



INSTITUTO DE CIÊNCIAS EXATAS E NATURAIS  
FACULDADE DE COMPUTAÇÃO  
CURSO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

LUCAS RODRIGUES GUIMARÃES

## **Informatizando o fluxo de portarias na UFPA com o sistema SIPRO**

Belém  
2018

LUCAS RODRIGUES GUIMARÃES

## **Informatizando o fluxo de portarias na UFPA com o sistema SIPRO**

Trabalho de Conclusão de Curso apresentado ao curso de Bacharelado em Ciência da Computação, como parte dos requisitos necessários à obtenção do título de Bacharel em Ciência da Computação.

Orientador: Claudomiro de Souza de Sales Junior

Coorientador: Lucas de Melo Silva

Belém  
2018

*Dedico esse trabalho à minha família que sempre me apoiou no que necessário, em especial a minha mãe e minha vó, que não pouparam esforços para que eu concluísse esse projeto e que sempre dedicaram a vida para me trazer alegria.*

## Agradecimentos

Gostaria de agradecer a minha mãe que sempre me sustentou e nunca hesitou em me dar do bom e do melhor, a minha vó que sempre cuidou de mim e faz uma comida muito boa.

Ao meu orientador que sempre me ajudou no desenvolvimento do trabalho, mesmo estando ocupado.

O meu coorientador e a equipe de TI da PROGEP que sempre foram muito amigáveis e receptivos, me ajudando nos trabalhos do estágio e participando no meu desenvolvimento profissional e pessoal.

E por último, mas não menos importante: aos meus amigos, que não me ajudaram em nada no escopo do trabalho, mas são meus amigos.

*"Deus é TOP" - Neymar*

## Resumo

Ao longo dos anos, a informática vem sendo aperfeiçoada para melhorar a vida das pessoas. A informatização de processos é um exemplo desse fenômeno: processos e tarefas muito burocráticas vêm se tornando cada vez mais fáceis e rápidas de serem feitas graças à informática. O objetivo desse trabalho é de mostrar um exemplo prático de como a informatização de tarefas pode melhorar um ambiente de trabalho com o novo sistema de portarias da Universidade Federal do Pará. Esse sistema foi criado em 2017 com o objetivo de suprir várias demandas internas dos servidores que cadastram e assinam portarias e demandas externas, de servidores que acessam portarias, como: padronização de documentos, centralização de armazenamento, buscas de portarias usando parâmetros diversos e a implantação de um módulo de assinatura digital, integrado a este sistema para permitir assinaturas de documentos mais fáceis e automatizadas. Com a implantação deste sistema, os resultados obtidos foram muito positivos: o uso de papel e tinta para a impressão de portarias foi reduzido em cerca de 66% e a expectativa é que essa redução chegue em 100%. Além disso, a assinatura de portarias agora ficou muito mais simples, confiável e segura com a implantação do módulo de assinatura digital.

## Abstract

Over the years, computation is being constantly being improved in order to make people's lives easier. The computerization of processes is an exemple of this phenomenon: very bureaucratic processes and tasks are becoming easier and quicker to be done thanks to computation. The main purpose of this work is to present a practical example of how computerization of processes can improve a workplace with the new system of ordinances of the Federal University of Pará. This system was created in 2017 in order to satisfy many internal demands from employees that are directly involved with ordinance registration and signature, along with external demands from employees that access ordinances, such as: document padronization, storage centering, ordinance queries using many parameters and the implantation of a digital signature module, integrated to this system to allow easier and automated ordinance signatures. With the implantation of this system, the results were very satisfying: the use of paper and ink from the printer to print ordinances has been reduced by 66% and the expectation is that this redution goes up to 100%. Futhermore, the ordinance signing has become much more simpler, trustable and safe thanks to the new digital signature module.

## Lista de ilustrações

Figura 1 – Fluxo antigo de portarias . . . . .	17
Figura 2 – Máquina de estados dos status de portarias . . . . .	20
Figura 3 – Estrutura textual de uma portaria . . . . .	22
Figura 4 – Primeira versão da abstração de um “Interessado”. . . . .	23
Figura 5 – Segunda versão da abstração de um “Interessado”. . . . .	23
Figura 6 – Representação simplificada da entidade “TemplatePortaria” no banco de dados. . . . .	24
Figura 7 – Representação da entidade “Portaria” e seus relacionamentos. . . .	25
Figura 8 – Painel de busca da página de consulta de portarias . . . . .	28
Figura 9 – Exemplo do histórico de alterações de uma portaria . . . . .	29
Figura 10 – Fluxograma da primeira versão de assinatura de portarias do novo SIPRO . . . . .	30
Figura 11 – Tela de geração de arquivo para o DOU . . . . .	33
Figura 12 – Representação das entidades que possuem relacionamentos com entidades do SIG . . . . .	34
Figura 13 – Representação visual do template Jasper . . . . .	41
Figura 14 – Padrão MVC aplicado ao JavaFX . . . . .	42
Figura 15 – Escolha de interessados, visão geral . . . . .	45
Figura 16 – Exemplo da funcionalidade de sugestão de nomes . . . . .	46
Figura 17 – Opção de não escolher interessados . . . . .	46
Figura 18 – Inserção de um interessado que não está no banco de dados . . . .	46
Figura 19 – Inserindo interessados em progressão por mérito . . . . .	47
Figura 20 – Etapa de cadastro do conteúdo da portaria . . . . .	48
Figura 21 – Etapa de cadastro do conteúdo da portaria (progressão por capacidade) . . . . .	49
Figura 22 – Novo fluxo de criação de portarias . . . . .	50
Figura 23 – Tela de “assinatura” de portarias . . . . .	51
Figura 24 – Diálogo de envio de arquivo assinado . . . . .	52
Figura 25 – Fluxograma da antiga funcionalidade de “assinar” portarias . . . . .	53
Figura 26 – Exemplo de seleção de várias portarias para assinatura . . . . .	54
Figura 27 – Assinatura de portarias com visualização prévia . . . . .	55
Figura 28 – Assinatura automatizada de portarias . . . . .	56
Figura 29 – Fluxograma de assinatura usando o componente assinador . . . . .	56
Figura 30 – Novo método de publicação de boletim mensal . . . . .	57

## Lista de tabelas

Tabela 1 – Descrição dos status de uma portaria do novo SIPRO . . . . .	20
Tabela 2 – Definição das variáveis disponíveis nos templates e suas descrições	24
Tabela 3 – Lista de caminhos disponibilizados pelo webservice . . . . .	31
Tabela 4 – Escopos utilizados no SIPRO . . . . .	37

## Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>11</b>
<b>1.1</b>	<b>Contexto</b>	<b>11</b>
<b>1.2</b>	<b>Motivações</b>	<b>12</b>
<b>1.3</b>	<b>Justificativa</b>	<b>13</b>
<b>1.4</b>	<b>Trabalhos relacionados</b>	<b>14</b>
<b>1.5</b>	<b>Estrutura do trabalho</b>	<b>15</b>
<b>2</b>	<b>METODOLOGIA</b>	<b>16</b>
<b>2.1</b>	<b>Contexto</b>	<b>16</b>
<b>2.2</b>	<b>Informatização total do fluxo de portarias</b>	<b>18</b>
<b>2.3</b>	<b>Modelagem de dados</b>	<b>19</b>
2.3.1	Portaria	19
2.3.2	Conteúdo	21
2.3.3	Interessados de uma portaria	22
2.3.4	Templates de portarias	23
2.3.5	Portaria assinada	25
<b>2.4</b>	<b>Proposta da primeira versão</b>	<b>26</b>
2.4.1	Geração do número de portaria	26
2.4.2	Criação de portarias	27
2.4.3	Publicação e exibição de portarias do boletim mensal	27
2.4.4	Geração de um documento de portaria	27
2.4.5	Consulta de portarias	28
2.4.6	Edição e histórico de portarias	28
<b>2.5</b>	<b>Proposta da segunda versão</b>	<b>29</b>
2.5.1	Webservice	30
2.5.2	Assinatura digital	31
2.5.3	Arquivo para publicação no Diário Oficial da União	32
<b>2.6</b>	<b>Integração com o SIG e a portabilidade entre bancos de dados</b>	<b>33</b>
<b>2.7</b>	<b>Tecnologias utilizadas</b>	<b>34</b>
2.7.1	JavaServer Faces	34
2.7.2	Spring Security	39
2.7.3	Hibernate	39
2.7.4	JasperReports	40
2.7.5	JavaFX	41
2.7.6	Retrofit	43
2.7.7	iText PDF	44

<b>3</b>	<b>ANÁLISE</b> . . . . .	<b>45</b>
<b>3.1</b>	<b>Criação de portarias</b> . . . . .	<b>45</b>
<b>3.2</b>	<b>Edição de portarias</b> . . . . .	<b>50</b>
<b>3.3</b>	<b>Assinatura de portarias</b> . . . . .	<b>51</b>
<b>3.4</b>	<b>Boletim mensal</b> . . . . .	<b>56</b>
<b>4</b>	<b>CONSIDERAÇÕES FINAIS</b> . . . . .	<b>58</b>
<b>4.1</b>	<b>Trabalhos futuros</b> . . . . .	<b>61</b>
	<b>REFERÊNCIAS</b> . . . . .	<b>62</b>

# 1 Introdução

## 1.1 Contexto

A informática surgiu como uma área do conhecimento completamente diferente de tudo que o ser humano já havia produzido ou experimentado até então. Inicialmente pensada para usos militares, mas logo foi reconhecida a sua capacidade de fazer parte de absolutamente todos os setores da humanidade. Com a criação e difusão da internet, o mundo vai caminhando num processo gradual de informatização. Isto é, muitas coisas que eram feitas com trabalho humano e ferramentas diversas começaram a ser substituídas por computadores e pelo trabalho computacional. Cálculos complexos, transações bancárias e compras são alguns dos numerosos exemplos da informatização presentes na atualidade.

Processos muito burocráticos e que demoravam muito para serem concluídos agora são feitos em questão de segundos com a internet, como, por exemplo, o envio de correios eletrônicos ou utilização de assinatura digital. Entretanto, essa não é a única função da informática. Uma área que há pouco tempo era apenas teórica, mas que vem cada vez mais se tornando viável é a chamada “internet das coisas”: como o nome diz, é a internet para comunicação não só de pessoas, mas de diversos objetos, incluindo carros, prédios, casas, entre outros. Dessa forma, seria possível controlar remotamente esses objetos, pois cada um iria possuir um identificador único. Um exemplo viável do uso da internet das coisas seria um termostato que pode verificar quais são as condições climáticas do seu bairro para deixar o ar condicionado na temperatura ideal para quando você chegar em casa. Assim, é possível adicionar “inteligência” aos objetos do dia-a-dia, usando diversos sensores e conectividade à internet.

Além dos diversos usos da informática, é importante reafirmar a correlação entre o nível de informatização dos países e o seu nível de desenvolvimento: de acordo com (INTERNATIONAL TELECOMMUNICATION UNION, 2017), os trinta países com os maiores índices estão também no maior grupo classificatório de IDH (muito elevado). Entretanto, isso não significa que um país com IDH baixo necessariamente chegue nesse grupo classificatório apenas expandindo o uso da informática. Na verdade, essa correlação indica uma tendência entre os países mais desenvolvidos, que é aplicar cada vez mais a informática onde for possível. Outro ponto a ser reiterado é que esse índice considera também outros fatores econômicos e sociais, como o preço dos serviços, o Produto Interno Bruto (PIB) do país, a renda per capita e níveis de escolaridade da população. Ou seja, o primeiro lugar da classificação não é necessariamente o mais informatizado.

## 1.2 Motivações

Como dito anteriormente, a informatização é uma tendência mundial. Um grande exemplo de país informatizado, se não o maior, é a Estônia. A Estônia é um país pequeno, mas com um ótimo IDH (0,865 de acordo com estimativas de 2015 e 30.<sup>º</sup> na classificação mundial, classificado como “IDH muito elevado”) e no índice de desenvolvimento das tecnologias de informação e comunicação, ocupa a 17<sup>a</sup> posição. O que mais chama a atenção nesse país é a forma como o governo utiliza a internet<sup>1</sup>: atualmente, 99% dos serviços públicos são disponibilizados para os cidadãos de forma eletrônica. Além disso, praticamente todos os cidadãos estonianos possuem uma identificação digital, que é um cartão com um chip embutido que utiliza criptografia de chave pública de 2048 bits e serve tanto para identificação em meio virtual quando para assinatura de documentos de forma digital. A Estônia também conta com eleições online desde 2005, sendo pioneira nessa funcionalidade, além de ser referência no mundo pelo uso de *blockchain* em seus sistemas, assim garantindo um altíssimo nível de segurança.

No caso do Brasil, a sua posição no índice de desenvolvimento das tecnologias de informação e comunicação é a 66.<sup>a</sup>. Embora a sua nota tenha aumentado em relação a 2016 (6,12 de 2017 contra 5,89 de 2016), o Brasil ainda tem um longo caminho para chegar num nível de informatização como a Estônia. Não só em questão de inclusão digital, mas na informatização de processos especialmente no setor público, onde mesmo com a grande presença de computadores em quase todos os órgãos públicos, ainda existem muitos processos que são feitos de forma analógica, o que resulta em custos e burocracia desnecessários.

Nas instituições públicas, um grande exemplo do sucesso da informatização de processos é o novo método de alistamento militar: antigamente, era necessário ir de madrugada para uma instituição militar para fazer o alistamento, e as filas duravam cerca de sete horas. Agora, no início do ano, foi implementado o alistamento online, onde a pessoa pode realizar todo o processo pela internet, no conforto de sua casa, e evitar filas.

Entretanto, a informatização não se trata apenas de eficiência. Outro ponto muito importante é a sustentabilidade que ela traz: processos que envolvem impressão de papel, que conseqüentemente envolve uso de tinta, são resumidos a processos virtuais, que envolvem sistemas computacionais e armazenamento digital. Além disso, documentos físicos ocupam muito espaço para serem armazenados, e ainda podem perder a validade com o tempo, dependendo no nível de conservação dos mesmos. Isso não acontece com documentos virtuais, que podem ser copiados sem perder a validade e que demandam capacidade de armazenamento muito baixa. Adicionalmente,

<sup>1</sup> Site oficial da Estônia: <https://e-estonia.com/>

com o uso de assinatura digital, esses documentos possuem a mesma validade jurídica que documentos físicos.

### 1.3 Justificativa

A Universidade Federal do Pará apresenta um bom nível de informatização, especialmente com o uso do SIG-UFRN (PORTAL DE COOPERAÇÃO - SINFO/UFRN, ): inscrição em processos seletivos, resultados, administração acadêmica com o SIGAA (Sistema Integrado de Gestão de Atividades Acadêmicas), entre outros. Entretanto, ainda existem muitas tarefas e fluxos que podem ser informatizados. O fluxo abordado nesse trabalho é o fluxo de portarias da UFPA.

(KASPARY, 2016) define “portarias” como: “atos pelos quais as autoridades competentes determinam providências de caráter administrativo, dão instruções sobre execução de leis e de serviços, definem situações funcionais e aplicam medias de ordem disciplinar”. As portarias são documentos públicos, e na universidade, são majoritariamente assinados pelo Reitor, ou substituto em exercício.

Por muitos anos a gerência de portarias na UFPA se fez de forma manual e não sistêmica. Por ser parte do fluxo de processos que tramitam em diversas unidades, cada unidade criava suas portarias com o uso de editores de texto e as anexavam ao processo para encaminhamento. Isso gerava portarias fora de padrão tanto na linguagem quanto na formatação. Além disso, por causa da dependência ao processo, eram muitos os passos necessários para se conseguir uma segunda via da portaria.

A forma de acessar essas portarias também era um problema: antigamente, os servidores só podiam ver as portarias por meio do boletim mensal, utilizando uma listagem classificada pelos tipos de portarias, e cada portaria possuía apenas um número e um ano, o que tornava muito difícil a busca de uma portaria utilizando critérios como: nome do interessado citado na portaria, servidor responsável por assinar a portaria, entre outros.

Além da facilidade de acesso à informação e da transparência do andamento das portarias, um objetivo também visado era a desburocratização do fluxo de portarias. Processos burocráticos são conseqüentemente menos eficientes, e esse era um dos maiores problemas do fluxo antigo de portarias: portarias levavam cerca de dois a três dias para serem assinadas.

