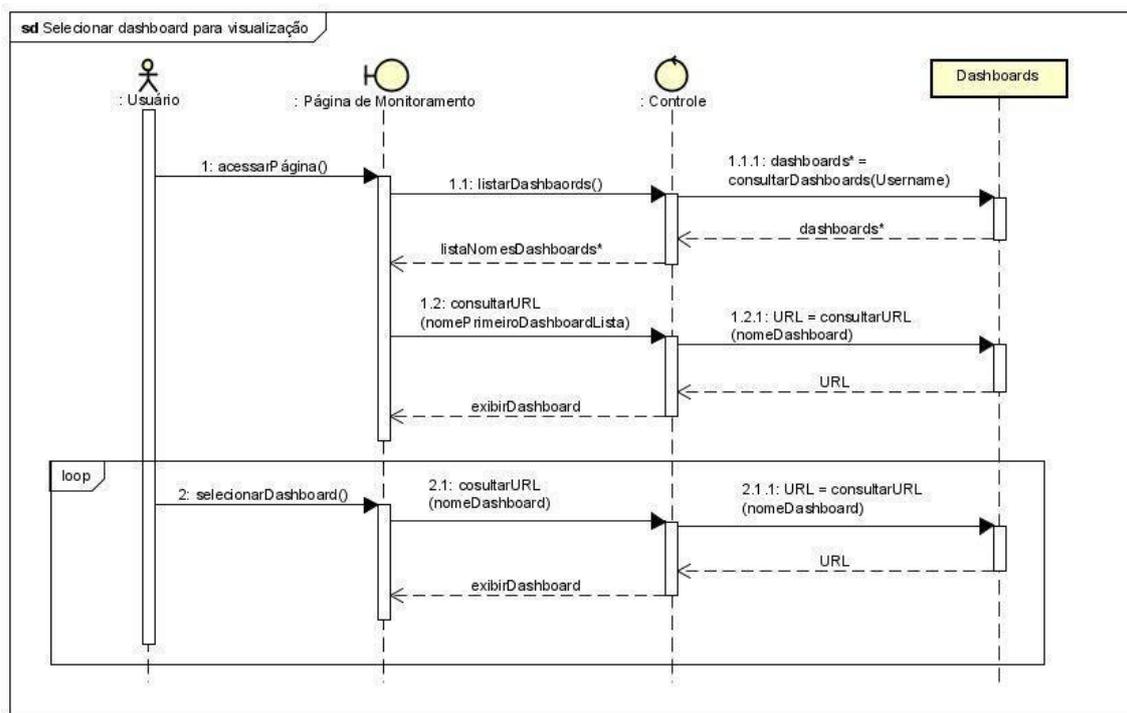
Através do identificador de definição do processo é possível recuperar, através de solicitação via API, o valor dos parâmetros associados e utilizá-los nas páginas. Dessa forma, torna-se possível anexar variáveis ao processo que possibilitem correlacionar com os respectivos usuários que possuem acesso à visualização dos *dashboards*. O exemplo mencionado é relativo à solução abordada pelo trabalho, seus detalhes são descritos no próximo capítulo.

## 5. DESENVOLVIMENTO

Neste capítulo são apresentados os conceitos de implementação utilizados na elaboração do projeto de integração. Serão abordados os procedimentos práticos empregados nas etapas de desenvolvimento, explicitando os métodos aplicados e especificações de execução para satisfazer os objetivos e requisitos do sistema.

A estrutura implementada na programação da página de monitoramento é intrínseca às funcionalidades que possibilitam ao dono de processos selecionar os *dashboards* no qual este é dono de seus respectivos processos. A relação lógica e o comportamento definido pelo conjunto de interações entre os componentes do sistema e usuário são representados no diagrama de sequência em UML (ver figura 7).

Figura 7 - Diagrama de sequência UML da página de monitoramento

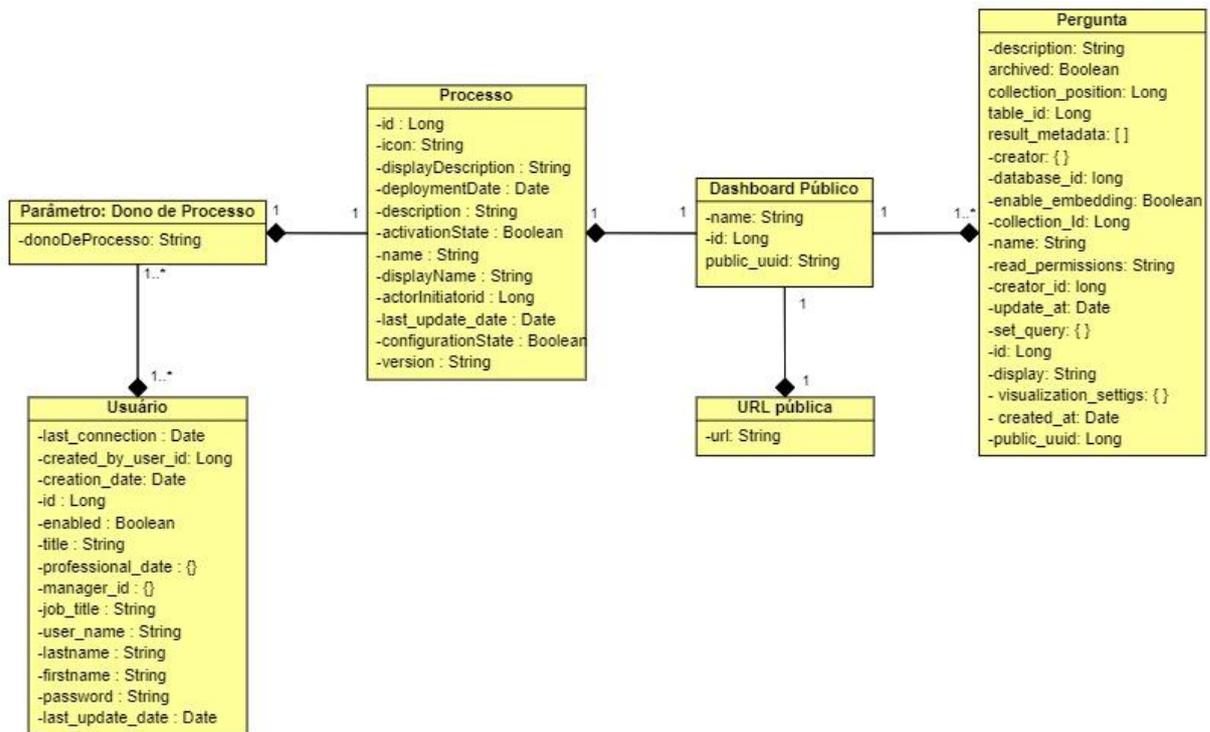


Fonte:

Autoria própria.

