

**UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

MÁRDENI FERREIRA DE SOUZA SÁ

**AUTOMATIZAÇÃO DO PROCESSO DE GESTÃO DO AUXÍLIO
SAÚDE EXECUTADO PELA SUGEP - UFRPE**

RECIFE- PE

2017

**UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO
MÁRDENI FERREIRA DE SOUZA SÁ**

**AUTOMATIZAÇÃO DO PROCESSO DE GESTÃO DO AUXÍLIO SAÚDE
EXECUTADO PELA SUGEP - UFRPE**

Projeto apresentado ao curso de Ciência da
Computação na Universidade Federal Rural de
Pernambuco – UFRPE, como requisito para a
obtenção do título de Bacharel em Ciência da
Computação
Orientador: Prof.º Lucas Albertins de Lima

RECIFE- PE

2017

**AUTOMATIZAÇÃO DO PROCESSO DE GESTÃO DO AUXÍLIO SAÚDE
EXECUTADO PELA SUGEP - UFRPE**

Projeto apresentado como requisito para a nota da disciplina: Trabalho de conclusão de curso, ministrado pelo Prof.º Marcelo Luiz Monteiro no curso de Ciência da Computação na Universidade Federal Rural de Pernambuco - UFRPE
Orientador: Prof.º Lucas Albertins de Lima

RECIFE- PE

2017



MINISTÉRIO DA EDUCAÇÃO E DO DESPORTO
UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO (UFRPE)
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

<http://www.bcc.ufrpe.br>

FICHA DE APROVAÇÃO DO TRABALHO DE CONCLUSÃO DE CURSO

Trabalho defendido por Márdeni Ferreira de Souza Sá como requisito para conclusão do curso de Bacharelado em Ciência da Computação da Universidade Federal Rural de Pernambuco, intitulado **Automatização do Processo de Gestão do Auxílio Saúde Executado pela SUGEP-UFRPE**, orientado por Lucas Albertins de Lima e aprovado pela seguinte banca examinadora:

Lucas Albertins de Lima
DEINFO/UFRPE

Ricardo André Cavalcante de Souza
DEINFO/UFRPE

André Aziz Camilo de Araújo
DEINFO/UFRPE

DOCUMENTO DE TRABALHO DE CONCLUSÃO DE CURSO

Identificação

Aluno(a): Márdeni Ferreira de Souza Sá (mardeniferreira@gmail.com)

Orientador(a): Lucas Albertins de Lima (lucas.albertins@gmail.com)

Título: Automatização do Processo de Gestão do Auxílio Saúde Executado pela SUGEP - UFRPE

Área de Concentração: Sistemas de Informação

Linha de Pesquisa: Sistemas de Gestão Pública

AGRADECIMENTOS

Agradeço primeiramente a Deus e aos meus pais, Márdeni Pereira e Angela Ferreira, pois sem o apoio incondicional deles não teria conquistado metade daquilo que tenho galgado.

Agradeço à minha companheira, Káthia Karine, por todo seu amor e companheirismo: estando sempre ao meu lado me dando forças durante o desenvolvimento deste trabalho e sempre acreditando em cada conquista minha, seja ela acadêmica ou profissional.

Agradeço também a todos meus colegas de turma que colaboraram de alguma forma nos mais diversos projetos, sobretudo meu companheiro Allan Monteiro, com o qual tive o prazer de desenvolver diversos trabalhos de alta qualidade, os quais contribuíram não só para a obtenção de ótimas notas, como para um excelente aprendizado acadêmico.

Agradeço ao meu orientador Lucas Albertins pela sua dedicação a este trabalho. Mesmo em meio aos prazos apertados não mediu esforços e se dispôs para ajudar nas orientações e revisão para aperfeiçoamento deste trabalho, até mesmo durante seu período de férias. Sem o seu empenho e disposição este trabalho não seria possível.

Agradeço a oportunidade que me foi dada, pois o desenvolvimento deste trabalho me conferiu uma imensa realização acadêmica e pessoal, não só pela possibilidade da conclusão do curso, mas pela possibilidade de colocar em prática os mais diversos conhecimentos relacionados à Engenharia de Software. Conhecimentos estes que atravessaram as principais fases da construção de um produto de software. Pouquíssimos discentes da área de Engenharia de Software têm a oportunidade de passar por este tipo de experiência ainda na graduação dentro de um único projeto. Desta forma, me sinto honrado pela oportunidade que me foi concedida.

Como se não bastasse esta realização acadêmica, a interação com os clientes do produto do software foi muito agradável. Principalmente, pela alegria e receptividade demonstrada pelos gestores do auxílio saúde, pois acreditavam no meu potencial em melhorar suas condições de trabalho. Esta alegria pôde ser notada nas reuniões: no sorriso dos servidores que me cercavam. Curiosos e ansiosos sempre perguntavam: “Quando o sistema vai ficar pronto?”; “O sistema vai fazer isso mesmo?” “Vamos te agradecer eternamente!”.

Nunca imaginei que teria tamanha satisfação durante e após a implantação de um sistema. O que inicialmente era apenas um trabalho acadêmico, acabou me conectando de uma forma muito especial a estas pessoas, as quais serei eternamente grato pela colaboração com o meu trabalho acadêmico. Estas demonstrações de afeto e alegria, sem dúvida, valem muito mais do que qualquer bem material.

RESUMO

A deficiência no tocante à automatização de tarefas administrativas ou de gestão dentro do setor público frente à iniciativa privada ainda é considerável no Brasil. Muitas atividades complicadas - cuja execução é extremamente mecânica e propensa a erros - são executadas apenas por humanos, quando na verdade deveriam ser realizadas com o apoio de sistemas computacionais específicos. Inserida neste contexto, está a SUGEP - órgão responsável por gerir os recursos humanos da UFRPE. A SUGEP é responsável pela gestão do auxílio saúde. Tal gestão envolve tarefas extremamente desgastantes aos servidores do referido órgão. Desta forma, com o objetivo de tornar o processo de gestão do auxílio saúde menos laborioso e menos propenso a erros, este trabalho propõe a construção de um sistema *web* para automatização das tarefas mais complicadas do processo. Tal sistema será fundamentado nos princípios da engenharia de software, sobretudo com base no processo de *software* RUP e inspirações nos valores ágeis. Esta fundamentação visa garantir a construção de uma solução que satisfaz às necessidades dos usuários e que viabiliza/facilita sua futura manutenção e continuidade. Além disto, este trabalho visa a realização da modelagem de negócio do processo de gestão do auxílio saúde e elaboração de uma especificação inicial para o sistema. Estas duas construções permitem que terceiros possam: ser introduzidos ao problema de forma que seja possível vislumbrar uma especificação completamente nova com base na modelagem de negócio realizada; construir um sistema completamente novo a partir da especificação inicial (por exemplo, com base numa tecnologia diferente da utilizada neste trabalho); ou simplesmente aumentar o nível de granularidade da especificação desenvolvida por este trabalho. Por fim, objetivando a validação da solução proposta, este trabalho ainda realiza a execução de um projeto piloto, o qual põe o sistema à prova numa situação real de utilização.

Palavras-chave: Sistema de Gestão Pública, Sistema Gerenciador de Conteúdo, *Drupal*, Engenharia de *Software*, RUP, Metodologias Ágeis.