Outros pontos que justificam a criação do novo sistema de portarias são:

- Economia de papel e tinta usados para imprimir portarias (cerca de três vias por portaria);
- Automatização de grande parte do processo de criação de portarias;

- Integração com o banco do SIG para reaproveitar dados importantes e atrelar portarias às entidades do banco de dados, como: interessado de uma portaria, órgão da portaria e o servidor que assina a portaria;
- Implementação de assinaturas digitais nas portarias;

Em resumo, com base em tudo o que foi dito até então nesse capítulo, o objetivo geral definido para este trabalho é o de informatizar e reformular o fluxo de portarias da UFPA, reduzindo custos e a burocracia. Para atingir esse objetivo, foram propostos alguns objetivos específicos, entre eles:

- Projetar o sistema para centralizar todo o fluxo de portarias, da criação até a assinatura;
- Gerar documentos (PDF) das portarias, eliminando a necessidade de editores de texto externos;
- Criação de uma seção pública de consulta de portarias, onde elas podem ser consultadas usando parâmetros diversos;
- Implementação de um módulo de assinatura digital;

O sistema criado conseguiu atingir todas as metas, informatizando completamente o fluxo de portarias e conseqüentemente eliminando a necessidade de imprimir portarias tanto para assinatura quanto para encaminhamento para os servidores interessados<sup>2</sup>. Além disso, a interface pública de consulta resolveu um problema antigo das portarias: a grande dificuldade de achar uma portaria quando o número da mesma não era conhecido. Além de resolver demandas internas, esse sistema chegou como uma grande contribuição para a UFPA, que agora possui um método mais eficiente, automatizado, econômico e transparente de gerenciar portarias.

#### 1.4 Trabalhos relacionados

No escopo do novo sistema, o trabalho mais relacionado é o SIG, que serviu como a base de dados necessária para viabilizar esse sistema, além de possuir interfaces, métodos e classes reutilizadas no novo sistema. Mesmo não seja estritamente necessário o uso do SIG, ele se tornou uma peça fundamental nos primeiros momentos de uso do sistema, pois já possuía dados completos de todos os servidores da UFPA.

<sup>2</sup> No contexto de portarias, um servidor interessado é aquele servidor que é diretamente afetado pela resolução da mesma. Ex.: em uma portaria de afastamento, o servidor afastado é o interessado.

Um sistema de portarias não é uma ideia totalmente nova. As universidades federais de Alagoas<sup>3</sup> e Mato Grosso<sup>4</sup> também possuem sistemas de portarias, entretanto, não foi feita uma pesquisa para obter a lista de funcionalidades desses sistemas. Para o contexto do novo sistema de portarias da UFPA, a única inspiração foi o antigo sistema SIPRO. A ideia era criar um sistema que agregasse as funcionalidades do antigo SIPRO, e que, ao mesmo tempo, possuísse novas funcionalidades e simplificações de processos, que serão mostrados mais adiante.

## 1.5 Estrutura do trabalho

Nesse trabalho, os tópicos abordados são: tecnologias utilizadas, estrutura, módulos e descrição do novo fluxo de elaboração de portarias, seguido de uma análise do sistema proposto, da criação até a assinatura de forma digital, e depois mostrando um pouco da mudança de paradigma causado pela implantação do sistema. E esses tópicos são apresentados da seguinte forma: no capítulo 2, o fluxo de portarias antigo é descrito e a arquitetura do sistema é descrita, seguido de suas principais funcionalidades e como se dá a integração com o SIG. No capítulo 3 é mostrado uma visão geral de como o sistema é usado. No capítulo 4 apresenta-se por fim as considerações finais seguidas das referências presentes nesse trabalho.

---

<sup>3</sup> Sistema de portarias da UFAL: <http://sistemas.ufal.br/portarias/>

<sup>4</sup> Sistema de portarias da UFMT: <http://sistemas.ufmt.br/portaria/Login/Login.aspx>

## 2 Metodologia

### 2.1 Contexto

Uma das motivações para a criação de um novo sistema de portarias foi a limitação do sistema de portarias antigo. O antigo SIPRO foi implementado em 2007, com o objetivo de centralizar o armazenamento de portarias e resolver o problema do número de portaria, que deveria ser único e incremental. Como as portarias eram, e ainda são, criadas por várias pessoas, era muito difícil saber qual seria o próximo número a ser usado em uma portaria. Esse problema foi resolvido com o antigo SIPRO, que passou a possuir o controle do número incremental.

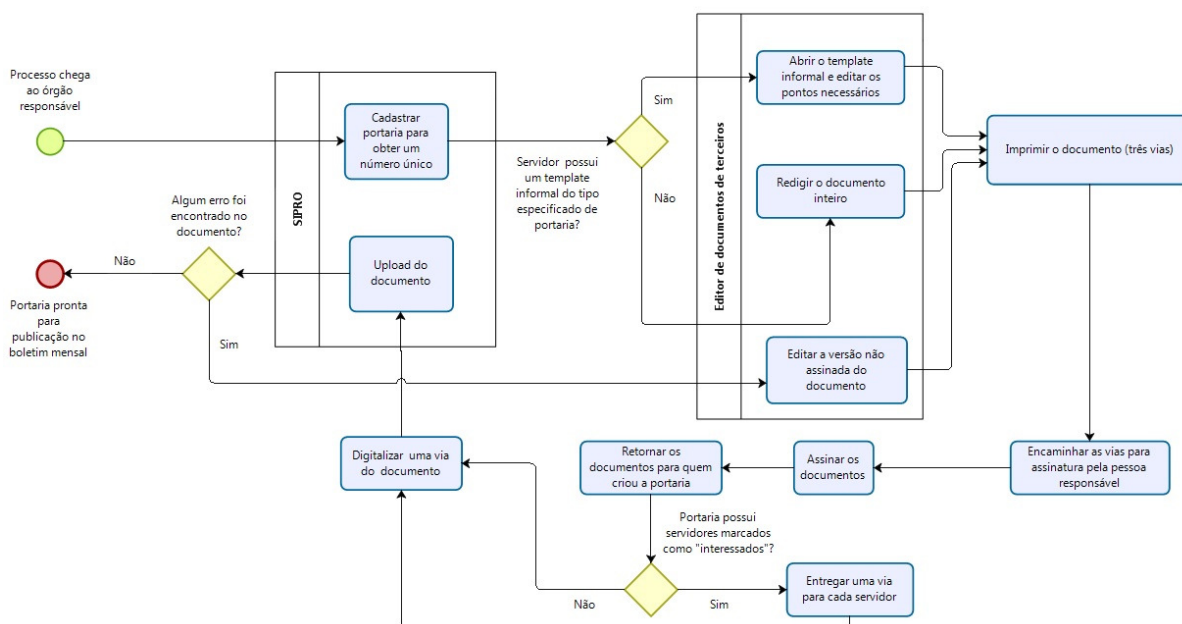
Entretanto, o fluxo de portarias após a criação do SIPRO contava apenas com duas partes informatizadas e centralizadas: a geração do número e o armazenamento do documento assinado à mão para posterior postagem no boletim. A criação do documento ficou da mesma forma que antes, isto é, usando um programa de terceiros para redigir o documento. A publicação no boletim mensal era feito por outro programa, que recuperava todos os documentos assinados no mês escolhido e gerava uma página (HTML) com os links para *download*. Essa página era exibida no endereço eletrônico da PROGEP.

Mesmo com uma nova forma de gerar boletins, ainda havia muitos problemas nesse fluxo, especialmente problemas relacionados à integridade dos documentos: como o SIPRO antigo era responsável por gerar o número da portaria, os servidores primeiro cadastravam a portaria vazia para conseguir o número da mesma, fazendo upload de um arquivo vazio ou de um documento que não tinha relação nenhuma com a portaria, e depois disso criavam o documento efetivamente, usando o número obtido no sistema. Feito isso, o servidor fazia novamente o upload do arquivo, dessa vez, do arquivo real. O problema era que muitas vezes os servidores esqueciam de enviar o novo arquivo, ou ainda enviavam um arquivo errado, e isso devia ser revisto manualmente antes da publicação do boletim mensal.

Embora esse sistema tenha se mostrado prático por permitir o armazenamento de portarias assinadas, assim como consultas simples, ainda não era o ideal, visto que o fluxo das portarias podia ser totalmente informatizado. Com o tempo foi surgindo novas demandas e novas situações que muitas vezes requeriam muitos passos exaustivos, como por exemplo: editar uma portaria já assinada, por causa de algum erro de escrita. Nesse caso, era necessário editar o documento, imprimir o mesmo número de vias impressas da primeira vez, reenviar todas as vias para o servidor responsável pela assinatura daquela portaria, digitalizar novamente uma das vias assinadas e fazer a

substituição no sistema. Esse foi um problema frequente enquanto o sistema antigo estava em uso, e isso dificultava a rastreabilidade por parte dos servidores interessados naquela portaria (servidores citados diretamente no documento e que seriam de alguma forma afetados por aquela portaria), que não tinham como saber o andamento do fluxo, pois, o sistema só era acessível por quem cadastrava as portarias.

Figura 1 – Fluxo antigo de portarias



Fonte: Arquivo da PROGEP

No fluxograma acima, é descrito o fluxo antigo de portarias, da criação até uma versão pronta para publicação no boletim mensal. Com essa imagem é possível identificar alguns problemas implícitos nesse fluxo, como o uso de *templates* informais para gerar os documentos e o número de passos necessários para editar uma portaria que já foi assinada, sendo que na maioria das vezes os erros eram pequenos (uma palavra ou frase incorreta, ou um nome errado).

Um problema que era causado pela falta de informatização completa do fluxo de portarias era o uso excessivo de papel, considerando que, em média, eram criadas de 20 a 30 portarias por dia, e cada portaria precisava ser impressa em três vias. Olhando de uma perspectiva ecológica, além do custo financeiro, o desperdício de papel era algo relevante nesse fluxo, mesmo quando tudo ocorria de forma correta.

Outro empecilho encontrado no gerenciamento de portarias era a dificuldade de consultar portarias quando o mês de publicação no boletim era desconhecido e a pessoa não possuía um cadastro no SIPRO. Nesse caso, era necessário pesquisar cada boletim até achar a portaria necessária.

## 2.2 Informatização total do fluxo de portarias

A ideia de criar um sistema como uma solução para os problemas encontrados no fluxo de portarias surgiu em 2017. Na época, duas escolhas eram possíveis: criar um sistema sem reutilizar os modelos de dados e as funcionalidades do sistema legado ou reformular o antigo SIPRO. A escolha mais plausível foi a de criar um novo sistema, pois o modelo lógico, o modelo de dados e até a aparência do sistema foi totalmente refeito. Foi definido que o sistema novo agregaria as funcionalidades de: cadastrar, consultar e gerar o boletim mensal, sendo este último migrado do sítio web da PROGEP para o novo SIPRO. Além disso, para possibilitar consultas mais diversificadas em parâmetros de busca, foi decidido usar o banco de dados do SIG. Como o banco do SIG é rico em informações de todos os servidores da universidade, isso eliminou a necessidade de criar credenciais para os servidores que usam o sistema. Com a integração, também se tornou possível criar relacionamentos entre servidores da universidade e portarias, sendo os exemplos mais relevantes: o servidor responsável por assinar a portaria, que agora pode entrar no sistema com as credenciais do SIG e acessar todas as portarias que ele precisa assinar, e o servidor interessado, ou servidores interessados, que agora podem consultar o sistema usando o nome de interessado como parâmetro de busca, não precisando saber o número da portaria.

Para viabilizar a criação do sistema, três grupos de pessoas precisariam ser satisfeitas, tendo as necessidades referentes à criação de portarias saciadas e o trabalho necessário reduzido e automatizado onde possível. Esses grupos são:

- 1) Servidores que não estão relacionados com a criação de portarias, mas precisam acessar portarias específicas;
- 2) Servidores que criam portarias;
- 3) Servidores que assinam portarias;

Um dos primeiros requisitos pensados para facilitar o uso do sistema por parte de todos os usuários da UFPA foi o de utilizar o mesmo esquema visual do SIGAA (Sistema Integrado de Gestão de Atividades Acadêmicas) e SIGRH (Sistema Integrado de Gestão de Recursos Humanos). Para isso, foi utilizado o mesmo padrão arquitetural desses sistemas, incluindo linguagem e algumas tecnologias. Com isso, foi possível reaproveitar classes e métodos de comunicação com o banco de dados de outros projetos que usam o banco de dados do SIG.

Inicialmente, o grupo priorizado foi o grupo das pessoas que efetivamente criam portarias, por serem as pessoas responsáveis pela maior parcela de trabalho no fluxo das portarias. Nesse estágio, o principal objetivo foi o de eliminar o uso de programas

de terceiros no fluxo completo de portarias, centralizando a criação de documentos no próprio sistema, assim forçando uma padronização de espaçamento, fontes, e outros aspectos visuais que eram diferentes entre as portarias publicadas nos boletins. O motivo disso eram os templates informais usados antigamente: como cada servidor possuía as suas próprias portarias criadas na sua estação de trabalho, não havia um consenso sobre a padronização de documentos.

Outro ponto importante que precisou ser pensado no desenvolvimento do sistema foi a implementação de um ambiente que seja similar a um programa de edição de terceiros, onde a característica mais importante é a de sempre mostrar uma prévia de como o documento vai ficar depois de pronto. Com isso implementado, se tornou possível remover o uso desses programas para gerar os documentos, deixando assim boa parte do fluxo de portarias centralizado no novo SIPRO.

## 2.3 Modelagem de dados

Para projetar um sistema eficiente que atendesse às demandas dos usuários que participavam direta ou indiretamente do fluxo de portarias, o primeiro passo foi o de abstrair as estruturas e as entidades que precisariam ser utilizadas nesse sistema. Esse passo foi um dos mais importantes da projeção da arquitetura, pois permitiu ter uma visão geral de como o sistema ficaria antes de começar a projetar a lógica e as telas do sistema.

### 2.3.1 Portaria

O ponto de partida da modelagem de dados foi a estrutura de uma portaria, que, como a descrição do sistema sugere, é a estrutura central do fluxo. O que deveria ser pensado nessa abstração era, além da estrutura do documento, os relacionamentos existentes com outras entidades, como os servidores da universidade, e como essa entidade podia ser estruturada de forma a resolver problemas antigos de fluxo, como a edição de portarias e uma publicação consistente no boletim mensal.

O primeiro problema que foi considerado na modelagem da portaria foi a edição de uma portaria já criada, considerando erros triviais e frequentes na etapa de criação de uma portaria. Para permitir uma solução rápida desse problema, foi decidido criar um modelo de portarias versionado, ou seja, que pode ser editado diversas vezes sem haver perda de histórico. Dessa forma, é possível restaurar versões anteriores sem a necessidade de escrever tudo de novo.

Assim, o conteúdo do documento foi desacoplado da entidade “portaria”, e passou a ser um relacionamento, onde uma portaria pode ter vários conteúdos, e a

partir de um conteúdo, o documento é gerado. A portaria passou a ter um tipo, autor, número, ano, data de cadastro e uma lista de conteúdos (com no mínimo um elemento).

Além de pensar em um método que permitisse várias edições de uma portaria, também foi preciso modelar a entidade de forma a respeitar o fluxo de portarias, se assegurando de que uma portaria que não foi assinada não poderá entrar no boletim mensal, e uma portaria já publicada no boletim não poderá mais ser editada, entre outras regras. Com o intuito de tratar esse fluxo, foi criada a entidade “Situacao”, que é uma abstração do *status* atual de uma portaria. Os status definidos foram:

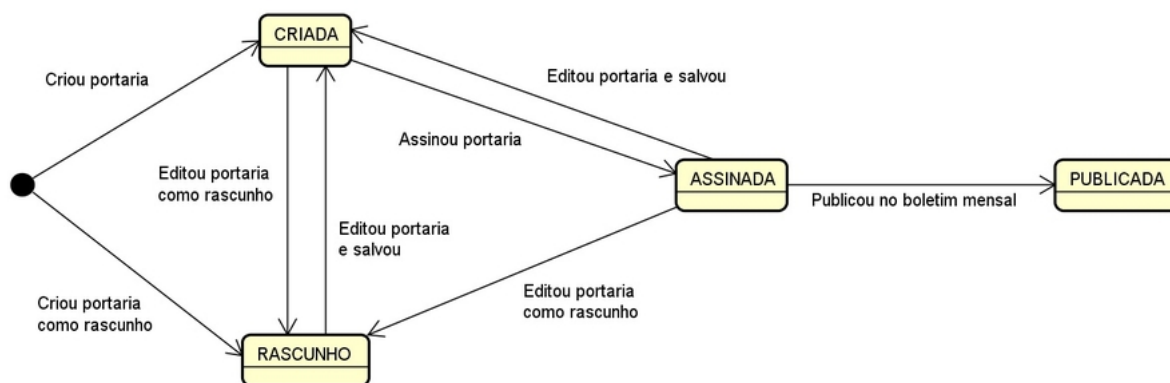
**Tabela 1 – Descrição dos status de uma portaria do novo SIPRO**

Nome	Descrição
Rascunho	Portaria criada no sistema, porém, não elegível para assinatura, pois, está incompleta.
Criada	Portaria criada com o conteúdo completo no sistema e elegível para assinatura.
Assinada	Portaria assinada pelo servidor responsável.
Publicada	Portaria publicada no boletim mensal de portarias.