Para ilustrar e definir a estrutura de relacionamento entre o conjunto de objetos, componentes da aplicação e os atributos que os compõem é utilizado o diagrama de classes UML (ver figura 8).

Figura 8 – Diagrama de classe UML



Fonte: Autoria própria.

As operações apresentadas nos diagramas definem de forma clara a visão geral sobre os aspectos que constituem a solução implementada. Dessa forma, estes se configuram como os requisitos do sistema, impondo as condições necessárias para desenvolvimento, conforme apresentadas nos subtópicos deste capítulo.

### 5.1. Configuração dos *dashboards*

O Metabase não disponibiliza APIs que possibilitem ao usuário identificar os dados internos empregados na criação das perguntas dos *dashboards*, impossibilitando reconhecer as respectivas tabelas utilizadas no banco de dados de origem. Dessa forma, não é possível correlacionar o processo automatizado no Bonita com o seu respectivo *dashboard* do Metabase, tornando-se necessário definir manualmente uma informação que seja comum em ambos os softwares.

Dentre as informações disponíveis, representadas no diagrama da Figura 8 pela classe “Processo”, optou-se por utilizar o nome do processo (*name*). Essa variável é definida diretamente pelo usuário, diferentemente de algumas que são definidas automaticamente pelo Bonita, como *id*. Além disso, a nomeação dos

processos segue um padrão definido pelo EAPn, garantindo ao desenvolvedor maior praticidade por ser previamente conhecida e de fácil acesso.

Assim sendo, o nome do *dashboard* deve corresponder ao nome do processo automatizado de origem dos dados. Através das APIs do Metabase é possível recuperar as informações de definição dos *dashboards* e com a utilização da Extensão de API REST, utilizar os resultados no ambiente do Bonita correlato aos processos.

## **5.2. Extensão de API REST: Requisição de dados dos *dashboards***

Para obter as informações dos *dashboards* no Bonita BPM, bem como as respectivas URLs públicas, é necessário criar uma extensão de API REST. Através de métodos de requisição HTTP é realizada a ação sobre um dado recurso, neste caso, é executado a autenticação de usuário no Metabase e a requisição para retorno de dados.

As extensões são estruturadas de acordo com as especificações do Bonita, contendo dois tipos de arquivos:

- Contém um arquivo desenvolvido em programação Groovy. Este define a lógica a ser implementada, contendo as requisições e operações sobre os dados.
- Contém um arquivo de propriedades. Este define as informações de chamada da API desenvolvida pelo Bonita, contendo principalmente nome e URL da extensão e o mapeamento entre a URL e o script Groovy desenvolvido.

### **5.2.1. Arquivo Groovy**

O código da extensão deve seguir o padrão exigido para que seja interpretado corretamente para implantação no Bonita. Para isso, são utilizadas bibliotecas disponíveis pelo Bonita em sua documentação. Essas bibliotecas configuram um conjunto de funções que possibilitam operações de controle e a chamada de dados internos ao Bonita, proporcionando as implementações pelo usuário.

Como a extensão desenvolvida é utilizada para obtenção de dados de uma plataforma externa, torna-se somente necessário declarar bibliotecas com classes operacionais para utilização, além de bibliotecas Java para possibilitar as requisições

HTTP e Groovy para formatação e interpretação de dados no formato Json (ver figura 9).

Figura 9 - Bibliotecas para a Extensão de API

```
import javax.servlet.http.HttpServletRequest
import javax.servlet.http.HttpServletResponse
import org.bonitasoft.web.extension.rest.RestAPIContext
import org.bonitasoft.web.extension.rest.RestApiController
import org.bonitasoft.web.extension.rest.RestApiResponse
import org.bonitasoft.web.extension.rest.RestApiResponseBuilder
import org.slf4j.Logger
import org.slf4j.LoggerFactory
import java.io.File
import groovy.json.JsonSlurper
import groovy.json.JsonBuilder
```

Fonte: Autoria própria.

As requisições de API do Metabase são executadas seguindo os métodos Java, executando uma instância de conexão para cada solicitação. As propriedades da solicitação HTTP são definidas pelos métodos utilizados e pelo cabeçalho instanciado que contém os parâmetros necessários para a chamada e a obtenção da resposta desejada. Os cabeçalhos podem incluir credenciais, parâmetros para seleção do tipo de conteúdo e *cookies* de sessão, compondo o corpo da solicitação.

Os recursos alvos de uma requisição são referenciados por uma identificação de recursos uniforme (URI), sendo um dos tipos de identificação para localização de recursos na Web os endereços URL (*Uniform Resource Locator*). O Metabase disponibiliza a documentação da API para acesso e compartilhamento de seus recursos, exibindo sua identificação URL, definição e métodos disponíveis. A Tabela 1 mostra a API utilizada e suas especificações para requisição.

Tabela 1 – Estrutura de requisição HTTP das APIs do Metabase

API	Método	Parâmetros
.../api/session	POST	Username, password
.../api/dashboard/public	GET	-

Inicialmente é realizada a autenticação do usuário para verificar se este possui acesso aos dados internos do Metabase para obtenção das informações dos

*dashboards*. Através do método POST são enviados os dados das credenciais de autenticação, conforme os métodos definidos na Figura 10.

Figura 10 - Requisição HTTP POST para autenticação de usuário

```
def post = new URL("http://processos.ufsc.br/api/session").openConnection();
def message = '{"username":"username", "password":"password"}'
post.setRequestMethod("POST")
post.setDoOutput(true)
post.setRequestProperty("Content-Type", "application/json")
post.getOutputStream().write(message.getBytes("UTF-8"));
```

Fonte: Autoria própria.

Ao iniciar a conexão é obtida, como retorno, uma instância que representa a conexão com o objeto referido pela URL destino, criada a partir da invocação do método `openConnection()`. Para definir o envio de dados ao servidor é necessário informar que a conexão de URL deverá ser utilizada para saída, conforme definido em `setDoOutput()`, com a variável boolean `true`. Dessa forma, torna-se necessário por meio do fluxo de saída da conexão o envio de um corpo para solicitação de autenticação, definido através do método `get.OutputStream().write(content)`.