ABSCTRACT

The deficiency in the automation of administrative or management tasks within the public sector in relation to private initiative is considerable in Brazil. Many activities whose execution is extremely mechanical, stressful, and error prone are performed only by humans, when in fact they should be performed with the support of specific computer systems. Included in this context is SUGEP - the department responsible for managing UFRPE's human resources. SUGEP is responsible for managing health care benefits. Such management involves extremely stressful tasks to the employees of this department. Thus, to make the health care management process less labor-intensive and less error-prone, this work proposes the construction of a web system to automate the more complicated tasks of the process. Such a system is based on the principles of software engineering, especially based on the RUP software process and inspirations on agile values. This rationale aims to ensure the construction of a solution that meets the needs of the users and that enables / facilitates their future maintenance and continuity. In addition, this work aims to perform the business modeling of the health care management process and elaboration of an initial specification for the system. These two constructs allow third parties to: be introduced to the problem so that it is possible to glimpse a completely new specification based on the business modeling performed; Construct a completely new system from the initial specification (for example, based on a technology different from that used in this work); Or simply to increase the level of granularity of the specification developed by this work. Finally, aiming at the validation of the proposed solution, this work executes a pilot project, which puts the system under test in real scenarios.

Keywords: Public Management System, Content Management System, Drupal, Software Engineering, RUP, Agile Methodologies.

LISTA DE FIGURAS

Figura 1 - Ilustração do processo de compra de refrigerante.	18
Figura 2 - Ilustração do processo de gestão do auxílio saúde sem o apoio de um sistema específico.	23
Figura 3 - Fases x disciplinas do RUP	39
Figura 4 - Ilustração de um diagrama de casos de uso.	44
Figura 5- Ilustração de um diagrama de classes.	46
Figura 6 Sistemática para automatização de processos.	60
Figura 7 Ilustração do processo de gestão do auxílio saúde automatizado pelo SAS.	66
Figura 8 - Diagrama de casos de uso para o SAS.	70
Figura 9 - Diagrama de classes conceitual para o SAS.	74
Figura 10- 1ª Pergunta do questionário de avaliação.	93
Figura 11- 2ª Pergunta do questionário de avaliação.	93
Figura 12- 3ª Pergunta do questionário de avaliação.	94
Figura 13- 4ª Pergunta do questionário de avaliação.	94
Figura 14- 5ª Pergunta do questionário de avaliação.	95
Figura 15- 6ª Pergunta do questionário de avaliação.	95
Figura 16- 7ª Pergunta do questionário de avaliação.	96
Figura 17- 8ª Pergunta do questionário de avaliação.	96
Figura 18- 9ª Pergunta do questionário de avaliação.	97
Figura 19 -10ª Pergunta do questionário de avaliação.	97
Figura 20 - Ilustração do processo de gestão do auxílio saúde sem o apoio de um sistema específico.	111
Figura 21 Ilustração do processo de gestão do auxílio saúde automatizado pelo SAS.	115
Figura 22 Modelo Cebola.	120
Figura 23 Diagrama de casos de uso.	121
Figura 24 Tela de <i>login</i>	129
Figura 25 Tela inicial - visão do beneficiário do auxílio saúde.	130

Figura 26 Tela inicial – visão do gestor do auxílio saúde.	130
Figura 27 Tela de cadastro de beneficiário do auxílio saúde - parte 1.	131
Figura 28 Tela de cadastro de beneficiário do auxílio saúde - parte 2.	132
Figura 29 Tela de cadastro de beneficiário do auxílio saúde - parte 3.	132
Figura 30 Tela para gerenciamento de todos beneficiários cadastrados no sistema.	133
Figura 31 Tela Adicionar Dependente – parte 1.....	133
Figura 32 Tela Adicionar Dependente - parte 2.	134
Figura 33 Tela para gerenciamento dos dependentes associados à pessoa que está acessando o sistema.	134
Figura 34 Tela para gerenciamento de todos dependentes cadastrados no sistema.	135
Figura 35 Tela de envio de comprovante de pagamento.	135
Figura 36 Tela para acompanhamento dos comprovantes enviados.	136
Figura 37 Tela para gerenciamento/avaliação dos comprovantes enviados – parte 1.....	136
Figura 38 Tela para gerenciamento/avaliação dos comprovantes enviados - parte 2.	137
Figura 39 Tela para gerenciamento/avaliação dos comprovantes enviados - parte 3.	137
Figura 40 Tela para gerenciamento/avaliação dos comprovantes enviados - parte 4.	138
Figura 41 Tela para gerenciamento dos prazos de envio de comprovante.	139
Figura 42 Tela com a carga batch gerada num editor WYSIWYG.....	140

LISTA DE TABELAS

Tabela 1 - Cronograma de execução das atividades.....	32
Tabela 2 - Questionário de avaliação do SAS.....	88

LISTA DE ABREVIATURAS, SIGLAS E SÍMBOLOS

SAS	Sistema Auxílio Saúde - construído neste trabalho.
NTI	Núcleo de Tecnologia da Informação da UFRPE
SUGEP	Superintendência de Gestão de Pessoas da UFRPE
SIAPE	Sistema Integrado de Administração de Recursos Humanos

SUMÁRIO

CAPÍTULO I - INTRODUÇÃO

1.1 O CONTEXTO DA PESQUISA	18
1.2 PROBLEMA DE PESQUISA	22
1.3 HIPÓTESE	22
1.4 JUSTIFICATIVA	22
1.5 OBJETIVOS	28
1.5.1 Objetivo Geral	28
1.5.2 Objetivos Específicos	28
1.6 METODOLOGIA	28
1.7 CRONOGRAMA	32

CAPÍTULO II - FUNDAMENTAÇÃO TEÓRICA

2.1 SISTEMAS DE APOIO À GESTÃO PÚBLICA	34
2.2 TRABALHOS RELACIONADOS	35
2.3 RUP E METODOLOGIAS ÁGEIS	36
2.3.1 RUP	36
2.3.1.1 RUP - Conceitos Fundamentais	37
2.3.1.2 RUP - Fases x Disciplinas	38
2.3.1.3 RUP – Fases	39
2.3.1.4 Disciplinas	41
2.3.1.4.1 Modelagem de Negócio	41
2.3.1.4.2 Elicitação de Requisitos	42

2.3.1.4.3 Análise e Projeto	44
2.3.1.4.4 Implementação	46
2.3.1.4.5 Teste	47
2.3.1.4.6 Implantação	47
2.3.1.5 RUP - Conclusão.....	48
2.3.2 Valores Das Metodologias Ágeis.....	49
2.3.2.1 RUP Simplificado: RUP + Valores Ágeis	49
2.4 SISTEMAS GERENCIADORES DE CONTEÚDO (SGC) E <i>DRUPAL</i>	51
2.4.1 Tipos de SGC	52
2.4.2 <i>DRUPAL</i>	54
2.4.2.1 Arquitetura do <i>Drupal</i>	55
2.5 LINGUAGEM DE PROGRAMAÇÃO PHP.....	56

CAPÍTULO III – DESENVOLVIMENTO

3.1 SISTEMÁTICA PARA AUTOMATIZAÇÃO DE PROCESSOS.....	59
3.2 APLICAÇÃO DA SISTEMÁTICA À REALIDADE DA SUGEP	63
3.2.1 Concepção.....	63
3.2.1.1 Descoberta do problema da SUGEP.....	63
3.2.1.2 Modelagem de negócio	64
3.2.1.2.1 Processo Antes x Processo Depois.....	65
3.2.1.3 Elicitação de requisitos – inicial	69
3.2.2 Elaboração.....	71
3.2.2.1. Elicitação de Requisitos – Refinamento	71

3.2.2.2 Protótipo.....	72
3.2.2.3 Análise e Projeto	73
3.2.3 Construção	75
3.2.3.1 Ambiente de desenvolvimento.....	76
3.2.3.2 Módulos do núcleo utilizados	77
3.2.3.3 Módulos de contribuidores utilizados.....	78
3.2.3.4 Dificuldades com módulos de contribuidores.....	79
3.2.3.5 Módulos customizados escritos.....	80
3.2.4 Transição.....	81
3.2.4.1 Implantação	81
3.2.4.2 Treinamento	82
3.3 LIÇÕES APRENDIDAS (SEÇÃO SECUNDÁRIO).....	83