Fonte: Banco de dados do novo SIPRO

Quando os status possíveis foram definidos, foi necessário revisar o fluxo antigo de portarias para verificar as relações entre os status e como o sistema deveria tratar problemas antigos como: editar uma portaria já assinada, mas não publicada no boletim mensal. Para isso, essas regras de negócio foram abstraídas para uma máquina de estados, como mostrado abaixo:

**Figura 2 – Máquina de estados dos status de portarias**



Fonte: Sistema SIPRO

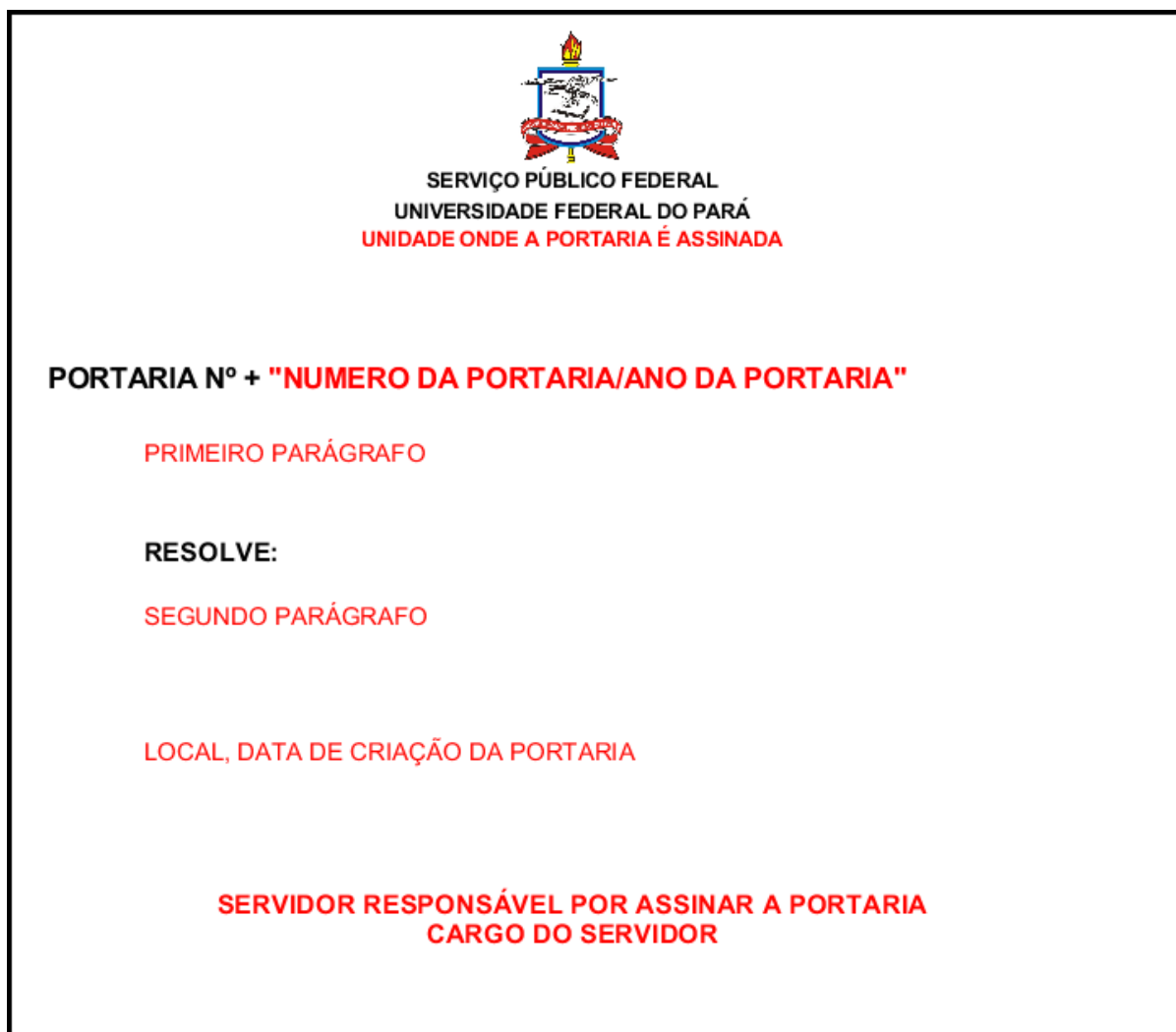
### 2.3.2 Conteúdo

Como dito anteriormente, o conteúdo da portaria foi desacoplado da entidade base. Isso ocorreu a fim de possibilitar um sistema de portarias versionado, onde era possível editar uma portaria sem perder os dados anteriores. Além disso, no sistema antigo, uma das regras de negócio do sistema era de possuir um documento em PDF atrelado a uma instância de “portaria”, e isso foi reformulado no novo sistema. A principal razão foi a falta de flexibilidade para fazer edições: pegar um documento pronto e fazer alterações “em cima” do que já estava feito poderia desorganizar o documento. Outro motivo pensado foi o alto custo de armazenamento em disco que um sistema de portarias versionado demandaria, com documentos prontos em cada versão. Para resolver isso, a decisão tomada foi a de abstrair para o banco somente o conteúdo textual dos documentos, deixando a geração do documento para a lógica da aplicação, que faz isso em tempo real, quando necessário.

A modelagem do conteúdo foi feita para funcionar de forma similar a um código-fonte de um programa: a partir do código-fonte do programa, que demanda pouco espaço em disco, é possível obter a versão final do programa, fazendo a compilação, e a versão compilada não permite a flexibilidade de fazer alterações. Analogamente, o banco de dados do novo SIPRO armazena o “código-fonte” do documento referente à portaria, e a camada de aplicação do sistema é responsável por fazer a “compilação” desse código em um documento PDF, pronto para assinatura.

Para obter os campos de dados necessários, foi feita uma análise da estrutura do documento, conforme a figura abaixo:

Figura 3 – Estrutura textual de uma portaria



Fonte: Arquivo da PROGEP

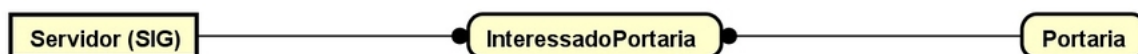
Com base na estrutura analisada, foi possível obter as variáveis necessárias para gerar um documento padronizado. Um detalhe importante é que, mesmo contendo apenas a informação textual no documento, ainda assim alguns campos são relacionamentos no banco de dados, como, por exemplo, o servidor responsável por assinar a portaria. Isso abre a possibilidade de usar esses relacionamentos como parâmetro de busca, além de permitir notificar os servidores que precisam assinar essas portarias.

### 2.3.3 Interessados de uma portaria

No contexto das portarias, um interessado é um agente que vai ser diretamente afetado pela resolução daquela portaria, e uma portaria pode ter zero ou mais interessados. Um exemplo prático disso é uma portaria de afastamento: nesse caso, o servidor afastado é o interessado dessa portaria. Outro exemplo é uma portaria de

progressão por mérito, onde vários servidores progredem (em torno de cinquenta), logo, são os interessados da portaria. Além disso, um servidor pode ser um interessado em zero ou mais portarias. Então, para abstrair essa entidade, foi feita uma relação do tipo “Muitos-para-muitos” entre a entidade “Servidor”, do SIG, e a entidade “Portaria”, do novo SIPRO.

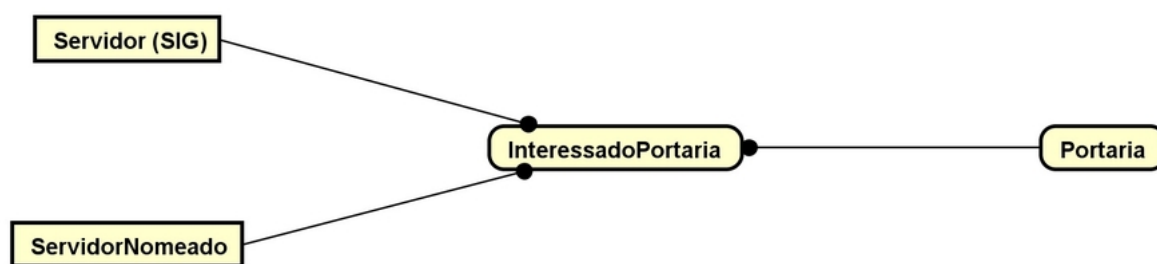
**Figura 4 – Primeira versão da abstração de um “Interessado”.**



Fonte: Banco de dados do novo SIPRO

Porém, existia outro caso a ser avaliado: quando alguém entra na UFPA por meio de um concurso público, existe um período entre a criação da portaria referente à nomeação dessa pessoa e o cadastro dela no banco de dados do SIG. Dessa forma, a primeira versão da abstração do “Interessado” não iria funcionar nesse caso, pois o servidor ainda não está cadastrado no banco de dados do SIG. Para resolver esse problema, foi incluído um outro relacionamento, com a entidade “ServidorNomeado”, do novo SIPRO. Vale ressaltar que os campos “Servidor” e “ServidorNomeado” são mutuamente excludentes, e isso é garantido pela lógica do sistema.

**Figura 5 – Segunda versão da abstração de um “Interessado”.**



Fonte: Banco de dados do novo SIPRO

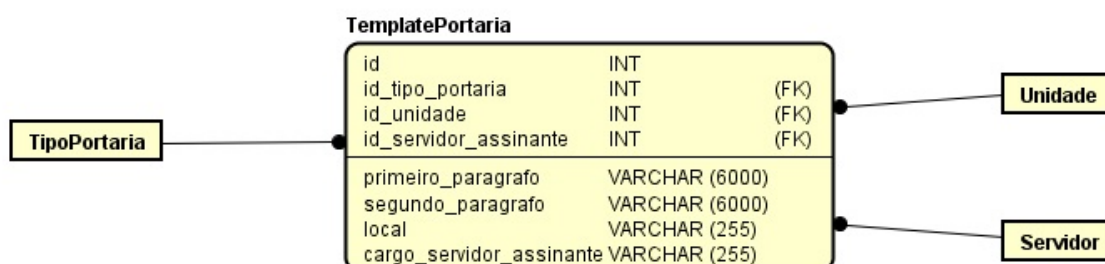
#### 2.3.4 Templates de portarias

Para poder remover o uso de programas de criação de documentos de terceiros sem haver resistência por parte dos servidores que criavam portarias, era necessário pensar em um jeito de formalizar os templates de portarias, assim economizando muito tempo no cadastro, considerando que os documentos finais tinham poucas partes modificadas dos templates previamente utilizados.

Os novos templates de portarias foram obtidos com base nos campos de dados da entidade “Conteúdo”, e foi criado um relacionamento do tipo “um para um” no banco

de dados entre a entidade “TipoPortaria” e “TemplatePortaria”. Assim, era garantido que cada tipo possuía o seu próprio template. E os templates não incluem somente texto, mas também outros relacionamentos, como o servidor responsável por assinar a portaria (alguns tipos de portarias eram assinadas somente pelo reitor, por exemplo).

Figura 6 – Representação simplificada da entidade “TemplatePortaria” no banco de dados.



Fonte: Banco de dados SIPRO

Os templates possuem também algumas variáveis incluídas no campo “segundo\_paragrafo”, e essas variáveis são representadas por uma palavra que segue o padrão: “##” + nome da variável + “##”. Dessa forma, quando uma variável é incluída no texto do segundo parágrafo do template, o sistema automaticamente substitui a variável pelo valor que ela representa. Isso permitiu uma maior taxa de automatização da criação de portarias.

Tabela 2 – Definição das variáveis disponíveis nos templates e suas descrições

Variável	Descrição
##SERVIDOR##	Inserir os nomes de todos os servidores marcados como “interessados”, separados por vírgula
##SERVIDOR_INDIVIDUAL##	Inserir um dos servidores marcados como “interessado”, de acordo com a ordem de inserção desses servidores na portaria

Variável	Descrição
##UNIDADE##	Inserir as unidades de lotação de todos os servidores marcados como “interessados”, separados por vírgula
##CLASSE##	Inserir as classes de todos os servidores marcados como “interessados”, separados por vírgula
##PADRAO##	Inserir os padrões de todos os servidores marcados como “interessados”, separados por vírgula

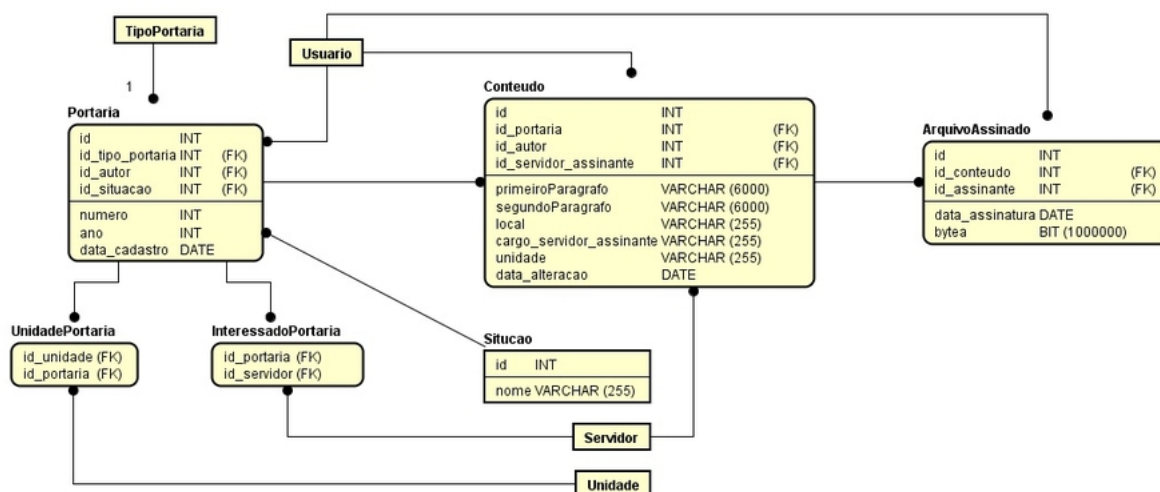
Fonte: Sistema SIPRO

### 2.3.5 Portaria assinada

Usando a ideia inicial de implementar assinaturas digitais nas portarias, se torna impossível armazenar portarias assinadas da mesma forma que os conteúdos, pois envolve criptografia, e modificar um documento assinado digitalmente o tornaria inválido. Por isso, a abordagem escolhida foi a de armazenar o documento assinado no banco de dados.

Para garantir o andamento normal do fluxo de portarias, foi decidido que cada conteúdo de portaria teria um relacionamento com o documento assinado. Assim, quando uma portaria assinada é editada, uma nova versão de conteúdo é criada, e para a portaria voltar a ser assinada, é necessário assinar a nova versão do conteúdo.

Figura 7 – Representação da entidade “Portaria” e seus relacionamentos.



Fonte: Banco de dados do SIPRO

## 2.4 Proposta da primeira versão

O sistema atualmente possui duas versões, sendo a primeira versão lançada em março de 2018. Essa versão conta com as funcionalidades antigas do antigo sistema, como: criação de uma portaria e geração do número da portaria, junto com novas funcionalidades já mencionadas previamente. Além de novas funcionalidades, o novo sistema deveria também adaptar funcionalidades antigas para o novo formato do sistema. Dessa forma, quem era acostumado a usar o sistema antigo não teria problemas com o novo.

### 2.4.1 Geração do número de portaria

A geração de número de portaria era algo bem arriscado de ser implementado, pois o sistema deveria garantir que esse número fosse único e incremental, e num contexto em que há a possibilidade de criação paralela de portarias, isso poderia ser um problema.

Pensando nisso, a solução implementada foi direto no banco de dados, pois o mesmo possui mecanismos que garantem a execução correta em contextos paralelos. Para isso foi criado um *trigger*: uma função que é chamada toda vez que certa condição é satisfeita. Nesse caso, a condição escolhida foi: a cada inserção de uma linha de “portaria” no banco de dados. Antes de a linha ser adicionada no banco, essa função recupera o maior número utilizado para aquele ano, incrementa em 1, e atribui o novo número para a nova linha. Depois disso, a linha é adicionada, já com o número de portaria atribuído.

#### **Código 2.1 – Declaração da função "gerar\_numero\_portaria"**

```
CREATE OR REPLACE FUNCTION gerar_numero_portaria()  
  RETURNS trigger AS  
$BODY$DECLARE  
  maior_numero integer;  
BEGIN  
  maior_numero = (SELECT max(numero_portaria) FROM  
    portaria WHERE ano_portaria = NEW.ano_portaria);  
  IF maior_numero IS NULL THEN  
    NEW.numero_portaria = 1;  
  ELSE  
    NEW.numero_portaria = maior_numero + 1;  
  END IF;  
  RETURN NEW;  
END;$BODY$
```

#### 2.4.2 Criação de portarias

Como descrito anteriormente, e diferente do sistema SIPRO antigo, a criação de uma portaria no novo sistema não consiste somente no cadastro de interessados e o tipo de portaria, mas também todo o conteúdo presente no PDF a ser gerado para assinatura. Com o uso dos templates de conteúdo por tipo de portaria e a possibilidade de prévia do documento PDF final, o tempo de criação de portarias pôde ser drasticamente reduzido.

#### 2.4.3 Publicação e exibição de portarias do boletim mensal

No fluxo antigo, o boletim mensal era uma página web que possuía um link para download de cada portaria assinada naquele mês. O objetivo era tornar público parte do fluxo de portarias, já que o SIPRO era um sistema exclusivamente interno. No novo sistema, a necessidade de um boletim mensal foi removida, pois a funcionalidade de busca parametrizada de portarias já supria essa necessidade.