Como retorno se obtém o código de status da solicitação HTTP, dado por uma resposta do tipo `int`. Ele retornará 200 para autorização confirmada e 401 para autorização negada, caso ocorram problemas na requisição e nenhum código puder ser discernido a partir da resposta o valor de retorno é -1, configurando uma resposta HTTP inválida.

Após a validação da autorização de usuário, torna-se possível fazer as requisições das informações dos *dashboards*. Através do método GET, executando essencialmente os mesmos procedimentos definidos na solicitação POST. Conforme a Figura 11, é possível verificar os procedimentos e principalmente a definição da propriedade de solicitação através da chave obtida no método POST, indicado pelo método `setRequestProperty(chave,valor)`.

Figura 11 - Requisição HTTP GET para retorno de informações do *dashboard*

```
def jsonSlurper = new JsonSlurper()
def obj1 = jsonSlurper.parseText(post.getInputStream().getText())

def getMeta = new URL("http://processos.ufsc.br/api/dashboard/public").openConnection();
getMeta.setRequestMethod("GET");
getMeta.setRequestProperty("Content-Type", "application/json")
getMeta.setRequestProperty("X-Metabase-Session", obj1.id)
```

Fonte: Autoria própria.

O próximo passo é verificar a resposta da requisição. Se o recurso solicitado foi processado e retornado com sucesso, espera-se obter um elemento no formato JSON constituído de objetos que contêm os atributos: name, id e public\_uuid.

### 5.2.2. Arquivo de propriedades

As informações relacionadas à extensão da API REST asseguram a identificação da extensão nos diferentes módulos do Bonita, constituindo informações para definição, operação e permissões, sendo elas:

- **contentType**: definição do tipo de recurso a ser implementado;
- **name**: nome da extensão da API REST. Obrigatoriamente deve começar com “custompage\_”;
- **displayName**: nome para exibição no Bonita Portal;
- **description**: incluir opcionalmente uma descrição;
- **apiExtensions**: definir a lista de APIs do Bonita utilizadas na extensão desenvolvida, separadas por vírgula. Opcional, caso nenhuma tenha sido utilizada;
- **method**: definição do método de operação para requisição da extensão implementada;
- **pathTemplate**: definir o caminho de URL para chamada da extensão de API;
- **classFileName**: declarar o arquivo groovy associado;
- **permissions**: declarar as permissões para autorização de acesso aos recursos para usuários.

O arquivo resultante com as devidas propriedades declaradas é demonstrado na Figura 12.

Figura 12 - Arquivo de propriedades da Extensão de API

```

userInformationRestAPIExtension.method=GET
userInformationRestAPIExtension.permissions=organization_visualization
name=custompage_userInformationRestAPIExtension
userInformationRestAPIExtension.classFileName=Index.groovy
userInformationRestAPIExtension.pathTemplate=userInformation
displayName=User information REST API Extension
contentType=apiExtension
description=Query Bonita BPM Engine to retrieve user information
apiExtensions=userInformationRestAPIExtension

```

Fonte: Autoria própria.

Os dois arquivos desenvolvidos devem ser implantados no Bonita Portal, adicionando um novo recurso do tipo extensão de API. Com isso é possível a sua utilização no Bonita UI Designer executando as requisições conforme as definições estabelecidas no arquivo de propriedades para incorporação dos dados de retorno em páginas e formulários.

### 5.3. Estruturação da página de monitoramento

Os elementos que constituem a página devem permitir ao utilizador dono de processo fundamentalmente visualizar *dashboards*, derivando desta, as funcionalidades do sistema. As operações são definidas em etapas de verificação e consulta de dados, com o objetivo de:

1. Identificar se o usuário é dono de processo;
2. Identificar os processos disponíveis, cujo o respectivo dono de processo tem acesso aos *dashboards*;
3. Selecionar o processo para visualização do seu respectivo *dashboard*;
4. Exibir informações de interesse sobre o processo e o *dashboard* selecionado.

A estrutura da página é composta por elementos relacionados às operações descritas, e as ligações entre estes devem possibilitar a realização da lógica de funcionamento da figura 7. Características associadas à página que decorrem de requisitos de integração ao Portal de Processos não serão abordadas, como por exemplo, o elemento associado à operação de *logout*. Este elemento foi desenvolvido

também para as demais páginas do Portal de Processos e está disponível para utilização, não exigindo esforço de implementação.

### 5.3.1. Layout

O *layout* da página segue os mesmos requisitos das demais páginas do Portal de Processos, caracterizada por uma interface simples e intuitiva, permitindo a adaptação aos novos conceitos e facilitando a procura de informações por parte dos utilizadores. Neste primeiro momento, é caracterizado por elementos básicos para funcionamento, gerando um modelo inicial para testes e validação.

A Figura 13 ilustra o layout visual implementado, contemplando os elementos gráficos visíveis ao usuário após a confirmação de seu papel de dono de processo.

Figura 13 - Layout da página web de monitoramento



Fonte: Autoria própria.

Exceto pelo elemento para visualização do conteúdo dos *dashboards*, os demais elementos são disponibilizados por padrão no UI Designer, compondo sua

estrutura básica em HTML e de controle em JavaScript. Para estes, torna-se necessário apenas configurar propriedades de funcionamento, definindo as variáveis de entrada e saída de dados, e aparência. A definição associada a cada elemento é:

- **Cabeçalho:** Elemento do tipo *text*. Exibir informações da organização e operações da sessão.
- **Título:** Elemento do tipo *title*. Exibir o título geral da página.
- **Seletor:** Elemento do tipo *select*. Listar os dashboards respectivos aos processos que o usuário possui permissão para visualização.
- **Informações Gerais:** Elemento do tipo *text*. Exibir informações gerais do usuário, processo e dashboard selecionado.
- **Rodapé:** Elemento do tipo *text*. Exibir informações complementares.

Para visualizar o conteúdo dos *dashboards* é utilizado o elemento em HTML, `IFrame`. Basicamente, o `IFrame` (*Inline Frame*) permite a incorporação de uma página secundária da *web* dentro da página principal que o contém. Isso significa que o contexto de navegação utilizado na página principal não interfere no conteúdo do `IFrame`, já que o seu conteúdo é de propriedade da página secundária.