CAPÍTULO IV - AVALIAÇÃO DOS RESULTADOS

4.1 AVALIAÇÃO DO SISTEMA.....	85
4.1.1 Planejamento	85
4.1.2 Metodologia	87
4.1.3 Execução	91
4. 1.4 Resultados	91
4.1.4.1 Resultado do Questionário	92
4. 1.4.2 Dúvidas enviadas por e-mail.....	98
4.1.5 Conclusões	98

CAPÍTULO V – CONCLUSÃO

5.1 CONSIDERAÇÕES FINAIS.....	101
5.3 DIFICULDADES ENCONTRADAS.....	104
REFERÊNCIAS.....	105
APÊNDICE I – DOCUMENTO DE REQUISITOS.....	108
APÊNDICE II – TELAS.....	129

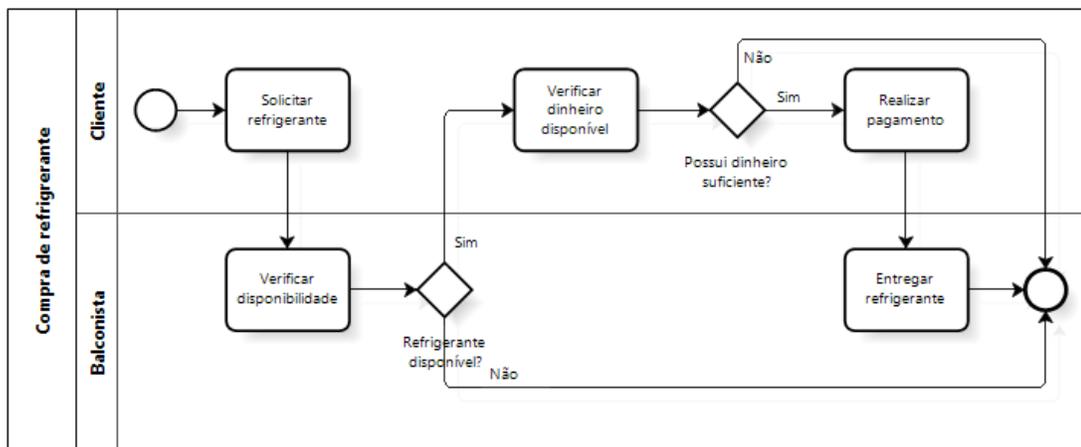
CAPÍTULO I - INTRODUÇÃO

Este capítulo apresenta os motivos que encabeçaram o desenvolvimento desta pesquisa tecnológica, bem como a justificativa para dedicação de esforços ao desenvolvimento da mesma.

1.1 O CONTEXTO DA PESQUISA

Antes de discutirmos o problema abordado, é importante compreendermos a definição de processo dentro do contexto desta pesquisa. Segundo Georgakopoulos, Hornick e Sheth (1995), um processo pode ser definido como um conjunto de atividades coordenadas com o propósito de alcançar algum objetivo ou gerar valor. A Figura 1 ilustra um processo simples, que consiste na compra de um refrigerante. Este processo é composto por algumas atividades básicas, como por exemplo: “Verificar se há dinheiro disponível para concluir a compra” – executada pelo cliente – e a entrega do refrigerante – executada pelo balconista.

Figura 1 - Ilustração do processo de compra de refrigerante.



Fonte: (BLOG IPROCESS, 2012)

Podemos citar um exemplo bem simples de um processo dentro de uma organização pública. Podemos imaginar a realização de uma consulta médica, dentro de um hospital público como um processo. Este processo envolveria as seguintes atividades: consultar guia médico (realizada pelo paciente); marcar consulta por telefone (executada pelo paciente em conjunto com o telefonista), esperar o atendimento médico (executada pelo paciente), avaliar o paciente (executado pelo médico).

A partir deste entendimento, é possível perceber que qualquer organização, seja ela pública ou privada, possui diversos processos que geram valor e apoiam seu funcionamento. Logo, dependem da eficiência destes processos para garantir o seu sucesso. De acordo com a publicação *Business Process Management* (WESKE, 2012), um processo maduro e eficiente é capaz de melhorar a satisfação do cliente com a organização, reduzir os custos operacionais e, conseqüentemente, aumentar o lucro/geração de valor. Sendo assim, os processos passam a ser um dos ativos mais importantes de uma organização e devem ser tratados com bastante cuidado.

Os sistemas de informação, por sua vez, são responsáveis pela automatização de vários processos dentro das organizações. Sua utilização permite a execução automática de várias atividades repetitivas e exaustivas ao ser humano (DINIZ, 2005). Esta automatização proporciona ao seu usuário uma comodidade considerável permitindo que ele direcione seus esforços para outras atividades que podem gerar ainda mais valor para organização para a qual presta seus serviços.

Dentro deste contexto, vale lembrar que há algumas décadas atrás, os processos executados no serviço público brasileiro eram puramente manuais, ou seja, não utilizavam nenhum tipo de sistema computacional para automatizá-los. Mais recentemente, segundo Diniz (2005), a introdução dos computadores pessoais nas organizações permitiu que algumas atividades - que antes levavam horas para serem finalizadas - passassem a serem resolvidas mais rapidamente através de aplicações básicas - como planilhas eletrônicas, editores de texto, calculadoras, entre outras.

Apesar deste avanço, é importante destacar que ainda há muitos processos dentro do setor público que, de tão particulares, precisam ser apoiados por aplicações construídas especificamente para atender àquelas necessidades, pois as aplicações básicas supracitadas não

são suficientes para atender à demanda. Nesse seguimento, podemos mencionar algumas aplicações específicas que foram sendo introduzidas no setor público para agilizar alguns processos (DINIZ, 2005).

A primeira delas surgiu na década de 1990 quando o Governo Federal implementou a Declaração de Imposto de Renda eletrônica. Esta aplicação é utilizada até hoje pelos cidadãos brasileiros para apoiar o processo de arrecadação do imposto de renda. Ela caracteriza-se por ser eficiente, confiável e pouco suscetível a erros - por parte do contribuinte - durante o preenchimento do formulário eletrônico. Além disso, graças a implantação do sistema, o processamento dos dados das Declarações de Imposto de Renda foi diminuído drasticamente.

Como segundo exemplo de aplicação específica, temos o Plantão Eletrônico que surgiu nos anos 2000. A aplicação permite ao cidadão a realização do registro de ocorrências policiais simples através da internet. Isto trouxe comodidade para o cidadão que não precisa deslocar-se até uma delegacia para prestar pequenas queixas, além de diminuir a espera pelo atendimento nas delegacias. De forma geral, o processo de registro de ocorrências foi melhorado substancialmente após da implantação do sistema informatizado. Após a implantação desta aplicação, várias outras do gênero surgiram em diversos estados, como foi o caso da “Delegacia pela internet” no estado de Pernambuco (PERNAMBUCO, 2016).

Além dos sistemas mencionados acima, podemos citar mais dois que se aproximam do contexto deste trabalho. Estes foram construídos sob demanda para atender às necessidades de duas universidades federais, são eles:

- O módulo de saúde do sistema da Universidade Federal do Rio Grande do Norte - UFRN (LIMA e NETO, 0000);
- O sistema de submissão de boletos referentes a planos de saúde da Universidade Federal do Pará;

Ambos os sistemas automatizam o processo de gestão do auxílio saúde – o processo alvo desta pesquisa e que será descrito em detalhes ao final desta seção. Por ora, basta sabermos que os dois sistemas facilitam o trabalho do corpo administrativo das universidades, ao automatizar algumas atividades do processo, e confere comodidade aos beneficiários, ao permitir a submissão de documentos através da internet.

Infelizmente, a realidade do setor público brasileiro, em geral, frente a esse tipo de automatização - aquisição/construção/operação de sistemas/aplicações para apoiar seus processos - ainda é bastante precária (DINIZ, 2005). Para a maioria dos processos, ou não há sistemas disponíveis ou os existentes possuem muitas limitações. Este é um problema que se estende a todas as esferas do poder público seja ela federal, estadual ou municipal.