Porém, como as pessoas envolvidas no fluxo de portarias e os interessados eram acostumados a usar um boletim mensal para consulta de portarias, foi criada uma forma eficiente de representar um novo boletim: usando o status da portaria. Agora, a página de boletim do novo SIPRO faz uma consulta de todas as portarias com status “PUBLICADA” do mês selecionado. Isso permitiu reformular essa funcionalidade sem precisar modificar a modelagem de dados da portaria. No novo sistema, o processo de publicação do boletim mensal é bem simples: a pessoa que publica o boletim precisa somente escolher o mês. O processo todo leva segundos para ser completo na lógica do sistema. Essa foi uma melhoria significativa, pois permitiu o uso de documentos já salvos no sistema para a exibição no boletim, diferente do método anterior, onde os arquivos precisavam ser copiados para outro servidor.

Para a exibição do boletim mensal, o conceito de boletim foi interpretado como: as portarias com status “PUBLICADA” no período que vai do início do mês escolhido até o fim do mês escolhido. Com isso em mente, a página do boletim mensal agora recebe o resultado dessa consulta no banco de dados. As portarias então são agrupadas por tipo, e os tipos são agrupados por grupo de portaria, assim como era organizado o boletim no site da PROGEP.

#### 2.4.4 Geração de um documento de portaria

Uma das características mais importantes do sistema é a de possibilitar a geração de um documento PDF usando os dados de um conteúdo de portaria, pois esse documento é usado tanto para consulta como para a assinatura e posterior

publicação no boletim mensal, e essa funcionalidade era crucial para o plano de informatizar completamente o fluxo de portarias.

#### 2.4.5 Consulta de portarias

O antigo SIPRO possuía uma tela de consulta de portarias, porém, como o sistema todo era interno, era uma funcionalidade pouco utilizada. Além de não ser vinculada com o SIG, o que impedia o uso de parâmetros de servidores, como os interessados ou o servidor assinante daquela portaria. Ao invés disso, eram utilizadas palavras simples. Com o novo sistema, a funcionalidade de consulta agora é pública e conta com diversos parâmetros do SIG, como: unidade, interessados, servidor assinante.

**Figura 8 – Painel de busca da página de consulta de portarias**

Parâmetros de busca

<input type="checkbox"/> Número da portaria:	<input type="text"/>
<input type="checkbox"/> Ano:	<input type="text"/>
<input type="checkbox"/> Período do cadastro:	<input type="text"/> a <input type="text"/>
<input type="checkbox"/> Assinante:	<input type="text"/>
<input type="checkbox"/> Tipo:	<input type="text" value="Escolha um tipo de portaria"/>
<input type="checkbox"/> Nome do(s) interessado(a)(s):	<input type="text"/> <input type="button" value="Limpar campo"/>
<input type="checkbox"/> Situação:	<input type="text" value="Escolha uma situação"/>
<input type="checkbox"/> Unidade:	<input type="text"/> <input type="button" value="Limpar campo"/>

Fonte: Novo SIPRO

Diferentemente do sistema antigo, na nova página de consulta é possível buscar portarias com mais de um parâmetro por vez, apenas marcando o *checkbox* ao lado do nome do parâmetro. Para evitar sobrecarga do sistema, os resultados são exibidos em uma lista paginada, onde são carregadas apenas as portarias da página atual do usuário. Além disso, para evitar ataques DoS (*Denial of Service*) ou DDoS (*Distributed Denial of Service*), onde o sistema é sobrecarregado por exaustivas requisições simultâneas, foi introduzido um *captcha*, onde o usuário precisa completá-lo antes de fazer uma consulta.

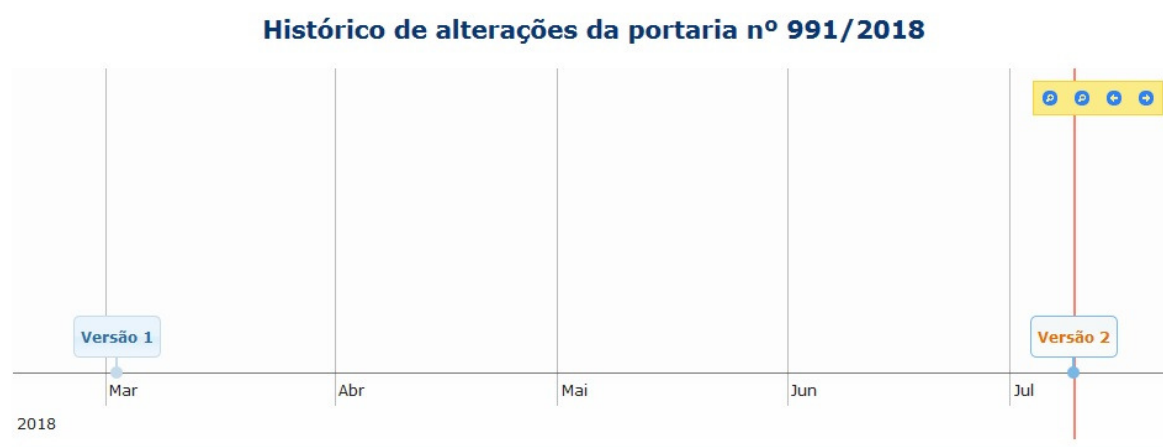
#### 2.4.6 Edição e histórico de portarias

Como dito anteriormente, a modelagem de dados das portarias e dos conteúdos das portarias foram feitos para suportar um formato versionado, onde o conteúdo de uma versão anterior não se perderia ao criar outra. Foi possível reutilizar a mesma página usada para criar portarias. A diferença é que a página carrega o conteúdo da

portaria a ser editada e preenche todos os campos automaticamente. Com isso, o usuário apenas modifica o que for preciso.

Junto com a edição, veio a funcionalidade de histórico de portarias. Embora não muito utilizada atualmente, essa funcionalidade é importante, pois permite visualizar as versões anteriores e até restaurar uma, caso necessário. Entretanto, quando uma versão é restaurada, mesmo assinada, ela vai precisar ser assinada novamente por motivos de segurança e consistência.

**Figura 9 – Exemplo do histórico de alterações de uma portaria**



Fonte: Novo SIPRO

## 2.5 Proposta da segunda versão

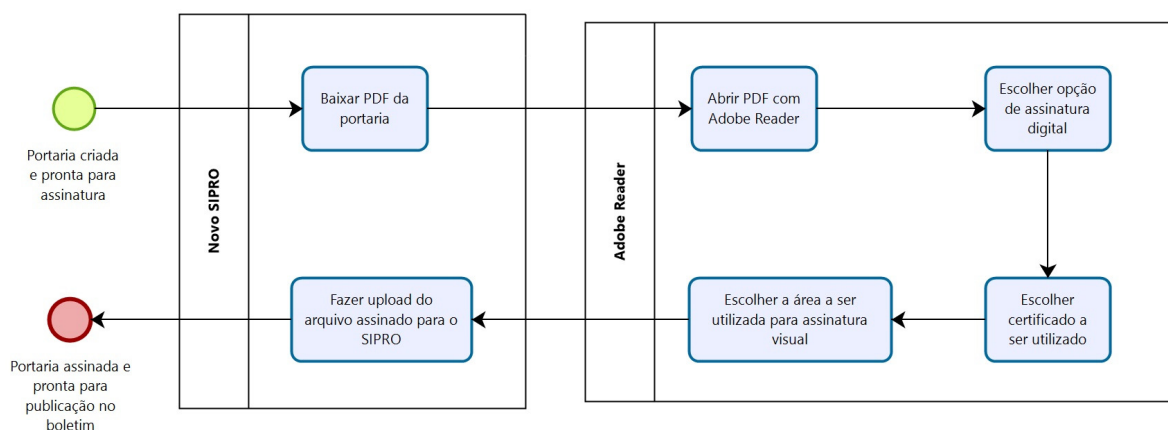
Com a primeira versão do SIPRO já implementada, foi verificado que o fluxo de processos do sistema não estava próximo do ideal que foi pensado. Uma das motivações de criar esse sistema foi justamente a economia de papel e tinta, e a completa informatização do fluxo de portarias, da criação até a publicação no boletim mensal. Porém, embora a criação, edição, consulta e boletim mensal estivessem todos agregados ao sistema, a assinatura ainda era feita manualmente: os servidores assinantes imprimiam a portaria no sistema (ou delegavam alguém para fazer isso), assinavam a portaria e faziam o upload no sistema, assim mudando o status da portaria para assinada, e a tornando elegível para o boletim mensal do mês de criação da portaria.

Em suma, o fluxo novo não estava totalmente informatizado e a economia de papel e tinta reduziu, mas não foi levada ao máximo, visto que antes, mais de uma cópia era impressa. Agora, somente uma cópia era impressa, para assinatura. Para otimizar mais o fluxo de portarias e completar a informatização, foi decidido implementar assinatura digital nas portarias, pois é um método informatizado e confiável. Dessa forma, não seria mais preciso imprimir portarias e as assinaturas seriam feitas dentro

do escopo do sistema.

A primeira solução pensada foi utilizar um programa externo para assinar os documentos, como o “Adobe Reader”. O método funcionava normalmente, porém, eram necessários 5 a 6 passos para assinar apenas uma portaria. Para quem era responsável por assinar frequentemente, como o reitor, que assina em média quarenta portarias por dia, esse método era inviável.

**Figura 10 – Fluxograma da primeira versão de assinatura de portarias do novo SIPRO**



Fonte: Novo SIPRO

Para otimizar esse fluxo, foi decidido criar um componente exclusivo para assinatura digital, onde se torna possível assinar várias portarias de forma automatizada e rápida, usando um *token* USB que contém o certificado digital de quem assina.

### 2.5.1 Webservice

A primeira funcionalidade implementada na segunda versão do SIPRO foi um *webservice*. A motivação para o *webservice* foi a possibilidade de adicionar QRcodes em cada portaria, para permitir uma consulta rápida sem precisar acessar a página pública do sistema e nem precisar fazer login.

(COSTA, 2015) define um *webservice* como uma API (*Application Programming Interface*) que utiliza a rede como meio de comunicação, e uma API é definida como uma interface que permite a disponibilização de serviços para outros sistemas. Dessa forma, quem utiliza a API para consumir algum serviço não precisa saber como a API é implementada, apenas os dados retornados.

Com a implementação da assinatura digital em mente, foi importante pensar em uma forma de validar uma portaria assinada digitalmente, mas impressa por algum motivo: acessando o endereço representado pelo QRcode, é possível ver rapidamente

o status da portaria, assim como baixar o PDF assinado para fazer a validação por alguma ferramenta externa, caso necessário.

O webservice implementado foi posteriormente melhorado para o formato REST (*Representational State Transfer*) para ser utilizado pelo novo componente de assinatura digital, e os serviços disponibilizados por esse webservice são descritos abaixo:

**Tabela 3 – Lista de caminhos disponibilizados pelo webservice REST**

Método HTTP	Caminho	Descrição	Parâmetros
GET	/consulta/nome_reitor	Método usado para obter o nome do reitor em exercício atualmente	Login, senha
GET	/consulta/portaria/{id}/pdf	Método usado para obter o documento PDF de uma portaria específica	Login, senha, id da portaria
GET	/consulta/minhas_portarias/pendentes	Método usado para obter as portarias pendentes de assinatura pelo usuário sinalizado pelo login informado	Login, senha
GET	/consulta/portarias/pendentes	Método usado para obter as portarias pendentes de assinatura, sem filtragem de servidor assinante	Login, senha
POST	/upload/uploadPortaria	Método usado para fazer upload de uma portaria assinada para o servidor	Login, senha, id da portaria, hash do PDF, bytes do PDF

Fonte: Novo SIPRO

### 2.5.2 Assinatura digital

A assinatura digital é regulamentada no Brasil pela Medida Provisória 2.200-2/2001. Ela estabelece que todo documento eletrônico assinado digitalmente com certificado emitido pela ICP-Brasil tem validade jurídica. Além de ter a mesma validade jurídica que uma assinatura manuscrita, ela também é mais segura. Isso acontece pois a assinatura digital é à prova de falsificações: é impossível um documento assinado digitalmente ser adulterado sem a adulteração ser identificada por alguma ferramenta de verificação de assinatura.

(LOWAGIE, 2012) introduz a assinatura digital em arquivos PDF usando o sis-

tema de chaves públicas e privadas, que é o sistema utilizado para assinar digitalmente documentos. Para fazer uma assinatura digital, é necessário primeiro gerar um *hash* do documento, que é um “resumo criptográfico” do mesmo, que possui um tamanho fixo, ou seja, independente do tamanho do arquivo. Esse hash é criptografado com a chave privada do assinante e anexado ao documento original. Para verificar a autenticidade da assinatura, é preciso primeiro descriptografar o hash anexado ao documento usando a chave pública do emissor. Em seguida, é preciso gerar o hash do documento recebido, comparar o hash obtido com o hash previamente descriptografado. Se os hashes forem iguais, significa que o documento não foi alterado e foi assinado pela pessoa que teve a chave pública utilizada para descriptografar a assinatura.

Como dito anteriormente, a assinatura de portarias de forma manuscrita era uma prática que deveria ser evitada com a implementação do sistema, e os métodos encontrados de assinatura digital necessitavam de muitos passos para cada documento. Isso fez com que esses métodos fossem descartados do contexto de portarias. Para tornar a assinatura digital viável no fluxo de portarias, foi criado um componente separado, dedicado à assinatura de portarias.

A primeira ideia foi criar um módulo dentro do sistema web do SIPRO. Entretanto, os navegadores possuem limitações nessa área: como a assinatura digital é feita nesse contexto por dispositivos USB, isso seria impossível no sistema web, pois o acesso à dispositivos USB é algo fora do escopo de aplicações web que são executados em navegadores. A segunda ideia foi criar um Java Applet, que é uma aplicação Java capaz de ser executada em um navegador, e que pode acessar dispositivos externos. O empecilho encontrado foi que Java Applets estão obsoletos atualmente, e os navegadores não estão mais oferecendo suporte a esse tipo de aplicação. Com isso, a ideia final foi de fazer um componente *desktop* sem vínculo direto com o SIPRO, mas que se comunica com ele por meio da API descrita previamente.

### 2.5.3 Arquivo para publicação no Diário Oficial da União



















Outra funcionalidade importante implementada na segunda versão do SIPRO foi o arquivo para o Diário Oficial da União. Embora não faça parte do fluxo de portarias, essa funcionalidade se configura como um complemento importante para o novo SIPRO. Esse arquivo era feito de forma manual pelos servidores, e consiste em agrupar todas as portarias de um período selecionado, porém resumindo cada portaria ao conteúdo do segundo parágrafo apenas. Como o SIPRO guarda o conteúdo de cada portaria, ao invés do documento PDF pronto, ficou bem mais simples de implementar essa funcionalidade.

Para gerar o arquivo, basta escolher o período e confirmar se as portarias estão corretas. Também há a opção de remover uma portaria do arquivo gerado.