Por não ser um elemento padrão do Bonita é necessário criar um elemento customizável através do recurso *custom widget*. A ferramenta disponibiliza o ambiente para criação no UI Designer, permitindo ao usuário configurar:

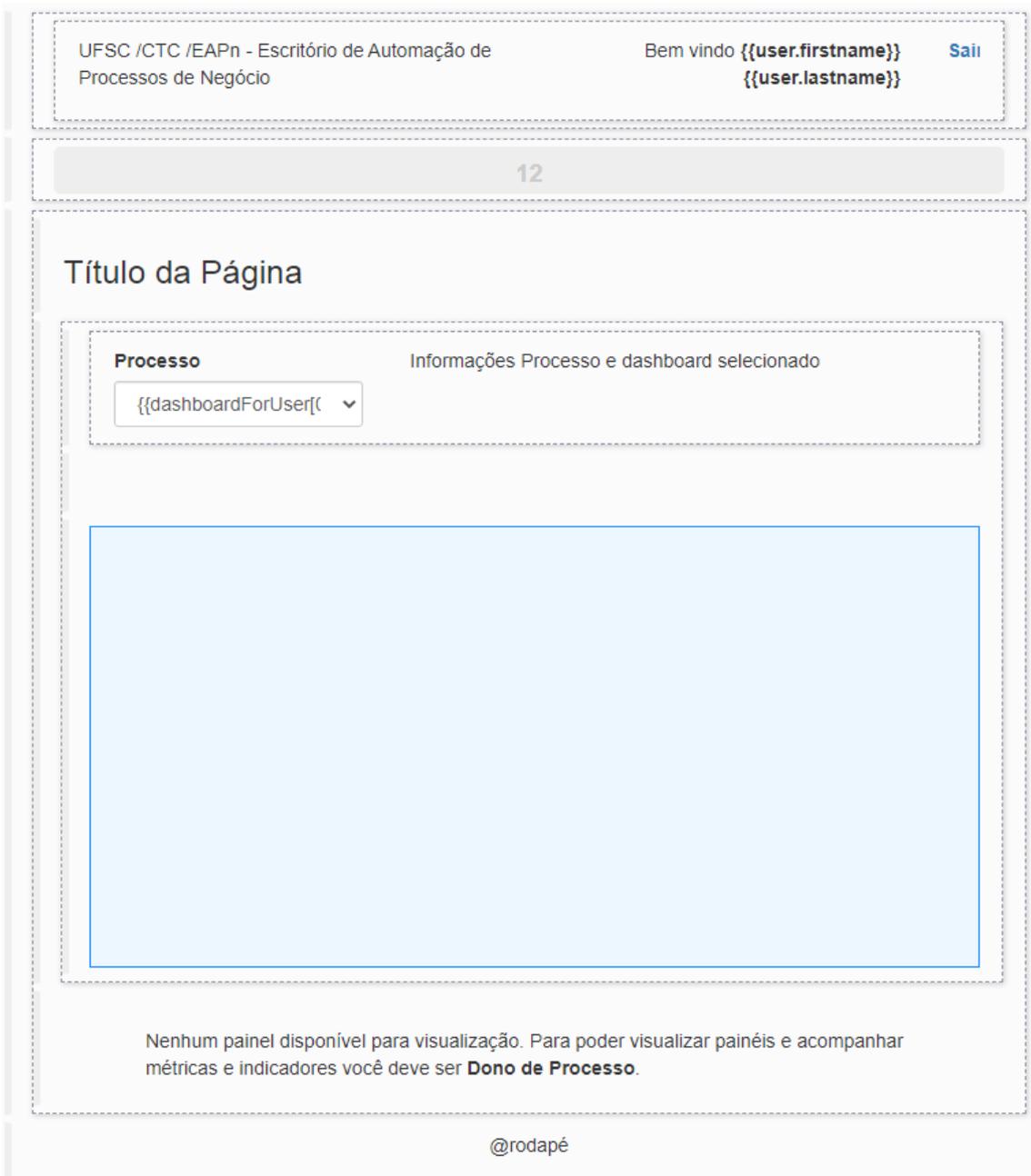
- **Description:** Inserir uma descrição para identificar o elemento.
- **Template:** Modelo que descreve a estrutura HTML do conteúdo.
- **Controller:** Definir a funcionalidade/comportamento em JavaScript.
- **Assets:** Incluir bibliotecas JavaScript, definições de CSS ou imagens.
- **Angular modules:** Incluir módulos AngularJS caso necessite usar serviços ou diretivas que não estão associadas por padrão.
- **Properties:** Definir propriedades dinâmicas para utilizar variáveis que dependem de páginas, formulários ou da definição do usuário, personalizando sua aparência ou comportamento.

Após realizar as configurações, o elemento se torna disponível para utilização em qualquer página. O `IFrame` é redimensionado de acordo com seu conteúdo que pode vir a ocupar um espaço de corpo maior que um *frame* de tela padrão,

apresentando um tamanho estático e barras de rolagem para deslocamento de seu conteúdo interno.

Em aspectos gerais, o posicionamento e *design* buscam oferecer uma arquitetura de informação que defina a organização em uma sequência lógica predominantemente estática. O resultado pode ser visto na Figura 14, exibindo os elementos de acordo com a interface do ambiente de desenvolvimento do UI Designer. Algumas nomeações estão em formato genérico para facilitar a correlação com a estrutura anatômica apresentada na Figura 13.

Figura 14 - Elementos da página no ambiente de desenvolvimento UI Designer



Fonte: Autoria própria.