A realidade descrita também se estende ao órgão público alvo desta pesquisa tecnológica, a Superintendência de Gestão e Desenvolvimento de Pessoas (SUGEP). Este órgão pertence à Universidade Federal Rural de Pernambuco (UFRPE) e possui a incumbência de gerenciar os recursos humanos da referida universidade. A SUGEP possui vários processos que apoiam o seu funcionamento e boa parte deles ainda são executados de forma manual ou apenas com auxílio de algumas ferramentas de escritório básicas, como planilhas e editores de texto.

Dentre os processos mencionados está o processo alvo desta pesquisa tecnológica: o processo de gestão do auxílio saúde dos servidores da UFRPE. Para uma melhor contextualização, é importante saber que todos servidores públicos federais, esfera na qual a SUGEP atua, fazem jus ao benefício que tem como objetivo ressarcir parte do valor gasto pelo servidor com seu plano de saúde individual ou familiar.

É papel da SUGEP controlar e entregar o benefício a cada servidor - ativo ou inativo - ou pensionista. Assim como outros processos dentro do órgão, este é feito manualmente (sem o apoio de um sistema específico), envolvendo o manuseio de uma grande quantidade de documentos físicos (papéis) e uma imensa quantidade de dados. Esta característica acaba tornando o processo bastante mecânico e propenso a erros.

O processo supracitado demanda muita energia e tempo dos funcionários da SUGEP, os quais ficam impedidos de executarem outras tarefas igualmente importantes dentro do órgão. Este problema acaba se estendendo aos usuários do benefício, que por conta das condições de trabalho dos funcionários da SUGEP, acabam não tendo um serviço célere e transparente. A universidade também é afetada a partir do momento que tem seus funcionários, os da SUGEP, atrelados a uma atividade extremamente dispendiosa, impedidos de gerar mais valor para a instituição através das suas outras atribuições.

Visando solucionar o problema apresentado, este trabalho tem como principal objetivo propor uma forma de agilizar o processo de gestão deste benefício, através da especificação e

implantação de um novo processo, automatizado por um sistema computacional. O novo processo visa diminuir o tempo despendido pelos servidores da SUGEP nas suas tarefas, para que os mesmos possam dedicar seus esforços a outras atividades que, por sua vez, podem gerar mais valor às pessoas servidas pelo órgão. Além disso, este trabalho propõe, após a definição do processo e construção de um arcabouço do sistema, uma comparação qualitativa entre a eficiência do processo antigo (manual) e o novo (automatizado pelo sistema) a fim de validar a solução proposta.

1.2 PROBLEMA DE PESQUISA

Como melhorar a eficiência do processo de gestão do auxílio saúde da SUGEP/UFRPE de forma a facilitar as atividades desempenhadas pelos beneficiários do auxílio saúde e minimizar o tempo e esforço da equipe administrativa deste órgão?

1.3 HIPÓTESE

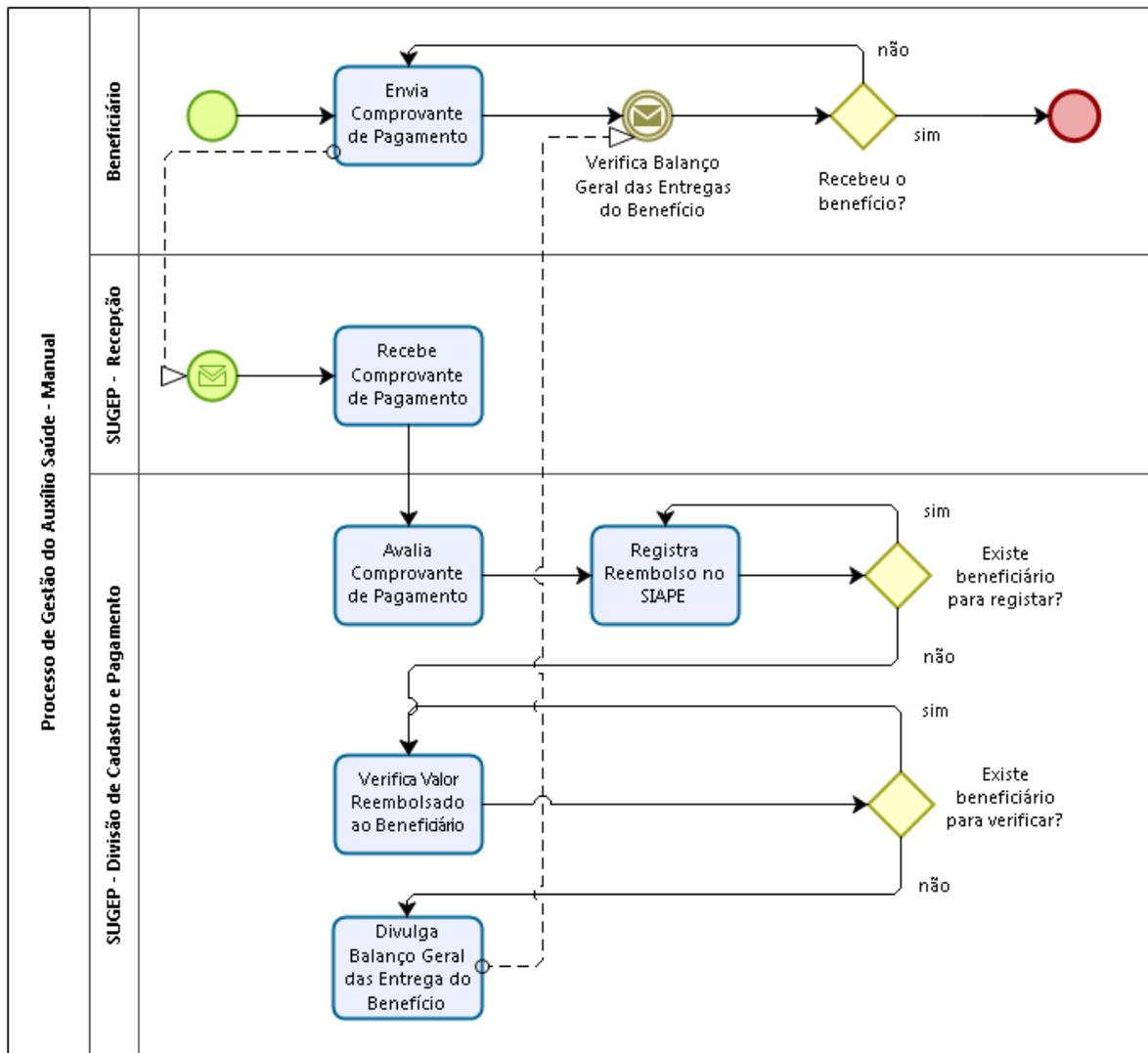
Acreditamos que é possível melhorar a eficiência através da reengenharia do processo de gestão do auxílio saúde que passará a ser apoiado por um sistema de informação de propósito específico, o qual será responsável pela automatização de várias tarefas dos diversos envolvidos.

1.4 JUSTIFICATIVA

Para entendermos melhor o problema abordado e sua escolha para este trabalho, é importante conhecermos as dificuldades apresentadas pela atual abordagem para o processo de gestão do auxílio saúde realizado pela SUGEP.

O diagrama da Figura 2 ilustra o processo de gestão do auxílio saúde manual, ou seja, sem o apoio de um sistema computacional customizado. Em seguida, o diagrama é explicado para que se possa entender bem o processo e os problemas desta abordagem.

Figura 2 - Ilustração do processo de gestão do auxílio saúde sem o apoio de um sistema específico.