Figura 11 – Tela de geração de arquivo para o DOU

Parâmetros de busca

Data início:   
 Data fim:

Resultados							Selecionar colunas
(1 of 1)							40
Nº	Ano	Data de cadastro	Autor	Tipo	Situação	Opções	
2080	2018	19/06/2018	lucas-ms	NOMEAÇÃO PARA CARGO EFETIVO TAE	CRIADA	 	
2059	2018	18/06/2018	lucas-ms	AFASTAMENTO PARA O EXTERIOR (EVENTOS, CONGRESSOS, SEMINÁRIOS E OUTROS)	ASSINADA	 	
2060	2018	18/06/2018	lucas-ms	AFASTAMENTO PARA O EXTERIOR (EVENTOS, CONGRESSOS, SEMINÁRIOS E OUTROS)	ASSINADA	 	
2061	2018	18/06/2018	lucas-ms	COOPERAÇÃO TÉCNICA	ASSINADA	 	
2052	2018	18/06/2018	lucas-ms	CESSÃO	ASSINADA	 	
2044	2018	18/06/2018	claudiabraga	DESIGNAÇÃO DE FG	CRIADA	 	
2043	2018	18/06/2018	claudiabraga	DESIGNAÇÃO FCC	CRIADA	 	
2045	2018	18/06/2018	claudiabraga	DESIGNAÇÃO E DISPENSA DE FG	CRIADA	 	
2046	2018	18/06/2018	claudiabraga	DESIGNAÇÃO DE FG	CRIADA	 	

(1 of 1) 40

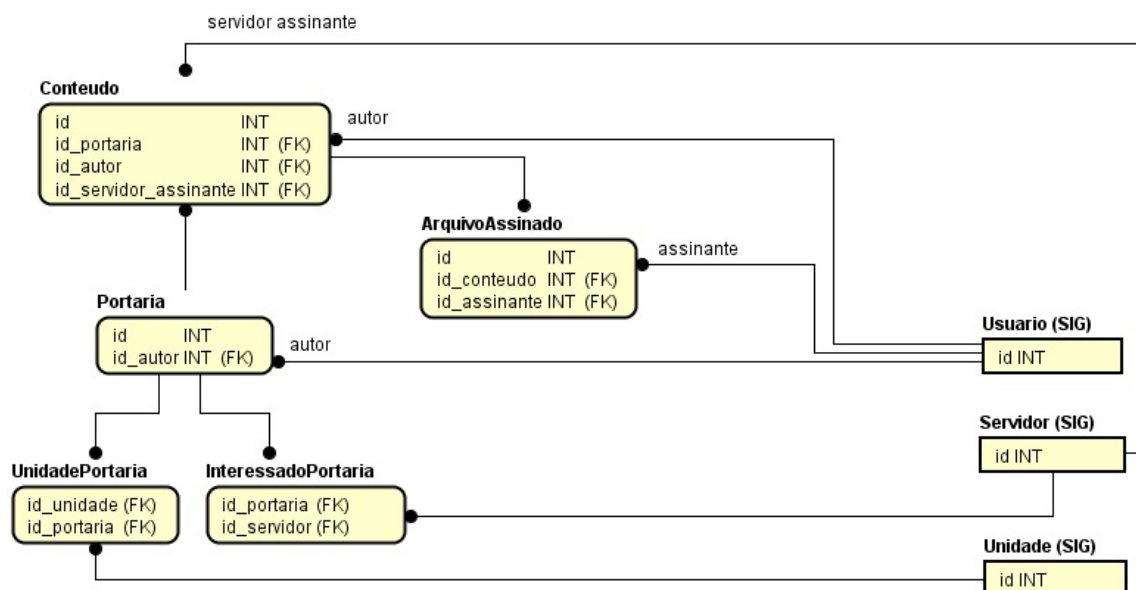
Fonte: Novo SIPRO

## 2.6 Integração com o SIG e a portabilidade entre bancos de dados

Como dito anteriormente, uma decisão de projeto foi integrar o novo SIPRO com o banco de dados SIG, com o objetivo de adaptar melhor o sistema ao contexto em que ele está inserido e permitir um reaproveitamento de dados de um banco rico em informações relevantes para o fluxo de portarias. Embora não tenha sido um requisito, uma consequência da forma como a integração foi implementada foi a alta portabilidade do SIPRO em relação a modelos de dados.

Na fase inicial do projeto, foi escolhido não modificar as entidades já prontas no SIG. No lugar disso, o sistema foi projetado visando suportar futuras alterações no banco de dados. A consequência disso foi uma grande portabilidade referente a banco de dados: o SIPRO pode ser utilizado em qualquer banco que possua apenas três entidades de fora do próprio *schema*, necessitando apenas de pequenas alterações sobre como pegar certos atributos de entidades diferentes.

Figura 12 – Representação das entidades que possuem relacionamentos com entidades do SIG



Fonte: Banco de dados SIPRO

## 2.7 Tecnologias utilizadas

### 2.7.1 JavaServer Faces

JavaServer Faces é uma especificação para projetos de interface de usuário web, que é baseada em componentes. Usando JSF, é possível incluir *frameworks* de interface gráfica que disponibilizam um conjunto de componentes prontos para serem usados em páginas web, não precisando se preocupar com a implementação desses componentes, apenas nos eventos do usuário (como cliques). Assim, o JSF abstrai a manipulação de eventos e organização dos componentes nas páginas, fazendo com que o programador precise se preocupar apenas com a lógica da aplicação.

O JSF atualmente usa internamente um sistema de templates chamado Facelets. De acordo com o criador da tecnologia (HOOKOM, 2005a), esse sistema foi criado para facilitar a vida dos desenvolvedores que utilizavam JSF, pois, com a forma antiga, era necessário a criação de vários arquivos XML diferentes para definir componentes e utilizá-los nas páginas. Esse sistema é responsável por permitir a inserção de componentes visuais que não fazem parte do conjunto de elementos padrões em páginas HTML. Dessa forma, é possível utilizar frameworks com diversos componentes prontos. Para suportar essa tecnologia, é necessário utilizar páginas em um formato específico, nesse caso, XHTML. Essas páginas têm a sua versão em HTML construída antes de enviar a resposta da requisição para o usuário. Assim, os navegadores recebem a resposta em HTML. Para usar um componente de alguma biblioteca, é necessário

primeiro declarar uma letra ou inicial na tag <html> da página para identificar cada biblioteca e assim inserir os componentes.

#### Código 2.2 – Declaração de bibliotecas

```
<?xml version='1.0' encoding='ISO-8859-1' ?>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:ui="http://java.sun.com/jsf/facelets"
      xmlns:h="http://java.sun.com/jsf/html">
</html>
```

Com as bibliotecas declaradas, basta inserir os componentes com as tags específicas na documentação da biblioteca:

#### Código 2.3 – Inserção de componentes

```
<body>
  <h:outputLabel value="ESSE É UM COMPONENTE DE TEXTO DA
    BIBLIOTECA PADRÃO DE JSF" />
</body>
```

O sistema de facelets também permite o uso de templates de página, onde uma página referencia outro arquivo XHTML (modelo mestre) que possui campos reservados para sobrescrita (HOOKOM, 2005b). Assim, o cliente de modelo, ou seja, quem o referencia, pode preencher o conteúdo reservado e reutilizar os componentes declarados no modelo mestre, assim economizando código.

#### Código 2.4 – Modelo mestre utilizado no SIPRO

```
<?xml version='1.0' encoding='UTF-8' ?>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:ui="http://java.sun.com/jsf/facelets"
      xmlns:h="http://java.sun.com/jsf/html">

<h:head>
  <meta http-equiv="Content-Type" content="text/html;
    charset=UTF-8" />
  <title>SIPRO - UFPA</title>
  <h:outputStylesheet library="css" name="geral.css" />
</h:head>
<h:body>
  <div id="container">
    <h:panelGroup>
      <ui:include src="menu.xhtml" />
    </h:panelGroup>

    <div id="conteudo">
      <ui:insert name="corpo" />
    </div>

    <div id="rodape">
```

```

        <div id="rodape_texto">
            <a href="http://www.progep.ufpa.br">
                Assessoria de Tecnologia de Informação</a>
            </div>
        </div>
    </div>
</h:body>
</html>

```

No exemplo acima, o nome do arquivo é “template\_geral.xhtml”. O componente “ui:include” sinaliza outro arquivo XHTML que precisa ser inserido nesse modelo, que no caso é “menu.xhtml”. Esse arquivo contém apenas a tag “ui:composition” que significa que esse arquivo é um componente composto, não uma página XHTML independente. Já a tag “ui:insert” define uma área a ser preenchida pelos modelos cliente, com o nome “corpo”. Dessa forma, o modelo cliente possui uma “implementação” dessa área.

#### Código 2.5 – Exemplo de uma página que referencia o modelo mestre definido previamente

```

<?xml version='1.0' encoding='ISO-8859-1' ?>
<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:ui="http://java.sun.com/jsf/facelets"
      xmlns:h="http://java.sun.com/jsf/html">
  <ui:composition template="template_geral.xhtml">
    <ui:define name="corpo">
      <p>CONTEÚDO DA PÁGINA AQUI</p>
    <ui:define />
  </ui:composition>
</html>

```

Atualmente, todas as páginas (XHTML) do novo SIPRO referenciam o mesmo arquivo “template\_geral.xhtml”, pois elas possuem o mesmo cabeçalho, rodapé e menu horizontal, em todo o sistema.

A comunicação entre as páginas (XHTML) e a lógica do sistema é possível utilizando os chamados “Managed Beans”. Managed beans são uma extensão dos Java Beans: objetos (Java) que seguem uma convenção específica de funcionamento (classes com construtores vazios, classes serializáveis e que tenham atributos acessíveis por métodos “get” e “set”) e que são administrados pelo JSF (JAVATPOINT, ).

#### Código 2.6 – Exemplo de um Java Bean

```

public class Pessoa implements java.io.Serializable {

    private String nome;
    private Integer idade;

    public String getNome() {
        return nome;
    }
}

```

```

public void setNome(String nome) {
    this.nome = nome;
}
public Integer getIdade() {
    return idade;
}
public void setIdade(Integer idade) {
    this.idade = idade;
}

public Pessoa() {
    this.idade = 0;
    this.nome = "";
}
}

```

Os Managed Beans possuem um escopo, ou seja, um qualificador que define as condições para o objeto continuar persistindo na memória da aplicação. Os diferentes escopos utilizados no SIPRO são descritos abaixo:

**Tabela 4 – Escopos utilizados no SIPRO**

Escopo	Descrição
<i>View</i>	O objeto continua na memória enquanto o usuário interagir com a mesma página XHTML, sem redirecionamentos e sem navegar entre páginas. Beans com esse escopo dependem de uma sessão HTTP ativa para serem instanciados. Assim, cada sessão HTTP implica em um objeto novo. Vale ressaltar que Beans com esse escopo são instanciados somente quando referenciados por uma página XHTML.
<i>Session</i>	O objeto continua na memória enquanto a sessão HTTP do usuário estiver ativa. Nesse caso, um bean é criado para cada sessão HTTP.
<i>Application</i>	O objeto continua na memória enquanto a aplicação web estiver sendo executada. Ou seja, Beans com esse escopo não dependem de sessões de usuário. Logo, apenas um bean é criado quando a aplicação inicia sua execução, e esse bean pode ser utilizado por todas as sessões HTTP

Escopo	Descrição
Fonte: Novo SIPRO	

Para fazer a comunicação entre páginas e Beans, é utilizada a EL (Expression Language), que é usado nas páginas XHTML (ORACLE, 2013). Usando EL, é possível referenciar tanto métodos quanto atributos, e também é possível validar expressões booleanas, além de poder realizar operações aritméticas.

#### Código 2.7 – Exemplo de uso do EL - Página web

```
<?xml version='1.0' encoding='UTF-8' ?>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://java.sun.com/jsf/html">

<h:head>
  <meta http-equiv="Content-Type" content="text/html;
    charset=UTF-8" />
</h:head>
<h:body>
  <div>
    #{testeBean.texto}
    #{testeBean.getTexto()}
  </div>
</h:body>
</html>
```

#### Código 2.8 – Exemplo de uso do EL - Managed Bean

```
@Named
@ViewScoped
public class TesteBean implements Serializable {

  private String texto;

  public TesteBean() {
    texto = "Texto teste";
  }

  public String getTexto() {
    return texto;
  }

  public void setTexto(String texto) {
    this.texto = texto;
  }
}
```

Na página renderizada, em ambos os casos, o texto exibido será o mesmo texto inicializado no construtor. Isso acontece pois o JSF se encarrega de chamar por padrão os getters e setters dos beans.

### 2.7.2 Spring Security

Para gerenciar a segurança do projeto web, é usado o Spring Security, um framework lançado em 2007 que provê autenticação (quem está usando tal recurso?) e autorização (a pessoa autenticada pode usar esse recurso?) para projetos corporativos (ALEX et al., 2017). Usando permissões cadastradas no banco de dados, é possível configurar o Spring Security para fazer a checagem usando essas permissões como parâmetros.

Outra característica importante do Spring Security no contexto do projeto é a possibilidade de integração com páginas XHTML. Usando uma tag definida na página, é possível aplicar condições para a renderização de componentes no HTML gerado, utilizando permissões. Assim, se o usuário que faz a requisição da página não possui permissão de usar um componente, esse componente não é renderizado. Isso pode ser feito como no exemplo abaixo:

#### Código 2.9 – Uso da tag do Spring Security para autorizar o uso de componentes

```
<!DOCTYPE html>
<html
  xmlns="http://www.w3.org/1999/xhtml"
  xmlns:sec="http://www.springframework.org/security/tags">

  <body>
    <div>
      <sec:authorize ifAnyGranted="PERMISSAO_CLIQUE">
        <button type="button" onclick="alert('Tem
          permissão')">Pressione</button>
      </sec:authorize>
    </div>
  </body>
</html>
```

### 2.7.3 Hibernate

Hibernate é um framework de mapeamento objeto-relacional para Java. Ou seja, com o Hibernate, é possível fazer a conversão automática de uma entidade relacional do banco de dados para um objeto Java. Além disso, as consultas são feitas pela linguagem de consultas do Hibernate, o HQL. Nessa linguagem, ao invés de utilizar tabelas e entidades, são utilizadas classes e objetos. Assim, a manipulação de entidades na aplicação fica muito mais simples.

Além do mapeamento, o Hibernate conta com várias facilidades para fazer consultas e para persistir entidades no banco. Uma delas é o salvamento em cascata: com essa funcionalidade, é possível persistir ou deletar em cadeia uma entidade e todos os seus relacionamentos (MIHALCEA et al., ). Um exemplo seria uma classe “Pessoa” que possui um relacionamento com outra classe, “Endereço”. Ao criar uma instância de “Pessoa”, uma instância de “Endereço” e atribuir o endereço à pessoa, ao salvar uma linha da entidade Pessoa no banco, uma linha de Endereço também será salva, sem precisar de codificações extras. O mesmo ocorre no processo de remoção de linha. Se a pessoa é removida, o endereço também é removido, o que evita uma falha de consistência no banco.

#### 2.7.4 JasperReports

Como a funcionalidade de gerar o documento da portaria é algo muito utilizado, especialmente na criação ou edição de portarias, onde o usuário precisa ter um feedback visual de toda alteração feita, foi necessário usar um método eficiente de geração de documentos. Para isso, foi utilizado o JasperReports, uma biblioteca Java capaz de gerar documentos de forma rápida usando uma estrutura definida pelo usuário.

A estrutura do documento da portaria, que é passada como parâmetro para o Jasper, é um arquivo com a extensão “.jasper”, um arquivo compilado originado de um arquivo “.jrxml”, um arquivo que utiliza uma linguagem de marcação para definir toda a estrutura do documento, incluindo os parâmetros de dados utilizados para preencher o documento final. Um detalhe importante sobre o Jasper é que ele trata corretamente características como quebra de página e reposicionamento dinâmico dos campos, caso algum fique maior que o esperado. Além disso, outro requisito para o documento gerado era a conformidade com o padrão de PDF/A-1A, algo que também é suportado pelo Jasper.

Figura 13 – Representação visual do template Jasper



Fonte: Sistema SIPRO

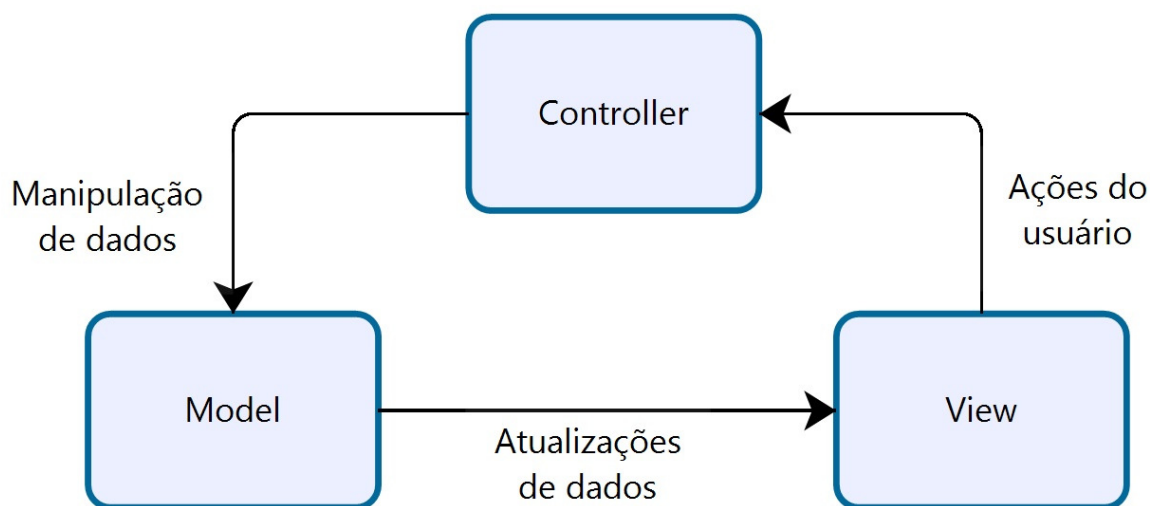
No template acima, os campos que contêm a expressão “\$F{campo}” são alimentados pela aplicação, que envia os dados de uma entidade do tipo “Conteúdo”. O link de consulta e o código QR são gerados em tempo de execução.

### 2.7.5 JavaFX

Para implementar o componente assinador, foi usado o framework visual JavaFX, um dos grandes recursos novos do Java 8. Com o JavaFX, é possível trabalhar com criação de telas de uma forma parecida com desenvolvimento web, com a linguagem de marcação conhecida como FXML. Na aplicação Java, os arquivos com a extensão FXML possuem um controller, uma classe responsável por fazer a comunicação entre a

lógica de aplicação e as telas do programa (para exibição e manipulação de dados, por exemplo), sendo os controllers todos implementados pelo programador. Dessa forma, o JavaFX é, por padrão, feito para ser usado com MVC.