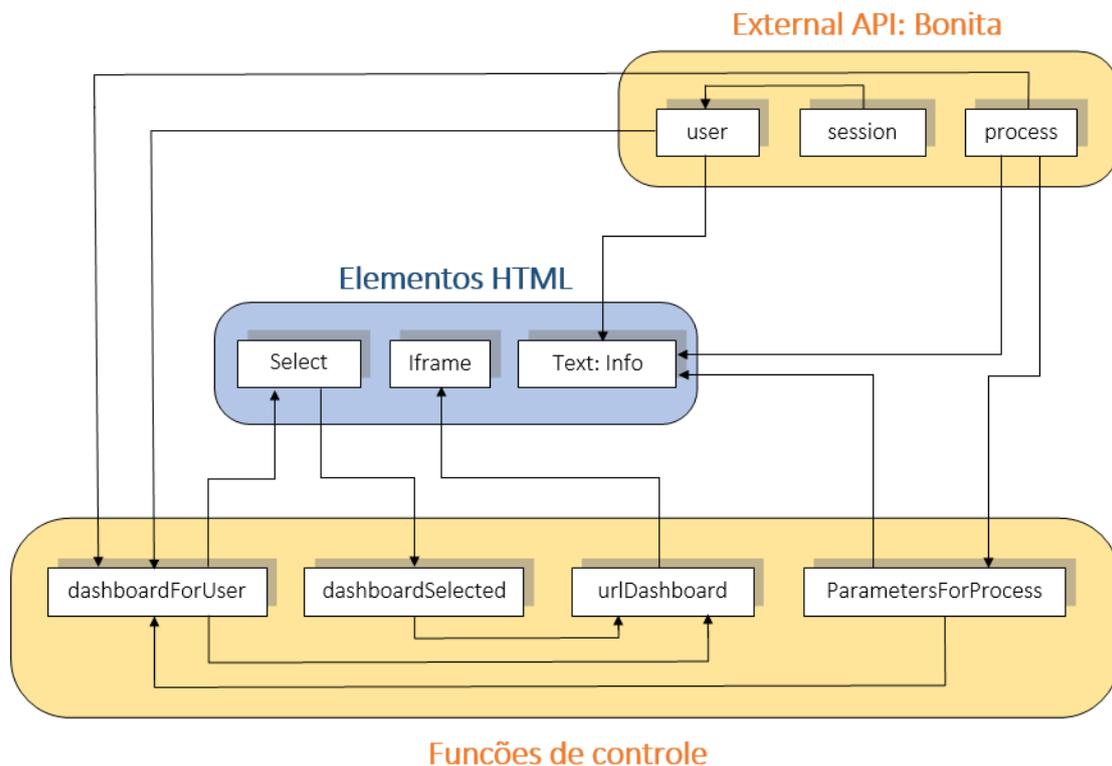
### 5.3.2. Funcionalidades operacionais

Para definir as funcionalidades da página, variáveis e scripts de comando são utilizados para manipular dados corporativos ou para conter informações de operações que determinam como os elementos da página se comportam. A conexão entre as variáveis de operação e elementos HTML dinâmicos são demonstrados na

Figura 15, definindo as interfaces de entrada e saída de dados e a dependência entre operações.

Para simplificar a ilustração foram mencionadas somente as operações relativas ao funcionamento da página após a confirmação do usuário como dono de processo.

Figura 15 - Relação entre as variáveis utilizadas no escopo da página



Fonte: Autoria própria.

As demais variáveis e as representadas externas aos elementos HTML da figura acima são instanciadas no escopo da página, na qual são programadas suas respectivas operações, ambas indicadas na Tabela 2.

Tabela 2 - Variáveis definidas no escopo da página de monitoramento.

Nome	Tipo	Operação
dashboardForUser	Javascript expression	Listar os dashboard do usuário da sessão
dashboardSelected	String	Armazenar o dashboard selecionado
enablePage	Javascript expression	Retornar a habilitação de acesso a página
parametersForProcess	Javascript expression	Retornar o parâmetro dono de processo
process	External API	Retornar os processos disponíveis
session	External API	Retornar dados da sessão
urlDashobard	Javascript expression	Listar as Urls de dashboard disponíveis
user	External API	Retornar informações do usuário da sessão

Os procedimentos são executados de modo sequencial, garantindo que ao iniciar qualquer operação esta tenha à disposição os dados necessários para processamento. Para qualquer sistema que dependa de requisições HTTP, utilizando especificações do protocolo 1.1, que não há paralelização de requisições, exige boas práticas de programação, demandando principalmente:

- Diminuir ao máximo o número de requisições.
- Diminuir o volume de dados nas requisições, requerendo somente os de interesse para as demais operações.

Essas práticas visam a performance da aplicação, garantindo que futuramente quando o fluxo de dados seja maior que o atual não ocorra perdas demasiadas de desempenho.

### 5.3.2.1. Requisição da Extensão de API REST

Através do recurso operacional definido na variável `dashboardForUser`, usufruindo do retorno das demais operações JavaScript, é realizada a requisição da extensão de API, conforme definido no Capítulo 5.2. A requisição segue o mesmo padrão HTTP para execução do método GET, com a informação de retorno do identificador público do *dashboard* (`public_uid`) é possível definir a sua respectiva URL pública.

Seguindo o padrão do Metabase conforme seu domínio de execução no EAPn, a URL que será executada no IFrame, é caracterizada pelo formato:

- [http://dashboard.processos.ufsc.br/public/dashboard/public\\_uuid](http://dashboard.processos.ufsc.br/public/dashboard/public_uuid)

### **5.3.2.2. Controle de acessibilidade**

O recurso operacional relacionado a variável enablePage está associado a uma limitação descrita no seção 1.4, na qual a URL da página definida para a aplicação pode ser executada de forma independente no navegador. Assim, caso algum usuário tenha informações do endereço este o poderá acessar.

Mesmo os dados no contexto apresentado não sendo sigilosos, decidiu-se implementar um filtro de usuário para efetuar o controle de acessibilidade à página. Dessa forma, ao acessar a página e ser verificado através das operações definidas em enablePage que o usuário não é dono de processo todo o conteúdo é desabilitado e é exibido um texto informativo indicando o ocorrido.

## **5.4. Incorporação da página de monitoramento ao Portal de Processos**

A página desenvolvida é inserida no contexto da aplicação das páginas do Portal de Processos para executar os processos automatizados. A página é integrada como um recurso independente podendo estar associada a um ou mais aplicativos, demandando um mapeamento específico a cada aplicação.