Fonte: Elaborada pelo autor

Etapas/Atividades do processo de gestão do auxílio saúde manual:

- 1. Envia Comprovante de Pagamento:** Mensalmente, mais de mil beneficiários (as pessoas que fazem jus ao benefício) entregam aos servidores da SUGEP seus respectivos boletos e comprovantes de pagamento de planos de saúde, referentes ao mês atual e/ou meses anteriores, além de um formulário padrão do auxílio preenchido manualmente, os quais são entregues pessoalmente na SUGEP.
- 2. Recebe Comprovante de Pagamento:** A recepção da SUGEP acolhe os comprovantes enviados e repassa para o setor de Divisão de Cadastro e Pagamento.
- 3. Avalia Comprovante de Pagamento:** Para cada documentação entregue, os gestores do auxílio saúde verificam se tudo está dentro dos padrões. Então, se tudo estiver correto, o beneficiário que entregou tal documentação é anotado numa planilha, para que posteriormente possa ser ativado, recebendo o benefício no mês seguinte.
- 4. Registra Reembolso no SIAPE:** Para cada beneficiário que entregou a documentação corretamente, o gestor do auxílio saúde acessa o sistema do Governo Federal referente à folha de pagamento (SIAPE) e informa se o beneficiário deve ser ativado no benefício, ou seja, deve receber o reembolso do valor gasto com o plano de saúde.
- 5. Verifica Valor Reembolsado ao Beneficiário:** Esta atividade acontece após a ativação dos beneficiários. Para cada beneficiário verifica-se o valor que o SIAPE creditou em sua folha de pagamento por conta do benefício. Estes valores são utilizados para geração do relatório mensal que precisa ser redigido pelo gestor do auxílio saúde.
- 6. Divulga Balanço Geral das Entregas do Benefício:** Enfim, depois de gerar o relatório, a SUGEP divulga o balanço geral de todos valores pagos aos beneficiários.

7. **Verifica Balanço Geral das Entregas do Benefício:** Logo após o a divulgação do balanço geral, o beneficiário verifica se o benefício foi adicionado a sua folha de pagamento. Caso não, o beneficiário realiza o reenvio da documentação.

Esta abordagem gera, tanto aos gestores quanto aos beneficiários do auxílio saúde, uma série de problemas. Tais problemas são apresentados abaixo e estão agrupados de acordo com as etapas/atividades do processo.

- **Envia Comprovante de Pagamento:** O beneficiário precisa interromper suas atividades para se deslocar até o órgão, a fim entregar a documentação. A UFRPE acaba sendo prejudicada quando o beneficiário é servidor ativo, pois o mesmo deixa de prestar os seus serviços à universidade mesmo que temporariamente.

Uma grande quantidade de pastas precisa ser mantida dentro da SUGEP para armazenar toda a documentação entregue. Além da restrição do espaço físico de trabalho que esta condição tem causado ao longo do tempo, sabe-se que os documentos físicos estão sujeitos a desgaste, catástrofes naturais/criminosas ou acidentes como incêndios, por exemplo.

- **Avalia Comprovante de Pagamento:** Existe certa limitação referente à busca de comprovantes para que seja feita a análise, justamente por causa da forma de armazenamento. Para a realização da análise ou criação de relatórios, os documentos precisam ser buscados manualmente em um sistema de arquivo físico, o que geralmente demanda um tempo considerável, principalmente quando se trata de centenas de documentos. Fatalmente, muitos erros acontecem por conta do “fator humano” e, eventualmente, alguns beneficiários não têm sua documentação avaliada já que a documentação está sujeita à perda.

Como se não bastasse, o beneficiário recebe tardiamente o *feedback* sobre o status do comprovante de pagamento submetido. Quando algum comprovante se perde, eventualmente o beneficiário não consegue reenviá-lo a tempo.

- **Registra Reembolso no SIAPE:** Aqui é feito um trabalho extremamente maçante e cansativo. É preciso navegar várias vezes entre as telas do sistema de folha de

pagamento para informar, um a um, os beneficiários (mais de mil) que devem ser ativados no benefício. Esta atividade é uma das mais desgastantes, quando se trata de consumo de energia e tempo do pessoal da SUGEP.

- **Verifica Valor Reembolsado ao Beneficiário:** Após a finalização da ativação dos beneficiários, o sistema do Governo Federal de folha de pagamento calcula o valor devido aos beneficiários por conta do auxílio saúde, de acordo com suas regras. Então, os gestores do auxílio saúde precisam navegar até a página de cada um dos beneficiários para recuperar esses valores. Em seguida, com bases nesses valores, geram um relatório contendo a relação de todos beneficiários e seus dependentes, juntamente com o valor pago a cada um dos beneficiários.

Esta atividade, juntamente com a atividade “Registra Reembolso no SIAPE”, compõe a parte mais desgastante de todo o processo. Juntas, estas atividades alocam aproximadamente três servidores e levam alguns dias para serem finalizadas. Estas duas atividades são responsáveis por tornar o processo de gestão do auxílio saúde extremamente mecânico e propenso a erros, além de comprometer as outras atribuições dos gestores do auxílio saúde.

Conhecida a situação em que se encontra atualmente o processo de gestão do auxílio saúde, diante de todas as dificuldades apresentadas, é levantada a seguinte questão: “É possível melhorar a eficiência do processo de gestão do auxílio saúde da SUGEP, conferindo aos funcionários do órgão a possibilidade de aplicar seus esforços a outras atribuições igualmente relevantes para o próprio órgão e a UFRPE?”.

Para responder a esta pergunta, este trabalho propõe como solução, o desenvolvimento de uma sistemática para automatização de processos e utiliza esta sistemática para especificar um novo processo de gestão para o auxílio saúde, automatizado por um sistema de informação *web*. O processo será idealizado para apoiar a gestão do auxílio saúde, automatizando as tarefas mais dispendiosas e facilitando as demais tarefas do processo.

Além disso, este trabalho propõe a execução de testes de validação do novo processo com os afetados pela mudança, através da execução de um projeto piloto. Estes testes ajudarão afirmar

a solução desenvolvida como satisfatória aos problemas apresentados pelo processo antigo e também possui a finalidade de identificar possíveis defeitos e melhorias do novo processo, com a finalidade de se entregar o melhor produto possível, considerando as restrições de custo, escopo e prazo definidos para este trabalho.

A princípio, acreditamos que uma solução automatizada pode eliminar vários dos problemas descritos até então. Acima de todos os problemas, esta solução possui um grande potencial para diminuir drasticamente a quantidade de passos do processo, o tempo despendido e quantidade de servidores alocados para gestão do benefício. Ademais, acreditamos que a implementação do novo processo, além das vantagens apresentadas:

- Melhorará a comunicação do processo, permitindo que o beneficiário fique ciente do estado da sua submissão, podendo submeter novamente a documentação sem prejuízos ao recebimento do benefício;
- Diminuirá consideravelmente a possibilidade de perda de documentos, entre outras falhas humanas;
- Permitirá aos servidores da UFRPE, sobretudo os da SUGEP, concentrar seu tempo disponível e energia em atividades que podem gerar ainda mais valor para os recursos humanos da UFRPE.

Tal processo bem como o sistema que o apoiará devem ser construídos com base em conceitos e técnicas da Engenharia de Software, como a modelagem, implementação, testes e implantação de sistemas de forma a apoiar o processo em questão e não gerar novos problemas, como por exemplo: dificuldade de manutenção, inconformidade com as reais necessidades dos usuários, dificuldade de utilização, baixa eficiência, entre outros.

1.5 OBJETIVOS

1.5.1 Objetivo Geral

Especificar e avaliar um processo de negócio no setor público, mais especificamente a gestão do auxílio saúde gerido pela SUGEP, com o suporte de um sistema de informação de propósito específico. Tal processo objetiva a diminuição da carga de trabalho mecânico, aumentando assim a disponibilidade dos servidores do referido órgão e, conseqüentemente, permitindo de geração de mais valor destes servidores para a UFRPE.