**Figura 14 – Padrão MVC aplicado ao JavaFX**



Fonte: Adaptado de "The theory behind Model View Controller- Disponível em <[https://developer.mozilla.org/en-US/docs/Web/Apps/Fundamentals/Modern\\_web\\_app\\_architecture/MVC\\_architecture](https://developer.mozilla.org/en-US/docs/Web/Apps/Fundamentals/Modern_web_app_architecture/MVC_architecture)>

Como em uma página HTML, os componentes declarados no FXML podem possuir um id, e no controlador, para acessar esse componente, o atributo responsável precisa possuir o mesmo nome e a mesma classe do componente, e a anotação @FXML em cima da declaração desse atributo. Assim, o carregador do JavaFX injeta automaticamente o componente no atributo na própria inicialização da tela.

**Código 2.10 – Componentes FXML com id**

```
<TextField fx:id="usuarioTF" />
<PasswordField fx:id="senhaPF" />
```

**Código 2.11 – Componentes FXML injetados no controlador**

```
public class LoginController implements Initializable {

    @FXML
    private TextField usuarioTF;

    @FXML
    private PasswordField senhaPF;
```

## 2.7.6 Retrofit

Para fazer a comunicação com o Webservice do SIPRO, é utilizada a biblioteca Retrofit, que simplifica a codificação de chamadas HTTP. Para usar o Retrofit, é necessário criar uma interface que contenha os métodos HTTP utilizados, e o retorno de cada método. Com o Retrofit, é possível também especificar o conversor utilizado nas respostas. O conversor, como o nome diz, é responsável por converter os dados de retorno da requisição para a aplicação. O servidor pode enviar a resposta em diferentes formatos (JSON, XML, texto aberto) e a aplicação precisa usar fazer a conversão de “resposta” para “objeto Java”.

O assinador utiliza apenas dois conversores: conversor de JSON e conversor de String. O conversor de JSON é o mais utilizado, e converte automaticamente a resposta do servidor para o objeto Java declarado na resposta de método, na interface usada pelo Retrofit.

### Código 2.12 – Interface de comunicação com o SIPRO

```
public interface ServicoSipro {  
  
    @GET("/rest/consulta/minhas_portarias/pendentes")  
    Call<List<Portaria>> getMinhasPortariasPendentes(  
        @HeaderMap Map<String, String> headers);  
  
    @GET("/rest/consulta/portarias/pendentes")  
    Call<List<Portaria>> getTodasPortariasPendentes(  
        @HeaderMap Map<String, String> headers);  
  
    @GET("/rest/consulta/portaria/{id}/pdf")  
    Call<PortariaPdf> getPortariaPdf(@HeaderMap Map<String,  
        String> headers, @Path("id") Integer id);  
  
    @GET("/rest/consulta/nome_reitor")  
    Call<String> getNomeReitor(@HeaderMap Map<String, String>  
        headers);  
}
```

A imagem acima mostra a interface utilizada para chamar os métodos REST do SIPRO. Acima da declaração do método, é necessário fazer uma anotação que contenha o método HTTP utilizado. No caso do sistema, só haviam caminhos com o método GET. Em todos os métodos, havia o Map de headers. No caso do assinador, esses headers eram utilizados para fazer o credenciamento do usuário em cada requisição, utilizando o nome de usuário e senha do usuário realizando a requisição. No método “getPortariaPdf”, existe outro parâmetro que é marcado com a anotação “Path”. Ou seja, o valor desse parâmetro irá ser inserido no trecho do caminho da URL sinalizado pelo Path, no caso, “id”. Assim, o trecho marcado com “{id}” será substituído pelo id passado como parâmetro. É importante observar também o retorno de cada

método: todos os retornos são encapsulados na classe “Call” do Retrofit. Essa classe é utilizada para fazer a requisição, que pode ser de forma síncrona ou assíncrona.

### 2.7.7 iText PDF

Para a funcionalidade de assinar as portarias, foi escolhida a biblioteca iText, uma biblioteca específica para manipulação de documentos PDF. Com ela, é possível fazer a assinatura usando o token USB do usuário e também posicionar a assinatura visual no local sinalizado no PDF. Vale ressaltar que o próprio Jasper utiliza essa biblioteca internamente para criar os PDFs das portarias. Porém, como o Jasper não suporta assinaturas digitais, foi necessário usar uma biblioteca independente.

Para fazer a assinatura digital do documento, é preciso primeiro escolher um certificado. Esse certificado será usado para identificar o assinante na hora de validar a assinatura. Para escolher o certificado, foi usado o sistema de *Keystore* do próprio Windows (Windows-MY). Dessa forma, não é preciso se preocupar com drivers e configurações de token, pois a autenticação e comunicação com o token são feitos pelo serviço do Windows (PI, 2016).

#### **Código 2.13 – Inicialização básica de um Keystore e consulta de certificados**

```
KeyStore ks = KeyStore.getInstance("Windows-MY");  
  
List<String> certificados = new ArrayList<>(ks.aliases());
```

Com o método mostrado acima, é possível obter uma enumeração de todos os certificados instalados no Keystore escolhido. O Keystore precisa ser recarregado toda vez que esse método é chamado, pois, se houver um novo certificado inserido, a recarga irá identificar corretamente. Assim, a string com o nome do certificado é guardada na aplicação e utilizada posteriormente para a assinatura, onde o token é consultado com base no nome do certificado escolhido.

## 3 Análise

### 3.1 Criação de portarias

A criação de portarias agora é dividida em três etapas: na primeira etapa, o usuário escolhe o tipo de portaria a ser criada. Essa etapa é importante, pois, existem dois tipos especiais de portarias onde o conteúdo difere dos outros tipos: portaria de progressão por mérito e progressão por capacitação. A portaria de progressão por capacitação se trata de uma progressão individual de um servidor através de um certo número de cursos de capacitação realizados. Esses cursos são listados em uma tabela, junto com a carga horária de cada curso. Para inserir tabelas no documento PDF, é preciso usar um template Jasper que tenha um espaço específico para tabelas. Dessa forma, dependendo do tipo de portaria, será utilizado um arquivo Jasper diferente na geração do PDF. Já a progressão por mérito é sobre um número variado de servidores que progrediram, e a tabela contém informações em cada linha como: SIAPE do servidor, nome completo, entre outros. Ou seja, nessa portaria, o conteúdo da tabela reflete o número de interessados daquela portaria, e vice-versa.

A segunda etapa é a de escolha dos interessados. Nessa etapa, o usuário escolhe o servidor (ou os servidores) interessados da portaria. Não é necessário haver interessado na portaria, assim como não é necessário haver apenas um interessado. Assim, há a opção de não inserir interessados, ou inserir vários.

**Figura 15 – Escolha de interessados, visão geral**

Fonte: Novo SIPRO

A inserção de interessados é feita a partir de uma busca pelo nome do interessado no banco do SIG, usando uma ferramenta de completar texto automaticamente: quando o usuário começa a digitar um nome no campo de texto, uma busca dinâmica é realizada automaticamente usando o trecho já digitado como parâmetro, para retornar sugestões de nomes. Essas sugestões vêm em forma de um objeto da entidade “Servidor”, do SIG. Dessa forma, é mostrado também a matrícula SIAPE e a situação do servidor sugerido (ativo, excluído, aposentado, entre outros). Para confirmar o interessado, basta clicar em um dos nomes completos sugeridos. Vale ressaltar que,

mesmo digitando o nome completo do servidor, se o usuário não clicar na sugestão, o interessado não é confirmado, por motivos de consistência com o banco do SIG.

**Figura 16 – Exemplo da funcionalidade de sugestão de nomes**

A interface apresenta dois tabs: "Interessado(s)" (ativo) e "Detalhes". Abaixo, há um campo de texto com o nome "EMMANUEL" digitado. Uma lista de sugestões é exibida, com "EMMANUEL ZAGURY TOURINHO" selecionado em amarelo. Um tooltip sobre o item selecionado mostra: "Nome: EMMANUEL ZAGURY TOURINHO", "Matrícula SIAPE: 6327480" e "Situação: Ativo". Um botão "Avançar" está à direita.

Fonte: Novo SIPRO

Caso não haja interessados, basta marcar o campo correspondente, assim permitindo avançar para a próxima etapa sem ser obrigado a escolher um interessado, como na figura abaixo:

**Figura 17 – Opção de não escolher interessados**

A interface apresenta os tabs "Interessado(s)" e "Detalhes". Abaixo, há o texto "Sem interessado(s) OU Interessados externo(s) ao órgão" seguido de uma caixa de seleção marcada com um checkmark. Um botão "Avançar" está à direita.

Fonte: Novo SIPRO

No caso do servidor interessado não estar no banco de dados, como no caso de alguém que foi recentemente nomeado para a universidade (tratado em 2.1.3), é possível inserir manualmente o nome completo e CPF da pessoa.

**Figura 18 – Inserção de um interessado que não está no banco de dados**

A interface apresenta os tabs "Interessado(s)" e "Detalhes". Abaixo, há campos para "Nome do servidor:" e "CPF do servidor:". Abaixo desses campos, há o texto "Servidor não encontrado:" seguido de uma caixa de seleção marcada com um checkmark. Um botão "Avançar" está à direita.

Fonte: Novo SIPRO

Na portaria de progressão por mérito, o cadastro de interessados fica diferente: para esse tipo de portaria, existe um programa externo responsável por gerar a portaria de forma automática, com a tabela incluída. Porém, para incluir essas portarias no novo SIPRO, seria muito trabalhoso para o servidor cadastrante digitar os interessados um por um, visto que portarias desse tipo geralmente têm mais de cinquenta interessados. Então, o que ficou acordado com os servidores cadastrantes foi que eles gerariam a

portaria pelo programa externo, e em seguida cadastrariam o conteúdo no novo SIPRO. E para a escolha de interessados, foi desenvolvido um método de identificação de tabela, para que o usuário cole o conteúdo da tabela no campo sinalizado, e o sistema identifique todos os interessados, linha por linha, e adicione à lista de interessados na portaria em questão.

**Figura 19 – Inserindo interessados em progressão por mérito**

Servidores identificados		
(1 of 1)		
Matrícula SIAPE	Nome completo	Opções
Sem resultados		

Fonte: Novo SIPRO

Assim, quando o conteúdo da tabela gerada é colado no campo e o botão “analisar tabela para identificar interessados” é clicado, o sistema analisa linha por linha e insere os interessados identificados na tabela abaixo do campo de texto. Também há a opção de adicionar um servidor manualmente. Assim, o servidor adicionado é automaticamente inserido no conteúdo da tabela.

A terceira e última etapa é o preenchimento do conteúdo do PDF. Nessa etapa, cada campo da estrutura da portaria possui um campo de texto próprio. Se o template cadastrado para o tipo selecionado de portaria possui alguma variável de template (variáveis descritas anteriormente), então o segundo parágrafo já é inicializado com os valores corretos. A variável de uso mais comum é a de adicionar os interessados. Nesse caso, o segundo parágrafo já é inicializado com os nomes dos interessados inseridos no texto.

Figura 20 – Etapa de cadastro do conteúdo da portaria

**Interessado(s)** Detalhes

Tamanho da fonte:

Unidade:  ?

Primeiro parágrafo: ?

**A PRÓ-REITORA DE DESENVOLVIMENTO E GESTÃO DE PESSOAL DA UNIVERSIDADE FEDERAL DO PARÁ,** no uso das atribuições que lhe confere a Portaria nº 4515/2016 de 13.10.2016 do Magnífico Reitor e tendo em vista o que consta no Processo n. \*NÚMERO DO PROCESSO \* desta universidade,

Segundo Parágrafo: ?

Conceder **ABONO DE PERMANÊNCIA** instituído pela Emenda Constitucional n. 41/2003 LUCAS DE MELO SILVA, nos termos do art. 3º, da Emenda Constitucional n. 47/2005, publicada no DOU de 06/07/2005, com vigência a partir de 23 de Novembro de 2015.


Local:  ?

Servidor a assinar:  ?

Designação do servidor a assinar:  ?

Autor(a) do cadastro: LUCAS DE MELO SILVA

Data do cadastro: 10/07/2018



**SERVIÇO PÚBLICO FEDERAL**  
UNIVERSIDADE FEDERAL DO PARÁ  
PRÓ-REITORIA DE DESENVOLVIMENTO E GESTÃO DE PESSOAL - PROGEP

**PORTARIA Nº \*\*NÚMERO GERADO AUTOMATICAMENTE\*\***


A PRÓ-REITORA DE DESENVOLVIMENTO E GESTÃO DE PESSOAL DA UNIVERSIDADE FEDERAL DO PARÁ, no uso das atribuições que lhe confere a Portaria nº 4515/2016 de 13.10.2016 do Magnífico Reitor e tendo em vista o que consta no Processo n. \*NÚMERO DO PROCESSO \* desta universidade,

**RESOLVE:**

Conceder **ABONO DE PERMANÊNCIA** instituído pela Emenda Constitucional n. 41/2003 LUCAS DE MELO SILVA, nos termos do art. 3º, da Emenda Constitucional n. 47/2005, publicada no DOU de 06/07/2005, com vigência a partir de 23 de Novembro de 2015.

Pró-Reitoria de Desenvolvimento e Gestão de Pessoal da Universidade Federal do Pará, Belém, 10 de Julho de 2018.

**KARLA ANDREZA DUARTE PINHEIRO DE MIRANDA**  
Pró-Reitora de Desenvolvimento e Gestão de Pessoal



Consulte a autenticidade dessa portaria no link abaixo:  
O LINK ESTARÁ DISPONÍVEL QUANDO A PORTARIA FOR CADASTRADA

Fonte: Novo SIPRO

Vale ressaltar que o documento em si não é editável. Na verdade, apenas os campos no lado direito são editáveis. O documento é gerado em tempo real com base no conteúdo dos campos de texto.

Na portaria de progressão por capacitação, a tabela de cursos realizados também tem o seu próprio campo. O resultado final é mostrado abaixo:

Figura 21 – Etapa de cadastro do conteúdo da portaria (progressão por capacitação)

Interessado(s) **Detalhes**

Tamanho da fonte: 13

Unidade: REITORIA

Primeiro parágrafo: ?

O VICE-REITOR DA UNIVERSIDADE FEDERAL DO PARÁ, no uso de suas atribuições, que lhe confere a Portaria nº 4514/2016 de 13.10.16 do Magnífico Reitor, e de acordo com o Processo nº

Segundo Parágrafo: ?

Conceder **PROGRESSÃO POR CAPACITAÇÃO PROFISSIONAL** de que trata o § 1º do Artigo 10 da Lei nº 11.091/2005, do **\*\*NÍVEL\*\*** para **\*\*NÍVEL\*\*** a partir de **\*\*DATA\*\***, LUCAS DE MELO SILVA, tendo em vista o § 4º do Artigo 10 da Lei supracitada, incluído pela Lei 12.772/2012, em virtude da conclusão dos eventos de aprendizagem discriminados abaixo:

Tabela: ?

1	Curso 1	12 hrs	10/07/2018
2	Curso 2	12 hrs	10/07/2018
3	Curso 3	12 hrs	10/07/2018

Local: Vice-Reitoria da Universi ?

Servidor a assinar: GILMAR PEREIRA DA SILV ?

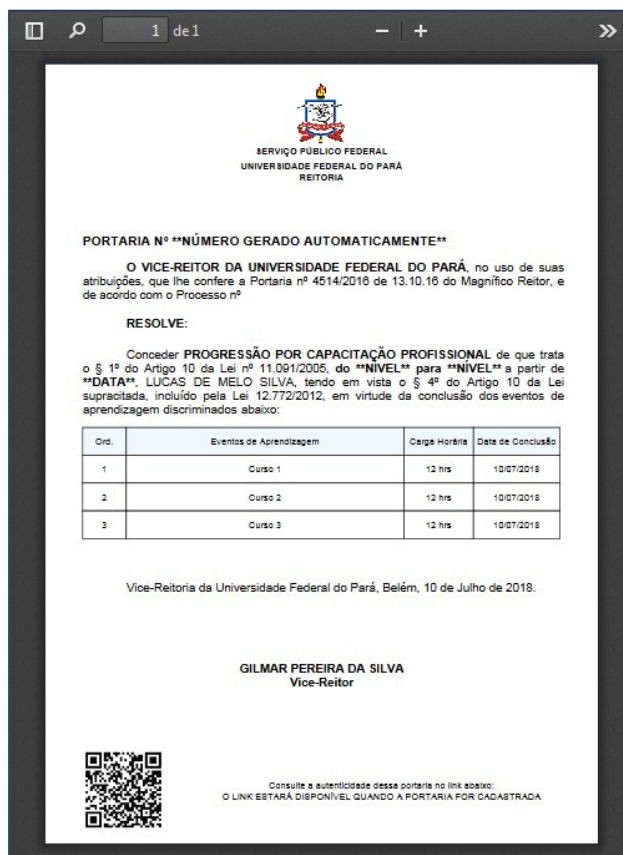
Designação do servidor a assinar: Vice-Reitor ?