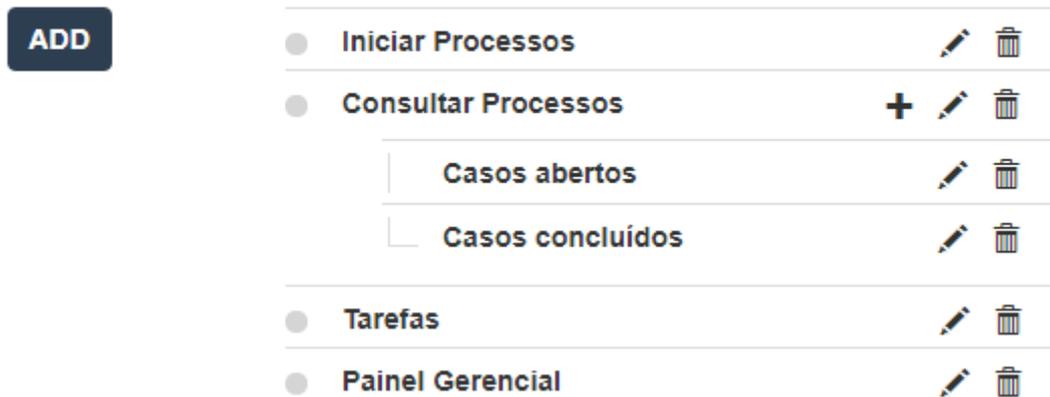
Para criar o vínculo é necessário definir o caminho para acesso ao recurso da página, estabelecendo o contexto para a invocar através do domínio definido para a aplicação. O acesso criado ao Portal de Processos é referenciado por:

- <https://processos.ufsc.br/bonita/apps/eapnPortal/PainelGerencial/>

A URL é especificada na estrutura de navegação do aplicativo, compondo o menu de nível superior do Portal de Processos. Porém, por ser um recurso independente o acesso à página pode ocorrer de maneira individual, não necessitando estar obrigatoriamente associado ao aplicativo gerado.

Figura 16 - Estrutura de navegação do Portal de Processos da UFSC

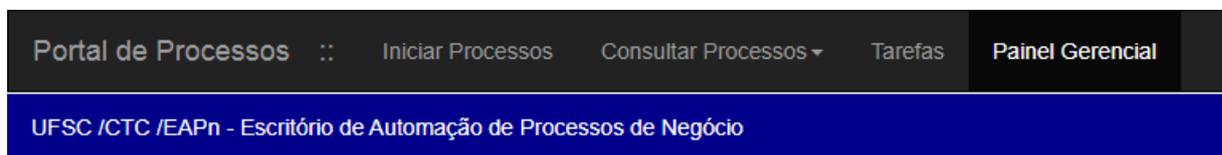
## Navigation



Fonte: Autoria própria.

Para a navegação definida é gerada a estrutura do Portal de Processos da Figura 17.

Figura 17 - Menu do Portal de Processos da UFSC



Fonte: Autoria própria.

### 5.4.1. Configuração do menu de navegação

Esse tópico apresenta as configurações necessárias para alteração da configuração de estruturação do menu de navegação da aplicação. Por ser uma alteração que necessita da modificação de um arquivo interno de instalação do Bonita, esta não foi implementada no desenvolvimento do projeto. Porém, para fins de documentação e conhecimento sobre a aplicação, os resultados são relatados.

A página de monitoramento é de interesse apenas dos usuários donos de processo, não havendo necessidade de estar disponível para acesso no menu aos demais usuários, conforme respaldado na seção 5.3.2.2. Por padrão, ao inserir a página no contexto de navegação da aplicação não é possível, através dos recursos

disponibilizados, configurar alterações na estrutura do menu que variem de acordo com o usuário, desabilitando quando necessário o botão de acesso.

O conteúdo das páginas associadas à aplicação são vinculadas para exibição a uma página principal estruturada internamente pelo Bonita, responsável também por construir a estrutura de apresentação do menu. O arquivo de interesse para as alterações se encontra localizado na pasta bonita-default-layout.zip, denominado:

- widgets-9a3031d7fd8c5934d6fac86470ab0272af06d82c.min.js

Esse arquivo JavaScript executa os mecanismos para identificar através de APIs internas as informações de definição das aplicações, páginas associadas e a estrutura do menu de navegação. A lógica aplicada retorna todas as páginas definidas na navegação pelo usuário, enviando os dados diretamente para estruturação do menu em HTML e CSS, gerando a estrutura estática apresentada na Figura 17.

Para possibilitar a estruturação do menu condicionada ao fator usuário, foi necessário alterar o código padrão do arquivo inserindo um filtro para estabelecer o controle de acessibilidade. Dessa forma, ao invés de enviar diretamente todas as páginas definidas para navegação, anteriormente é realizado operações que permitem:

- I. Identificar o usuário atual da sessão.
- II. Consultar o parâmetro dono de processo, de cada processo disponível.
- III. Cruzar as informações para verificar se o usuário é dono de processo.

Se for verificado que o usuário não é dono de processo é retirado dos dados de retorno utilizados para estruturação do menu as respectivas informações da página de monitoramento. Caso contrário, o procedimento padrão deve ser seguido, visto que não é necessário desabilitar o acesso à página, disponibilizando ao usuário todos os itens definidos na navegação.

## **5.5. Desenvolvimento dos *dashboards***

Neste ponto, a interface de monitoramento caracterizada pela estrutura de integração entre os softwares está concluída. Para operacionalizar o projeto resta construir o *dashboard* respectivo a determinado processo automatizado, para que seja possível através do Portal de Processos acompanhar os indicadores de desempenho.

Seguindo os recursos do Metabase descritos no tópico 4.1, a implementação do *dashboard* decorre da definição dos indicadores e sua implementação no formato visual. Serão atribuídas representações gráficas disponíveis no ambiente de desenvolvimento, selecionadas empiricamente de acordo com o respectivo indicador visando o modelo gráfico que facilite o seu acompanhamento.

### 5.5.1. Indicadores de desempenho implementados

A definição dos indicadores de desempenho está alinhada com a classificação do CBOK no gerenciamento de desempenho de processos em conjuntura ao cenário atual dos processos automatizados da instituição. Os indicadores selecionados se enquadram de modo estratégico ao gerar informações macro.

Partindo de uma visão ampla sobre os processos é empregado indicadores genéricos que podem ser aplicados aos diferentes processos automatizados de modo padronizado. Essa abordagem não considera dados internos sobre o modelo de negócio, isto é, informações específicas ao processo que definem seu fluxo, formulários e contratos<sup>4</sup>.

O objetivo é gerar uma estrutura primária a fim de identificar informações de nível superior facilitando o entendimento do negócio por parte dos donos de processos, propiciando a familiarização com a interface de *dashboards*. Entende-se que os indicadores utilizados irão auxiliar na contextualização do atual cenário do processo, sendo que atualmente não se tem de forma clara medidas de tempo e quantidade.