1.5.2 Objetivos Específicos

- Desenvolvimento de uma sistemática para automatização de processos manuais. Tal sistemática pode ser replicada em outros contextos para a automatização de processos dentro da SUGEP ou no setor público de uma forma geral.
- Criar a modelagem de negócio referente ao processo de gestão do auxílio saúde. Esta modelagem permitirá que pessoas interessadas no funcionamento do processo compreendam-no fim-a-fim. Esta compreensão pode guiar os interessados a especificações e soluções ainda mais sofisticadas que a proposta neste trabalho.
- Gerar especificação conceitual do sistema. Tal modelagem pode permitir a construção de um sistema completamente novo, diferente do proposto por este trabalho, apenas com base nesta especificação.
- Validar o novo processo desenvolvido com os afetados pelo sistema de forma a confirmar a importância da sua implantação.

1.6 METODOLOGIA

Esta seção descreve a metodologia e técnicas de pesquisa, deixando claro como o trabalho será conduzido através das etapas do processo de investigação.

Esta pesquisa possui caráter tecnológico e, portanto, consiste na aplicação de conhecimentos de Engenharia de Software, adquiridos durante a graduação, para a construção de um Sistema de Informação desenvolvido para um cliente real, a SUGEP, e que será implantado e utilizado num ambiente real. Para atingir o objetivo desta pesquisa, as seguintes etapas serão consideradas:

- 1 Revisão Bibliográfica:** Pesquisar referências que abordam o tema da pesquisa e alguns conceitos relacionados – como automatização de processo, sistemas de informação, sistemas de informação na gestão pública - e o material técnico que guiará o desenvolvimento, ao longo de toda pesquisa tecnológica. Serão considerados como relevantes para a revisão, manuais de boas práticas de desenvolvimento de software, livros e artigos sobre as técnicas de engenharia de software que serão utilizadas para modelagem, implementação e validação de sistemas. Além disto, será necessário descobrir e analisar sistemas de gestão de auxílio saúde existentes a fim de descobrir características que possam ser aproveitadas, melhoradas ou descartadas no momento da construção do sistema proposto para a SUGEP.
- 2 Pesquisar Políticas do Núcleo de Tecnologia da Informação (NTI):** O NTI é o departamento da UFRPE responsável pelos serviços de TI da universidade. O sistema desenvolvido por esta pesquisa será implantado nos servidores do NTI. Logo, torna-se necessário descobrir as políticas do NTI para identificar possíveis restrições tecnológicas e regras que versam sobre a implantação do sistema em seu domínio.
- 3 Estudar Tecnologias para Desenvolvimento do Software:** Após a descoberta na etapa 2, aprofundar-se nas tecnologias que apoiarão o desenvolvimento do sistema.
- 4 Estudar Legislação Específica:** Estudar as leis que apoiam e regem a entrega do auxílio saúde, a fim de entender quais são as regras e restrições do benefício. Vale

lembrar que a legislação também será uma das fontes de requisitos para o sistema. Outras principais fontes de requisitos serão exploradas na etapa 6.

- 5 Criar Sistemática p/ Automação de Processos:** Definir com base na literatura uma sistemática para automatização de processos. Esta sistemática será a aplicada para solucionar os problemas da SUGEP, mas poderá ser utilizada integralmente ou com as devidas modificações em outro contexto para automatização de processos dentro da SUGEP ou no setor público de uma forma geral. A aplicação desta sistemática no contexto da SUGEP ocorrerá através das próximas etapas.
- 6 Modelar Processo de Negócio:** Entender a fundo o funcionamento do atual processo de gestão do auxílio saúde e desenhá-lo. Este esforço colaborará com a identificação de todos problemas e oportunidades de automatização deste processo. Isto permitirá que um novo processo, apoiado por um sistema de informação de propósito específico, seja proposto de forma a mitigar ou erradicar os problemas apresentados pelo processo manual.
- 7 Elicitar Requisitos:** Identificar, a partir dos problemas e oportunidades de automatização descobertos da fase “Modelar Processo de Negócio”, as funcionalidades do sistema proposto para automatização do processo. Esta etapa envolve a utilização de técnicas para elicitação de requisitos junto às fontes de requisitos mais relevantes para o projeto e redação de um documento com as especificações do sistema. A técnica de entrevista será aplicada com os principais interessados no projeto - beneficiários do sistema e funcionários da SUGEP - para entender as necessidades referentes ao atual processo de gestão do auxílio que, conseqüentemente, serão traduzidas em requisitos do sistema. Como citado na etapa “Estudar Legislação Específica”, a legislação também será uma importante fonte de requisitos para o sistema.

- 8 Analisar e Projetar:** Modelar, a partir dos requisitos levantados na etapa anterior, a arquitetura base do sistema. Esta modelagem, juntamente com os artefatos criados nas fases anteriores, guiará o desenvolvimento do sistema, o qual ocorrerá na etapa “Implementar o Sistema”.
- 9 Implementar o Sistema:** Configurar o ambiente de desenvolvimento e iniciar a escrita do código do sistema. Esta etapa será desenvolvida de forma intercalada com a etapa “Implantar o Sistema”.
- 10 Implantar o Sistema:** Disponibilizar, no domínio do NTI, o sistema para uso dos gestores do auxílio saúde. Esta etapa será feita ao longo de toda a pesquisa, de forma incremental. A cada duas semanas, uma nova funcionalidade será entregue ao cliente. A ideia é que nestas entregas parciais, possíveis falhas possam ser identificadas e ajustadas de acordo com o *feedback* dos usuários.
- 11 Validar o Processo Automatizado:** Fazer uma análise comparativa entre o processo de gestão do auxílio saúde antigo, feito de forma manual, e o processo automatizado pelo sistema. Para isso, a execução de projeto piloto e um questionário serão realizados com os principais interessados pelo sistema com o objetivo de confirmar a melhoria do processo após a implantação da solução.
- 12 Escrever o TCC:** A escrita do documento se dará durante todo o processo, desde o estudo inicial até a implantação e avaliação do desempenho do processo apoiado pelo sistema.
- 13 Entregar/Defender o TCC:** Entregar e defender o documento perante a banca, após a realização de todas as etapas.

1.7 CRONOGRAMA

Na Tabela 1 estão descritas as etapas, definidas na seção anterior, no tempo disponível para realização da pesquisa, considerando a escrita e a entrega do documento com o resultado da pesquisa.

Tabela 1 - Cronograma de execução das atividades.

Atividade	JUL	AGO	SET	OUT	NOV	DEZ	FEV
Revisão Bibliográfica	X	X	X	X	X		
Pesquisar sobre as Políticas do NTI	X	X					
Estudo das Tecnologias para Implantação	X	X	X	X	X		
Estudar a Legislação Específica	X	X					
Criar Sistemática p/ Automação de Processos	X	X					
Modelar Processo de Negócio	X	X					
Elicitar Requisitos	X	X					

Analisar e Projetar	X	X	X				
Implementar o Sistema		X	X	X	X		
Implantar o Sistema			X	X	X		
Validar o Sistema					X		
Escrever o TCC	X	X	X	X	X	X	X
Entregar/Defender o TCC							X

1.8 ESTRUTURA DO TRABALHO

Além deste capítulo inicial, outros 4 capítulos compõem este trabalho.

O Capítulo II apresenta os conceitos teóricos que foram utilizados como base para o desenvolvimento deste trabalho.

O Capítulo III apresenta os detalhes da construção da solução proposta para os problemas apresentados até então, bem como os subprodutos originados durante os esforços de desenvolvimento da solução.

O Capítulo IV avalia os resultados advindos da construção da solução. Tal capítulo tem como objetivo principal verificar se a solução proposta é válida.