Autor(a) do cadastro: LUCAS DE MELO SILVA

Data do cadastro: 10/07/2018

Cadastrar portaria e baixar cópia Salvar rascunho

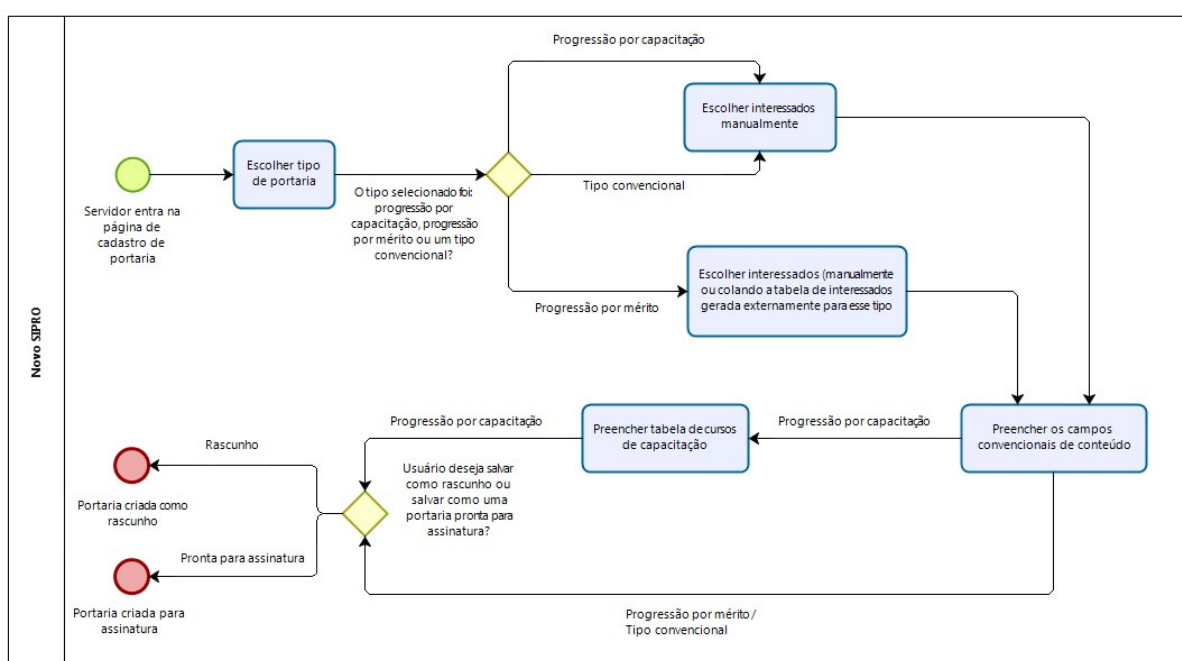
← Voltar



Vale ressaltar que o campo de “servidor a assinar” e “unidade” seguem os mesmos padrões de auto sugestão do campo de texto dos interessados. Assim, não é possível marcar para assinatura alguém que não é servidor da UFPA, e também não é possível cadastrar em uma unidade que não exista no banco de dados do SIG.

Finalizado o cadastro do conteúdo, o usuário pode escolher salvar a portaria como um rascunho ou salvar como uma portaria pronta para assinatura. Em ambos os casos há a possibilidade de edição posterior, porém um rascunho não pode ser assinado. O fluxo completo do novo cadastro de portarias é exibido abaixo:

**Figura 22 – Novo fluxo de criação de portarias**



Fonte: Novo SIPRO

Comparando o fluxograma acima com o fluxo antigo de portarias, a impressão passada é a de que no novo sistema, o processo é mais longo e com mais etapas para a criação de uma portaria. Porém, é importante ressaltar que todos os passos descritos acima são feitos dentro do novo SIPRO, e em uma página somente. Assim, na prática, o processo todo de criação de uma portaria leva pouco tempo, dependendo do tempo que o servidor que está criando a portaria leva para digitar o conteúdo e considerando que os templates de tipo de portaria preenchem quase tudo que é necessário.

### 3.2 Edição de portarias

O uso da edição de portarias é exatamente igual à criação de portarias: ambos utilizam a mesma página (XHTML) e os mesmos Managed Beans. A diferença é que na edição, os campos já estão todos preenchidos de acordo com a portaria escolhida

e o seu conteúdo (tipo de portaria, interessados, campos de texto). Assim, basta o usuário modificar o que for necessário. Quando a portaria editada é salva, uma nova versão é criada e inserida no banco de dados, assim, a versão anterior não se perde e a possibilidade de restaurar uma versão anterior fica aberta.

### 3.3 Assinatura de portarias

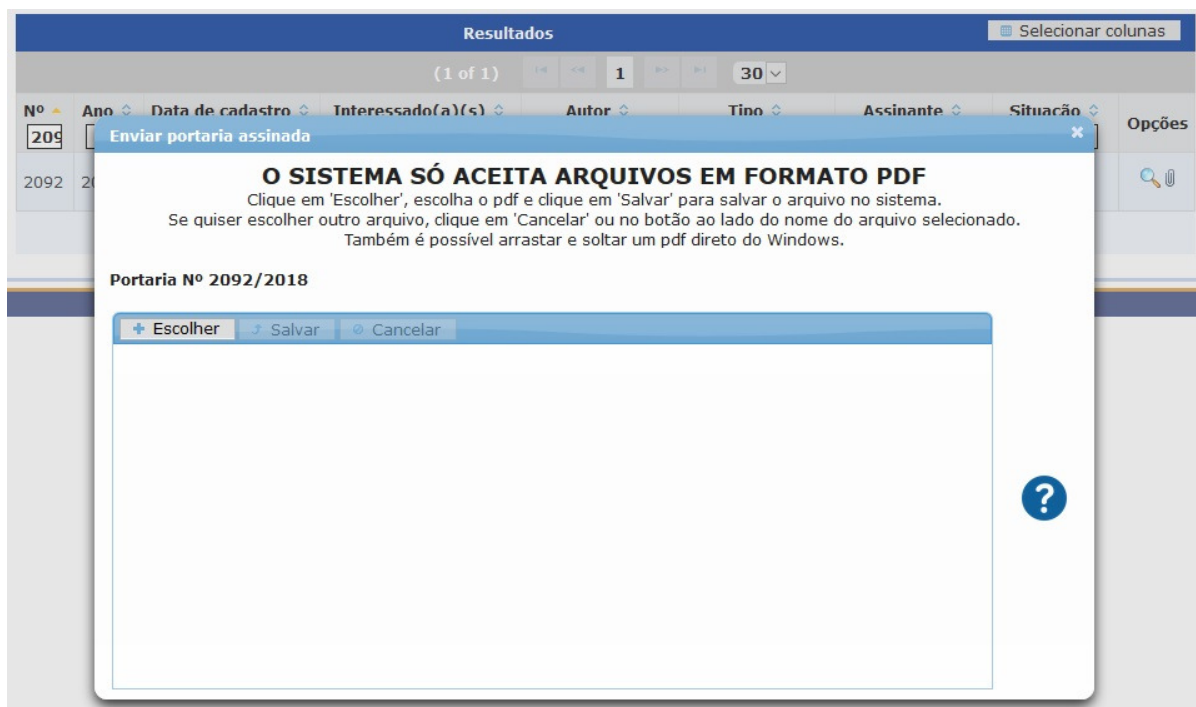
A primeira versão do novo SIPRO não possuía a delegação de tratar e verificar a assinatura das portarias. Ao invés disso, as portarias eram assinadas à mão pelo servidor responsável, e essa versão assinada era digitalizada e enviada para o sistema por upload. Então, o sistema não possuía uma funcionalidade de “assinatura”, apenas de armazenamento de documentos assinados, esses podendo ser assinados à mão ou digitalmente, isso ficava a critério de quem assinava.

**Figura 23 – Tela de “assinatura” de portarias**

Resultados									
Seletor de colunas									
(1 of 36) 1 2 3 4 5 6 7 8 9 10 30									
Nº	Ano	Data de cadastro	Interessado(a)(s)	Autor	Tipo	Assinante	Situação	Opções	
987	2018	01/03/2018		SUELLEN PATRICIA SILVESTRE RIBEIRO	DESIGNAÇÕES DIVERSAS	GILMAR PEREIRA DA SILVA	CRIADA	 	
988	2018	01/03/2018		SUELLEN PATRICIA SILVESTRE RIBEIRO	DISPENSA DE FUNÇÃO GRATIFICADA (FG)	GILMAR PEREIRA DA SILVA	CRIADA	 	
989	2018	01/03/2018		ANA CLAUDIA MELO BRAGA	TORNAR SEM EFEITO	GILMAR PEREIRA DA SILVA	CRIADA	 	
990	2018	02/03/2018		SUELLEN PATRICIA SILVESTRE RIBEIRO	DESIGNAÇÕES DIVERSAS	GILMAR PEREIRA DA SILVA	CRIADA	 	

Fonte: Novo SIPRO

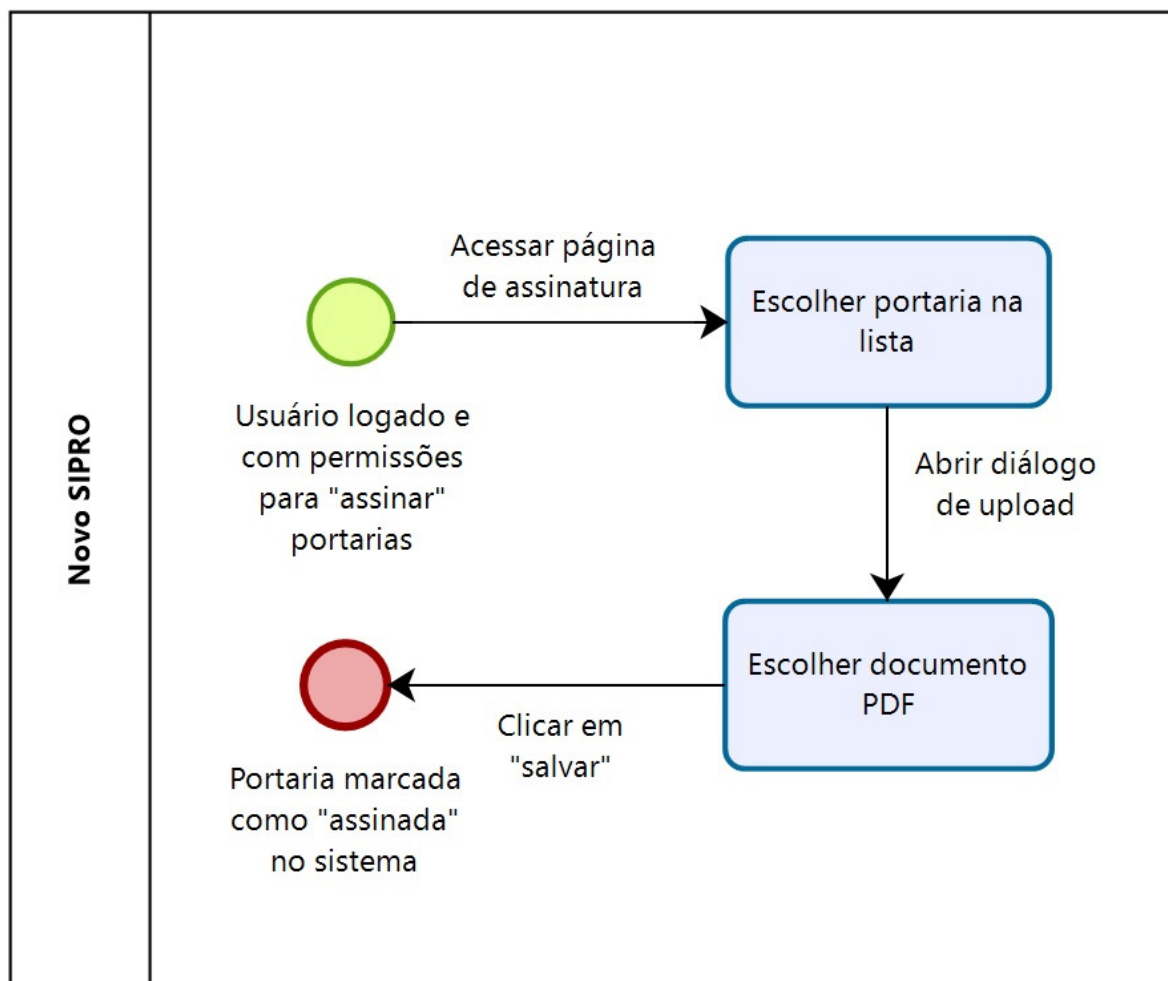
Figura 24 – Diálogo de enviar um PDF assinado para fazer a ligação com a portaria escolhida



Fonte: Novo SIPRO

Como essa funcionalidade é bem simples, tanto na implementação quanto na execução, o fluxo também conta com poucos passos. Entretanto, vale ressaltar que esses passos são os que são feitos apenas no sistema. Por fora, ainda existem várias etapas para a assinatura de uma portaria, incluindo download do documento, impressão, assinatura e digitalização, para então começar o fluxograma do sistema.

Figura 25 – Fluxograma da antiga funcionalidade de “assinar” portarias

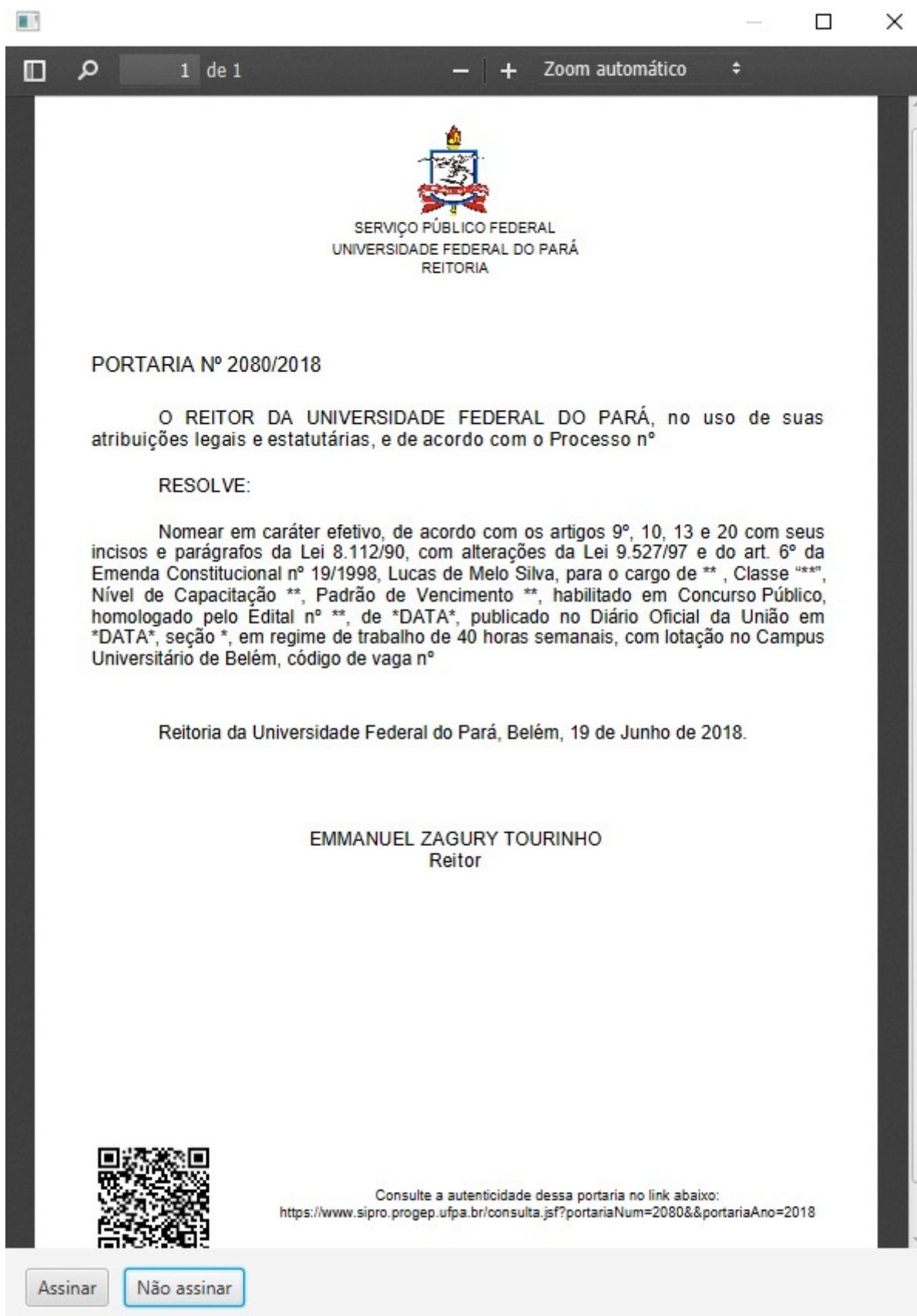


Fonte: Novo SIPRO


Já na segunda versão, por causa dos problemas apontados na assinatura manuscrita previamente (papel, tinta, fluxo não informatizado), foi implementado um componente a parte com a função de realizar assinaturas virtuais em portarias do SIPRO. A vantagem desse novo componente é a grande facilidade de assinar portarias: com acesso direto ao sistema SIPRO, não é mais necessário baixar cada portaria manualmente, pois o usuário ao logar já pode ver a sua lista de portarias para assinar, selecionar todas e assiná-las todas de uma vez só, apenas precisando digitar a senha do token USB (uma vez por sessão). Além disso, também é possível verificar o conteúdo de cada portaria antes de assinar. Assim, caso haja algum erro no documento, a pessoa pode reportar a situação para quem fez a portaria em questão.