Mediante os requisitos de implementação, os indicadores considerados são resultantes de duas perspectivas de estudo:

- I. Revisão sistemática da literatura desenvolvida no EAPn sobre métodos aplicados na avaliação de processos de negócio e indicadores utilizados ou propostos para medição de desempenho.
- II. Pesquisas exploratórias para identificação de softwares BPMS (incluindo o Bonita BPM) que incluem tecnologias de monitoramento e avaliação de

---

<sup>4</sup> O contrato especifica as informações que o processo requer para ser iniciado e executar sua lógica de negócio.

processos, identificando os indicadores para medição de desempenho utilizados em sua interface.

A partir da concatenação dos resultados, os indicadores genéricos selecionados para implementação são:

- Tempo médio de execução do processo.
- Tempo médio de espera do processo.
- Quantidade de casos de processo abertos e concluídos por mês.
- Tempo médio de execução das tarefas do processo.
- Quantidade de tarefas concluídas por mês.
- Quantidade das últimas tarefas executadas em casos de processos concluídos.

Analisando os dados disponíveis e os objetivos propostos, além de indicadores foi adicionada métricas ao *dashboard*. Caracterizado por uma informação bruta sem correlação a demais parâmetros, as métricas não propiciam uma visão ampla e direcionada quanto os indicadores, mas são fundamentais ao prover resultados quantitativos simples.

As métricas definidas que inclusive serviram para composição dos indicadores selecionados, são:

- Casos de processos abertos.
- Casos de processos concluídos.
- Quantidade de tarefas concluídas.

### **5.5.2. Programação SQL para consulta de dados**

Mediante a lista de indicadores e métricas são realizadas as consultas SQL sobre a base de dados a fim de se obter os resultados correspondentes, conforme a estrutura de consulta SQL descrita no tópico 4.2.1. Por se tratar de implementação puramente prática de programação não será abordado todas as consultas realizadas. Assim, serão descritas apenas as características consideradas possibilitando o entendimento sobre os resultados obtidos.

Conforme ilustrado na figura 4, o sistema de armazenamento do Bonita BPM é composto por duas bases de dados, neste caso será utilizado no desenvolvimento

o *Engine Data* visto que nenhum indicador necessita de dados do negócio. Esta base de dados é composta por 82 tabelas, sendo que destas, foram utilizadas somente 3:

- `process_instance`: Dados sobre as instância de processos abertas.
- `arch_process_instance`: Dados sobre as instância de processos arquivadas.
- `arch_flownode_instance`: Dados sobre os nó de fluxo(tarefas) dos processos arquivados.

Para garantir a consistência das informações dos indicadores são consideradas apenas as instâncias de processos que foram arquivados e concluídos corretamente. Caso contrário, considerando dados de uma instância aberta, não se pode garantir que a instância será concluída corretamente visto que pode ocorrer falhas de operação nas tarefas futuras, fazendo que o dado que estava sendo considerado deixe de ser válido.

Um exemplo de uma consulta simples realizada no ambiente de criação de perguntas nativas em SQL e o seu resultado gráfico pode ser visto na figura 18.

Figura 18 - Exemplo de consulta SQL e resultado gráfico



Fonte: Autoria própria.

Neste caso, as operações empregadas foram de seleção, junção, filtragem, agrupamento e ordenação. Para as demais consultas as operações se restringem majoritariamente a estas mencionadas, divergindo de complexidade conforme torna-se necessário correlacionar dados de diferentes tabelas para definição de condições.

Para limitar o escopo de dados do *dashboard* é inserido um filtro para definição de um intervalo de tempo. Dessa forma, pode-se concentrar a análise sobre um

determinado período facilitando na visualização e compreensão através do refinamento dos resultados exibidos.

Uma amostra do *dashboard* contendo o filtro seletor de datas e o resultado de algumas das implementações de indicadores e métricas é visto na figura 19.

Figura 19 - Resultado gráfico do *dashboard*



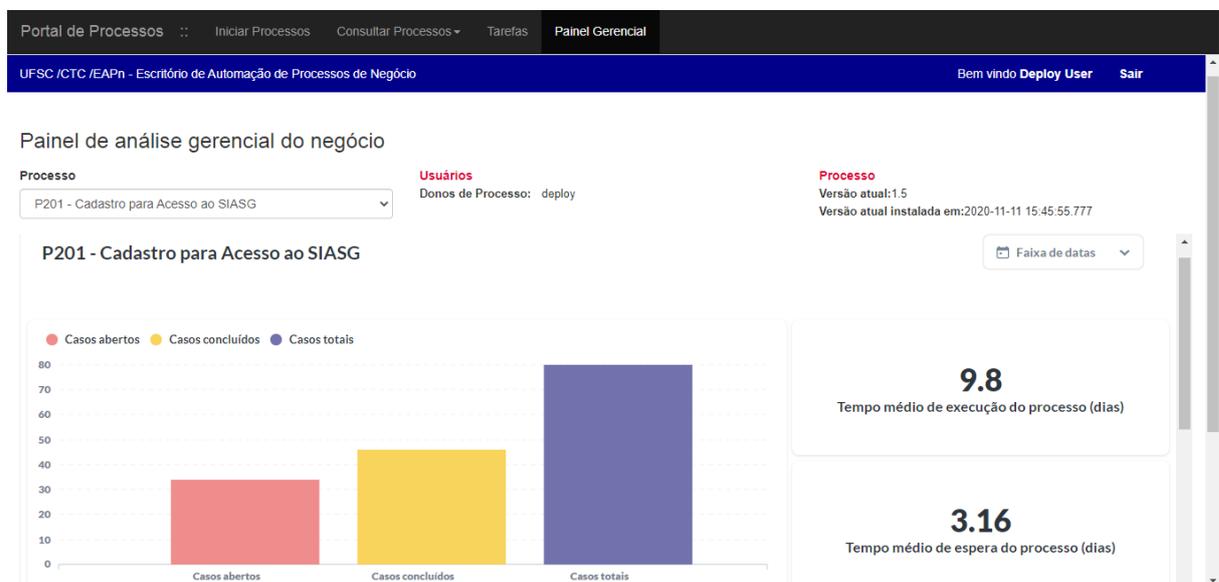
Fonte: Autoria própria.

Com a estruturação da página de monitoramento gerando a integração entre os sistemas, a configuração e desenvolvimento do *dashboard*, toda a infraestrutura é finalizada. Dessa forma, o *dashboard* desenvolvido estará disponível no Portal de Processos para visualização dos seus respectivos donos de processo.