Enfim, o Capítulo V apresenta as principais colaborações deste trabalho, além de sugestões de trabalhos futuros, limitações e as dificuldades encontradas.

CAPÍTULO II - FUNDAMENTAÇÃO TEÓRICA

Este capítulo tem a finalidade de apresentar os conceitos fundamentais ao desenvolvimento deste trabalho. As próximas seções abordam itens primordiais para o entendimento e concretização da solução proposta - o Sistema Auxílio Saúde (SAS) UFRPE.

A Seção 1 aborda de forma sucinta o tema “SISTEMAS DE APOIO À GESTÃO PÚBLICA”, visto que o sistema proposto neste trabalho dará suporte a um processo de gestão dentro do setor público. A Seção 2 aborda a análise de sistemas similares ao proposto neste trabalho.

A Seção 3 apresenta princípios de desenvolvimento de software incorporados a este projeto. Além disso, apresenta a proposta da simplificação da metodologia RUP com base nos princípios das metodologias ágeis.

A Seção 4 apresenta uma breve introdução ao tema “Sistemas Gerenciadores de Conteúdo”, além de apresentar as principais características do gerenciador de conteúdo utilizado (*Drupal*). Por fim, a Seção 5 mostra uma breve introdução à linguagem de programação PHP, ao qual é utilizado como base para o gerenciador de conteúdo adotado (*Drupal*).

2.1 SISTEMAS DE APOIO À GESTÃO PÚBLICA

No âmbito das empresas privadas percebeu-se que gerir e utilizar-se da melhor maneira possível as Tecnologias da Informação e Comunicação (TICs) como agente de mediação de comunicação e interação é um dos fatores chave de sucesso e tem-se buscado investir neste sentido, sob o olhar atento de gestores e investidores interessados em obter sucesso em seus empreendimentos (SANT’ANA, 2009). No entanto, na esfera pública, a motivação nem sempre é tão clara.

O setor público possui um déficit considerável se comparado aos avanços tecnológicos da iniciativa privada (DINIZ, 2005). Esta defasagem estende-se à automatização de diversos processos. Por conta desta deficiência, muitos processos extremamente custosos são executados

por diversos órgãos públicos, aumentando os custos operacionais, o tempo gasto pelo processo fim-afim, bem como a força de trabalho necessária a realização das tarefas.

A SUGEP - unidade da UFRPE alvo desta pesquisa tecnológica - é apenas um recorte do retrato nacional. Assim como a SUGEP, outros órgãos realizam muitas atividades manuais que levam horas para serem concluídas. Tais tarefas, ao serem realizadas com o apoio de um sistema, poderiam ser concluídas em poucos minutos. Esta possibilidade permitiria o uso mais eficiente da força de trabalho aos funcionários cujas horas restantes poderiam ser utilizadas para prestar novos serviços ao órgão. Como consequência, o público em geral seria beneficiado, uma vez que passaria a desfrutar de um serviço público mais prestativo e eficiente.

Neste contexto, é de suma importância que processos críticos dentro do setor público possam ser descobertos, analisados e automatizados por sistemas de *software*, de forma a aumentar a sua eficiência e, conseqüentemente, a capacidade de geração de valor dos órgãos que executam estes processos.

2.2 TRABALHOS RELACIONADOS

Assim como a Universidade Federal Rural de Pernambuco, diversas outras instituições federais, no Brasil, precisam gerir a entrega do Auxílio Saúde ao seu quadro de servidores. Algumas dessas instituições já possuem um sistema para dar suporte ao processo, os quais podem prover inspirações ao sistema proposto neste trabalho.

A exemplo, podemos mencionar a Universidade Federal do Rio Grande do Norte (UFRN) que possui um módulo de saúde, dentro do seu Sistema Integrado de Gestão de Recursos Humanos (UFRN, 2016), e a Universidade Federal do Pará (UFPA) que, dentro do Sistema de Auxílio Saúde, possui um Módulo de Envio de Boleto referente ao auxílio saúde (UFPA, 2016). Em ambos os exemplos, os módulos são responsáveis por apoiar toda ou parte do processo de gestão do auxílio saúde.

Porém, vale ressaltar que nós optamos pela realização do levantamento das funcionalidades do sistema, exclusivamente, através de um intenso trabalho de levantamento de

requisitos junto aos beneficiários do auxílio saúde e aos servidores da SUGEP - os principais afetados pela implantação do sistema. Esta restrição das fontes de requisitos que elimina a análise dos sistemas similares tem como objetivo trazer inteiramente o foco do trabalho para o exercício de práticas bem consolidadas da engenharia de software - voltadas para a elicitação de requisitos.

Alternativamente, poderíamos optar por adquirir um destes sistemas. No entanto, a UFRPE teria que adquirir um módulo completo de gestão de pessoas e não só o módulo de saúde, o que elevaria bastante o custo de aquisição. Além do mais, o tempo para implantação não seria curto, já que o processo envolveria a formalização de um convênio entre as duas universidades e contratação de pessoal para implantação do mesmo.

De qualquer forma, preferimos fazer um intenso trabalho de elicitação de requisitos junto aos usuários do futuro sistema. Desta forma, poderíamos desfrutar da vantagem de criar um sistema personalizado com as características da UFRPE, que atende as necessidades bem específicas dos servidores da SUGEP.

2.3 RUP E METODOLOGIAS ÁGEIS

O desenvolvimento do software proposto por esta pesquisa tecnológica foi parcialmente baseado no RUP com inspirações nos princípios das metodologias ágeis. Desta forma, esta seção descreve de forma sucinta os principais conceitos do RUP que foram utilizados ao longo do desenvolvimento deste projeto.

Em seguida, é feita uma breve apresentação dos princípios da modelagem ágil que foram adotados neste projeto. É importante salientar que a apresentação de como esses conceitos foram utilizados para realizar este trabalho será feita no Capítulo III, que trata justamente dos detalhes de concretização da solução proposta para o problema apresentado no Capítulo I.

2.3.1 RUP

À medida que a sociedade evolui, a complexidade dos problemas que precisam ser resolvidos através de sistemas de *software* aumenta. Consequentemente, os *softwares* crescem

em tamanho, complexidade e importância (KRUCHTEN, 2003). Este crescimento quando não acompanhado das melhores práticas de desenvolvimento de *software*, acaba gerando produtos de má qualidade que apresentam alguns desses problemas ou uma combinação deles: difícil manutenção; custo elevado - para o cliente ou para o desenvolvedor; prazo estourado; entre outros.

Para mitigar os riscos de um projeto de *software* fracassar, a indústria acaba criando padrões de desenvolvimento que consistem na compilação de várias práticas muito bem consolidadas. Estas práticas quando bem adotadas conferem um nível aceitável de qualidade ao software construído, impedindo que o produto apresente algum dos problemas mencionados (KRUCHTEN, 2003).

2.3.1.1 RUP - Conceitos Fundamentais

O *Rational Unified Process* (RUP) é processo de desenvolvimento de software (RATIONAL SOFTWARE, 2001). Seu objetivo é garantir a produção de software de qualidade, ou seja, que atenda às necessidades de seus usuários, dentro do prazo e orçamento estipulado. O padrão foi desenvolvido pela instituição Rational® Software que ainda hoje é responsável por sua divulgação e atualização, de acordo as melhores práticas que surgem no mundo do desenvolvimento de *software*.

Segundo o *Rational® Software* (RATIONAL SOFTWARE, 2001), o *Rational Unified Process* é um processo configurável, ou seja, o processo foi criado de forma a se encaixar em pequenas equipes de desenvolvimento, bem como em grandes organizações, podendo ser modificado para acomodar diferentes tipos de projeto. Esta característica torna o RUP bastante atrativo, já que é inerente o fato de que nenhum processo é inteiramente adequado a qualquer desenvolvimento de software.