Figura 27 – Assinatura de portarias com visualização prévia



1 de 1 Zoom automático

  
SERVIÇO PÚBLICO FEDERAL  
UNIVERSIDADE FEDERAL DO PARÁ  
REITORIA

PORTARIA Nº 2080/2018


O REITOR DA UNIVERSIDADE FEDERAL DO PARÁ, no uso de suas atribuições legais e estatutárias, e de acordo com o Processo nº

RESOLVE:

Nomear em caráter efetivo, de acordo com os artigos 9º, 10, 13 e 20 com seus incisos e parágrafos da Lei 8.112/90, com alterações da Lei 9.527/97 e do art. 6º da Emenda Constitucional nº 19/1998, Lucas de Melo Silva, para o cargo de \*\*, Classe "\*\*\*\*", Nível de Capacitação \*\*, Padrão de Vencimento \*\*, habilitado em Concurso Público, homologado pelo Edital nº \*\*, de \*DATA\*, publicado no Diário Oficial da União em \*DATA\*, seção \*, em regime de trabalho de 40 horas semanais, com lotação no Campus Universitário de Belém, código de vaga nº

Reitoria da Universidade Federal do Pará, Belém, 19 de Junho de 2018.

EMMANUEL ZAGURY TOURINHO  
Reitor



Consulte a autenticidade dessa portaria no link abaixo:  
<https://www.sipro.progep.ufpa.br/consulta.jsf?portariaNum=2080&&portariaAno=2018>

Assinar Não assinar

Fonte: Componente assinador do SIPRO

Na assinatura com visualização, o usuário pode escolher assinar ou não cada portaria, feita a revisão. Caso o usuário opte por assinar, o processo é feito automaticamente, incluindo posicionamento da assinatura visual e envio para o servidor, e em seguida, a próxima portaria da lista é exibida (se houver).

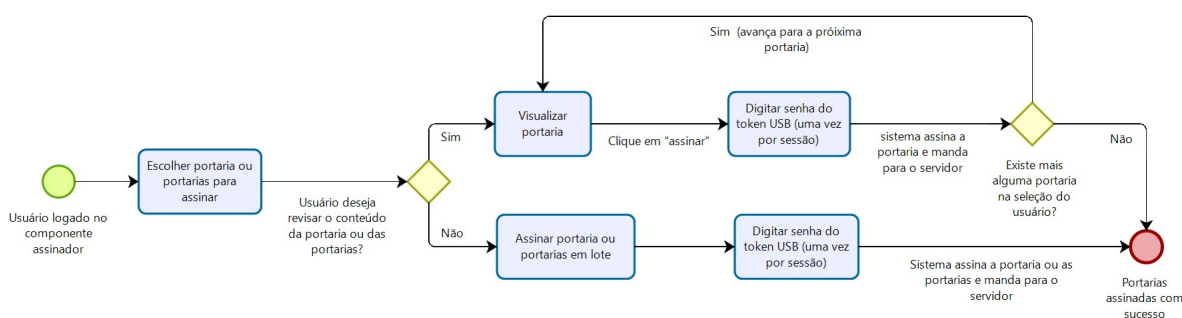
**Figura 28 – Assinatura automatizada de portarias**



Fonte: Componente assinador do SIPRO

Já na assinatura em lote, todas as portarias são assinadas de uma vez só, sem revisão do usuário. E, feito isso, as portarias assinadas são enviadas automaticamente para o servidor, que atrela o documento assinado à portaria selecionada.

**Figura 29 – Fluxograma de assinatura usando o componente assinador**



Fonte: Novo SIPRO

### 3.4 Boletim mensal

Comparado com as outras funcionalidades e as suas simplificações no fluxo antigo de portarias, a funcionalidade de publicar um boletim mensal foi a mais simplificada de todas. Considerando que o boletim anterior era em outro site, e o processo para gerar o boletim e publicar no portal da PROGEP era demorado, agora, o processo se resume a escolher o mês e o ano, e o boletim é gerado e publicado instantaneamente.

**Figura 30 – Novo método de publicação de boletim mensal**

Mês/ano:

Fonte: Novo SIPRO

## 4 Considerações finais

Este trabalho surgiu da demanda de uma nova forma de gerenciar portarias na universidade, em especial na PROGEP. Quando o fluxo de portarias antigo foi analisado, foram identificados diversas oportunidades de informatização, e com isso, essas oportunidades foram postas em prática com o lançamento do novo sistema. Entretanto, a informatização completa do fluxo de portarias não foi alcançada na primeira versão do sistema, que não abrangia a funcionalidade de assinatura de portarias de forma digital. Com o lançamento da segunda versão, o objetivo foi concluído com sucesso: as portarias são criadas e assinadas de forma digital, sem a necessidade de imprimir documentos. Porém, o módulo de assinaturas digitais possui uma limitação de sistema operacional: atualmente, só funciona em computadores com o sistema “Windows” instalado. Isso acontece por causa do token USB: o driver desses tokens, disponibilizados pela empresa “Certisign”, não está disponível para o sistema Linux (apenas para Mac OS e Windows). E, para desenvolver uma versão para Mac, é necessário um computador disponível com esse sistema, para testes. Entretanto, essa é uma exigência inviável para o contexto atual.

A criação do sistema e este trabalho estão intimamente ligados: neste trabalho foi descrito todo o processo intelectual de concepção do novo sistema de portarias da universidade. Isto é, todas as análises e discussões que foram feitas acerca do antigo e do novo fluxo de portarias de forma a obter uma implementação que, ao mesmo tempo, deveria conter facilidade de acesso e deveria englobar todo o fluxo de portarias, sem descartar nenhuma necessidade ou passo. E considerando todas essas discussões, foi possível implementar o sistema da maneira que foi projetado, cumprindo os objetivos estipulados.

Quanto aos resultados em valores, a ideia inicial era de fazer uma análise estatística do uso do sistema para achar algum padrão de melhoria entre o novo e o antigo fluxo de portarias. Entretanto, quando essa pesquisa foi posta em prática, foram identificadas diversas limitações, e, conseqüentemente, essa análise se tornou inviável, pois as estatísticas foram inconclusivas. O primeiro problema encontrado foi nas estatísticas de criação de portarias. Quando verificado o número de portarias criadas tanto no sistema antigo quanto no novo, a conclusão foi a de que não há como estabelecer uma relação direta entre a eficiência dos sistemas usados e o número de portarias criadas por dia. Isso porque a criação de portarias depende dos processos correspondentes, então, esse era o único fator que afetava e ainda afeta o cadastro de portarias. Dessa forma, haviam dias em que foram criadas duas portarias, da mesma forma que houveram dias com mais de quarenta portarias criadas. Não há um padrão

a ser explorado nessa estatística.

A segunda limitação encontrada na pesquisa foi em relação à assinatura de portarias. Atualmente, para acompanhar processos, é utilizado o SIPAC (Sistema Integrado de Patrimônio, Administração e Contratos). A estatística que estava sendo cogitada era a diferença do tempo de assinatura de uma portaria da forma convencional (impressão, assinatura manuscrita e digitalização) para a forma digital com o componente de assinatura. No novo SIPRO, esses dados são fáceis de ser obtidos, pois, os marcadores de tempo estão todos no banco de dados. Entretanto, na forma convencional, o SIPAC é a única forma digital de rastrear processos, e lá não consta nenhuma informação referente a encaminhamento de portarias, apenas do processo em si. Dessa forma, a análise estatística do tempo de assinatura de portarias também se tornou inviável.

Com isso, análises estatísticas foram descartadas. Restou uma análise de paradigmas organizacionais. Ou seja, descrever o que mudou em termos organizacionais após a implementação do sistema. Vale ressaltar que essa análise ficou limitada ao prédio da reitoria.

Depois do lançamento da primeira versão do sistema, a mudança imediata foi o uso reduzido de papel e de tinta para imprimir várias vias da portaria, pois agora só passou a ser necessária a via de assinatura, e para fazer o acompanhamento da situação da portaria, os interessados podiam acessar a parte pública do sistema. Considerando que, no fluxo antigo, cada portaria era impressa em três vias (uma para anexar ao processo, uma para arquivar internamente e uma para repassar para a unidade do servidor interessado), e no fluxo da primeira versão, apenas uma via era necessária, e economia de papel e tinta para a impressão de portarias chegou a cerca de 66%.

Outra mudança foi em relação ao trânsito de processo junto com a portaria para o assinante. Antigamente, o processo inteiro era encaminhado para quem era responsável pela assinatura, incluindo a portaria. Isso acontecia, pois, às vezes era necessário revisar o processo para verificar se estava tudo correto. Após o lançamento da primeira versão do novo SIPRO, isso não acontece mais. Agora o servidor entra no sistema e verifica as portarias que ele precisa assinar. Se houver alguma dúvida quanto à resolução, ele pode solicitar o encaminhamento do processo para revisão. Essa é uma mudança que, por enquanto, foi verificada somente no gabinete do reitor. Entretanto vale ressaltar que o reitor é quem mais assina portarias na UFPA. Logo, a tendência é que essa prática seja padronizada.

Com o lançamento da segunda versão, a primeira mudança foi a requisição de tokens USB para todos os servidores que assinam portarias, com o intuito de usarem o componente assinador. Entretanto, nem todos estão usando a assinatura digital ainda. Como atualmente os servidores assinantes estão em um período transitório do

método convencional para o uso do componente assinador, o sistema ainda aceita uploads de documentos pela interface web. Porém, a partir de uma data específica, a funcionalidade de upload de PDF será removida do sistema, e a única forma aceita de assinar portarias será usando o componente assinador. Com isso, a expectativa é de que o uso de papel e tinta chegue a zero no fluxo de portarias, e que o intervalo de tempo entre a criação de uma portaria e a sua assinatura também reduza drasticamente. Vale ressaltar que, mesmo com o fluxo totalmente informatizado, uma prática comum dos servidores ainda é de imprimir uma via da portaria assinada (mesmo digitalmente) para anexar ao processo. Entretanto, para o escopo do trabalho, essa prática não foi considerada, pois, a impressão da portaria não é mais necessária para o fluxo de portarias, é algo complementar e que pode ser substituído futuramente por uma referência ao número da portaria no processo, ao invés do documento em si. E com esse número, é possível a busca na parte pública do sistema.

Assim, tendo em base as informações apresentadas nesse trabalho, a conclusão é a de que o novo sistema de portarias da UFPA serviu como um exemplo prático para demonstrar a eficácia da informatização de processos. Além da centralização do fluxo de portarias no novo SIPRO e da simplificação das funcionalidades, essa informatização também promoveu uma maior rastreabilidade das portarias, resolvendo um problema muito antigo no fluxo de portarias, que é: a grande dificuldade de achar uma portaria quando o número não era conhecido. Esse problema foi resolvido com as consultas ricas em parâmetros que agora são possíveis. Adicionalmente, apenas nos primeiros meses, é importante destacar as mudanças organizacionais causadas pela implantação do sistema, visto que esse tipo de mudança costuma ocorrer apenas a médio ou longo prazo. Isso ocorreu em partes por causa da grande aceitação do sistema por todos os envolvidos diretamente no fluxo de portarias, que passaram por um curto período de treinamento. Esse é um exemplo de como a facilidade de uso de um sistema é tão importante quanto a informatização em si: se o novo SIPRO fosse complexo demais para ser usado, o fluxo de portarias poderia ser até mais demorado do que na forma antiga. Além disso, a aceitação dos servidores seria bem baixa, o que aumentaria ainda mais o período transitório de um fluxo para o outro, ou seja, o período em que os servidores abandonariam o fluxo antigo para se acostumar ao novo. Dependendo da complexidade, o sistema poderia até ser inviabilizado para uso, mesmo atingindo um nível de informatização completo.

Dado o sucesso da implementação do sistema, o mesmo foi apresentado no XII WTICIFES (XII Workshop de Tecnologia da Informação e Comunicação das Instituições Federais de Ensino Superior do Brasil), evento que “é uma promoção do Colégio de Gestores de TIC das IFES que, a partir da similaridade dos problemas e desafios de gestão, busca incentivar a troca interinstitucional de experiências e soluções para a efetivação da TIC como um instrumento de avanço de todas as Instituições.” (UNIVER-

SIDADE FEDERAL DA INTEGRAÇÃO LATINO-AMERICANA, 2018) e nesse evento, o sistema despertou um grande interesse por parte de outras instituições federais, que não possuem um sistema de assinaturas digitais em portarias, apenas eletrônicas.

#### 4.1 Trabalhos futuros

Para o sistema web, a melhoria mais imediata pensada seria a criação de uma versão *mobile*, visto que o sistema não possui suporte nativo para telas pequenas, pois, usa o mesmo esquema visual do SIG.

Outra possibilidade de melhoria é o módulo de assinaturas digitais: a primeira melhoria pensada para esse componente é a compatibilidade com outros sistemas operacionais como Linux ou Mac. Além disso, criar um sistema de notificações para avisar o usuário logado no módulo que uma nova portaria para assinatura está disponível.

De forma geral, a expectativa para o novo SIPRO é de que ele deixe de ser apenas um sistema de portarias e passe a possuir mais funcionalidades, que não estão diretamente ligadas às portarias. Atualmente, isso já vem sendo implementado com o novo módulo de estágio probatório, que é um período de cerca de três anos onde o servidor é avaliado por vários critérios como: responsabilidade, assiduidade, entre outros. Esse novo módulo, no seu lançamento, irá possibilitar o cadastro e consulta de avaliações de estágio probatório, além de gerar relatórios contendo listas de servidores com avaliação pendente e avaliação atrasada, o que atualmente é feito de forma manual. Dessa forma, o sucesso da implantação do novo SIPRO e a criação de novos módulos pode servir como um incentivo para o aumento da informatização em todos os setores da UFPA.

## Referências

ALEX, B. et al. *Spring Security Reference*. 2017. Disponível em: <<https://docs.spring.io/spring-security/site/docs/5.0.7.RELEASE/reference/htmlsingle/#history>>. Citado na página 39.

COSTA, F. *Diferença entre API e Web Service de maneira simples*. 2015. Disponível em: <<https://fxcosta.wordpress.com/2015/05/31/diferenca-entre-api-e-web-service-de-maneira-simples/>>. Citado na página 30.

HOOKOM, J. *Facelets*. 2005. Disponível em: <<http://hookom.blogspot.com/2005/05/facelets.html>>. Citado na página 34.

HOOKOM, J. *Facelets: Templating*. 2005. Disponível em: <<http://hookom.blogspot.com/2005/05/facelets-templating.html>>. Citado na página 35.

INTERNATIONAL TELECOMMUNICATION UNION. *2017 Global ICT Development Index*. 2017. Disponível em: <<http://www.itu.int/net4/ITU-D/idi/2017/index.html>>. Citado na página 11.

JAVATPOINT. *JSF - Managed Beans*. Disponível em: <<https://www.javatpoint.com/jsf-managed-beans>>. Citado na página 36.

KASPARY, A. J. *Redação Oficial - Normas e Modelos*. 19. ed. [S.l.]: Livraria Do Advogado, 2016. Citado na página 13.

LOWAGIE, B. *Digital Signatures for PDF documents*. 2012. Citado na página 31.

MIHALCEA, V. et al. *Hibernate ORM 5.3.5.Final User Guide*. Disponível em: <[http://docs.jboss.org/hibernate/orm/5.3/userguide/html\\_single/Hibernate\\_User\\_Guide.html](http://docs.jboss.org/hibernate/orm/5.3/userguide/html_single/Hibernate_User_Guide.html)>. Citado na página 40.

ORACLE. *Overview of the EL*. 2013. Disponível em: <<https://docs.oracle.com/javase/6/tutorial/doc/bnahq.html>>. Citado na página 38.

PI, K. *Different types of keystore in Java - Windows-MY*. 2016. Disponível em: <<https://www.pixelstech.net/article/1452337547-Different-types-of-keystore-in-Java---Windows-MY>>. Citado na página 44.

PORTAL DE COOPERAÇÃO - SINFO/UFRN. *Sobre a cooperação técnica*. Disponível em: <<http://www.portalcooperacao.info.ufrn.br/pagina.php?a=sobre>>. Citado na página 13.

UNIVERSIDADE FEDERAL DA INTEGRAÇÃO LATINO-AMERICANA. *XII WTICIFES - Sobre*. 2018. Disponível em: <<https://eventos.unila.edu.br/wticifes2018/sobre/>>. Citado na página 61.