## 6. RESULTADOS

Como resultado foi obtida a página para monitoramento dos processos automatizados em estágio operacional completa, pronta para iniciar testes e validação com os usuários. A estrutura apresentada demonstra sua aplicação em um ambiente de produção, integrado ao servidor online do Portal de Processos da UFSC (ver figura 20).

Figura 20 - Página de monitoramento no Portal de Processos da UFSC



Fonte: Autoria própria.

As informações resultantes são oriundas de um dos processos automatizados pelo EAPn. O nome do dono do processo real é ocultado e é apresentado o acesso via usuário *deploy* que contém as mesmas permissões de dono de processo. A página de monitoramento com os respectivos *dashboards* integrados, independente do processo de utilização, seguirá o mesmo padrão da Figura 20.

Além da página, o trabalho resultou na obtenção de experiência na utilização do recurso de Extensão de API REST, até então não utilizada para nenhuma aplicação no EAPn. Dessa forma, para implementações futuras que necessitem da obtenção de dados de sistemas externos através de APIs, o presente trabalho juntamente aos códigos desenvolvidos poderão ser utilizados como fundamentação.

## 7. CONCLUSÃO

A integração da página de monitoramento ao Portal de Processos da UFSC irá promover aos donos de processos e desenvolvedores do EAPn uma poderosa ferramenta para análise e posteriormente melhoria dos processos automatizados. Através de um sistema padronizado e analítico será possível com maior propriedade e formalidade elaborar e definir estratégias.

O gerenciamento de desempenho irá proporcionar o direcionamento de todos os esforços em prol da efetividade das atividades do processo. Dessa forma, identificar se os objetivos traçados pela organização estão de acordo com a execução, garantindo a máxima eficiência. A utilização dos *dashboards* se apresenta bastante vantajosa devido ao aumento do número de casos de processos executados mensalmente, possibilitando da forma mais eficiente possível a visualização de indicadores de desempenho.

Utilizando uma plataforma única para gerenciamento e execução dos processos, o trabalho empregado para controle dos dados da organização é minimizado, gerando melhor usabilidade aos usuários. Especificamente aos usuários desenvolvedores do EAPn, já que controle de recursos da página e acessibilidade para visualização dos *dashboards* é feito através do Bonita BPM. Dessa forma, além da criação de *dashboards* através do Metabase, nenhum conhecimento mais avançado é necessário, centralizando as operações ao Bonita.

A continuidade do projeto poderá ser dada pelo desenvolvimento de indicadores específicos a cada processo, incluindo informações do negócio. Além disso, o projeto pode ser estendido futuramente, agregando tecnologias para a geração de relatórios inteligentes, identificação de variações de desempenho automáticas para análises pontuais e demais serviços de inteligência. Dessa forma, facilitando a análise dos usuários envolvidos e a identificação de melhorias para os processos automatizados.

Fora do escopo previsto ao *Business Intelligence* de análises gerenciais com perspectiva histórica, o projeto pode ser estendido ao monitoramento de atividades do negócio em execução. Para isso, dentro do contexto de BPM, se encontram as ferramentas BAM (*Business Activity Monitoring*). Estas são aplicadas na análise e apresentação de informações em tempo real sobre as atividades de negócio em execução.

De modo geral, através do BAM é possível monitorar a situação atual do negócio. Com a implementação do BI, torna-se possível entender os resultados a fim de conceber uma visão estratégica e entendimento sobre a organização gerando parâmetros que possibilitem atuar sobre o BAM, complementando a análise.

## REFERÊNCIAS

- BACH, M. P.; CURKO, K.; RANDONIC, G. **Business Intelligence and Business Process Management in Banking Operations**. Conference: Information Technology Interfaces, 2007.
- DE MORAIS, R. M.; DE PÁDUA, S. I. D.; COSTA, A. L. **An analysis of BPM lifecycles: From a literature review to a framework proposal**. Business Process Management Journal, [S. l.], v. 20, n. 3, p. 412–432, 2014.
- GALIMBERTI, M. F.; MARIANI, A. C.; CORDEIRO, V. C.; TRIDAPALLI, J. V.; DE PIERE, E. R.; PERTERS, S. **Modelagem e Automação do Processo de Negócio “Afastamento Exterior (Inclusive com Diárias e Passagens)”**. Relatório Técnico INE 002. Universidade Federal de Santa Catarina, 2019.
- GONÇALVES, H. P. **Guia para Modelagem e Automação de Processos de Negócios Acadêmicos: estudos de casos com processos da UFSC**, 2016.
- HARMON, P. **Evaluating an Organization’s Business Process Maturity**. [S.l.], Vol. 1, 2004.
- HARMON, P. **Business Process Change**. Burlington, MA, USA. 2007.
- LE CLAIR, C. **RPA, DPA, BPM, And DCM Platforms: The Differences You Need To Know**. Apply Forrester’s Automation Framework To Process Automation. [S. l.], 2019.
- LEE, J.; LEE, D.; SUNGWON, K. **An Overview of the Business Process Maturity Model (BPMM)**. Seoul, Korea, 2007.
- MAISEL, L. S. **Performance Measurement Practices: A Long Way from Strategy Management**. The Balanced Scorecard Report. p 12, 2001.
- MOORE, C. **BPM CBOK: Guia para o Gerenciamento de Processos de Negócio**. Corpo Comum de Conhecimento ABPMP. Brasil, 2013.
- NEELY, A. D. **Measuring Business Performance: Why, What, and How**, The Economist and Profile Books Ltd. London, UK. p 5-6. 1998.

NEELY, A.; FRANCO-SANTOS, M.; KENNERLEY, M.; MICHELI, P.; MARTINEZ, V.; MASON, S.; MARR, B.; GRAY, D. **Towards a definition of a business performance measurement system**. Cranfield University, Cranfield, UK. 2007.

ROSEMANN, M.; BRUIN, T. de. **Application of a Holistic Model for Determining BPM Maturity**. [S. l.], 2005.

ROSEMANN, M.; DE BRUIN, T.; HUEFFNER, T. **A Model for Business Process Management Maturity**. ACIS 2004 Proceedings. 6, Queensland University of Technology, 2004.

SHAFIEI, A.; HAJIHEYDARI, N. **Developing a business process management maturity model: A study of 300 Iranian superior companies**. [S. l.], 2014.