Geralmente, ou o processo fornece ferramentas demais (supérfluas) ao desenvolvimento ou não dispõe de mecanismos suficientes. Sendo assim, esta versatilidade foi essencial à escolha desta metodologia para dar suporte ao desenvolvimento do software proposto neste trabalho, uma

vez que o RUP em sua totalidade é muito dispendioso e muitas de suas atividades/artefatos, embora muito bem fundamentados, não são necessários ao desenvolvimento deste projeto.

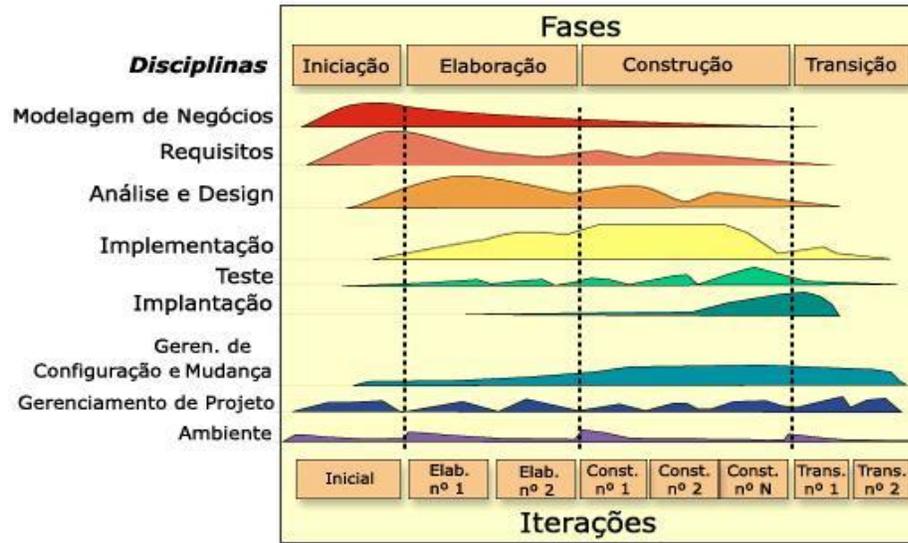
2.3.1.2 RUP - Fases x Disciplinas

Como pode ser visto na Figura 3, o RUP divide o processo de desenvolvimento em várias fases e atribui a cada fase um conjunto de disciplinas que podem ser entendidas como um conjunto de atividades a serem realizadas para o sucesso da respectiva fase (RATIONAL SOFTWARE, 2001). Cada uma destas atividades pode estar atribuída a diversas pessoas e gerar diferentes artefatos (documentos, diagramas, manuais, código do sistema).

Cada fase do RUP pode ser dividida em iterações. Uma iteração é um loop de desenvolvimento completo, que resulta em uma liberação de um artefato ou parte do mesmo (dentro do ambiente de desenvolvimento ou para o cliente). Ao longo do projeto, a iteração pode ser repetida quantas vezes forem necessárias. O software cresce a cada iteração até se tornar a versão final do sistema.

É importante saber, que a percepção desta distribuição Fases x Disciplinas é fundamental para o entendimento da organização do desenvolvimento do sistema proposto, pois o projeto e a construção do mesmo foi inteiramente baseada nesta metodologia, com pequenas modificações inspiradas nos princípios das metodologias ágeis descritas na Seção 3.2.1. Basicamente, a construção do SAS - UFRPE segue todas as fases aqui descritas e incorpora as principais disciplinas/atividades do RUP.

Figura 3 - Fases x disciplinas do RUP



Fonte: (WIKIPÉDIA, 2012)

2.3.1.3 RUP – Fases

Como pode ser observado na Figura 3, o *Rational Unified Process* divide o ciclo de desenvolvimento em quatro fases e cada uma dessas fases pode ocorrer de forma iterativa e incremental. Ou seja, podem ser revisitadas várias vezes ao longo do desenvolvimento do software (iterativa) e em cada visita parte das atividades é executada, gerando um incremento/melhoria nos artefatos já existentes ou no próprio produto (o código do software). As fases são:

- **Fase de iniciação:** Nesta fase ocorre o entendimento do processo de negócio do cliente e a identificação das entidades que irão interagir com o sistema. Além do mais, são descobertas as necessidades do cliente que são, então, mapeadas em possíveis requisitos/funcionalidades do sistema.

- **Fase de elaboração:** Em linhas gerais, o objetivo desta fase é validar e refinar os requisitos descobertos na fase de iniciação. Estes requisitos, após serem consolidados, são utilizados como base para a criação de uma arquitetura bem fundamentada. Dentro deste contexto, entende-se arquitetura como: as partes que compõe o sistema e como elas se comunicam. Tal arquitetura é usada como modelo para construção do sistema que ocorre na próxima fase - Construção.
Ainda nesta fase, é desenvolvido um plano do projeto responsável por guiar as próximas fases, ajudando a manter o custos, prazos e qualidade dentro do que foi planejado (KRUCHTEN, 2003). Por fim, mas não necessariamente nesta ordem, os riscos que podem levar o processo como todo ao fracasso são mitigados ainda nesta fase.
- **Fase de Construção:** Durante a fase de construção, todos os componentes restantes da arquitetura definida na fase anterior são desenvolvidos e integrados ao produto, e todos os recursos são cuidadosamente testados. A fase de construção é, de certa forma, um processo de fabricação onde a ênfase é colocada no gerenciamento de recursos e no controle de operações para otimizar custos, cronogramas e a qualidade do produto (RATIONAL SOFTWARE, 2001).
É importante salientar que a construção no RUP é feita de forma incremental: ao longo de todo o desenvolvimento, pedaços utilizáveis de software são liberados para a implantação no ambiente do usuário.
- **Fase de transição:** O objetivo da fase de transição é fazer a transição do produto de software para o ambiente de operação do usuário e treinar o usuário para a utilização do mesmo. Esta fase inicia quando existe um novo pedaço de software maduro o suficiente - que possa ser utilizado pelo usuário. E, assim como a fase de construção, esta fase é executada ao longo de todo o desenvolvimento do software, sempre que um novo pedaço de software está pronto para ser implantado.

Uma vez que o produto foi entregue ao usuário final, eventuais problemas podem surgir, o que pode exigir o desenvolvimento de novas funcionalidades ou correções das já existentes. Esta fase é essencial para que esses problemas sejam detectados o mais rápido possível e corrigidos, bem como perceber o nível de satisfação do usuário com o produto que está se formando, a fim de adaptá-lo e aproximá-lo daquilo que é esperado pelo usuário.

2.3.1.4 Disciplinas

Como mencionado na introdução desta seção, cada uma dessas fases é composta por um ou mais disciplinas/atividades, que são responsáveis por concretizar de fato as fases. São elas:

- Modelagem de Negócio
- Elicitação de Requisitos
- Análise e Projeto
- Implementação Testes Implantação

Segundo Todd Tomlinson e John K. Vandyk (2010), apesar da possibilidade do nome das atividades dar a entender que as fases ocorrem de forma sequencial, devemos ter em mente que o RUP é um processo iterativo. Isso significa que cada uma das fases anteriores e as atividades podem ser revisitadas diversas vezes ao longo do ciclo de vida do software.

2.3.1.4.1 Modelagem de Negócio

Nesta fase os processos de negócio do cliente não desenhados, visando sua melhor compreensão. Esta modelagem permite que os envolvidos no projeto possam comparar o funcionamento atual do processo (AS-IS) com a projeção do processo após a implantação do sistema (TO-BE). Neste momento, é possível identificar as possíveis melhorias e decidir se a relação “custo x benefício” é interessante tanto para a equipe de desenvolvimento quanto para o cliente darem continuidade a construção do software.

Como se não bastasse, esta compreensão colabora com a fase de elicitação de requisitos, já que ao fornecer uma visão completa de como a organização do cliente funciona, é possível identificar necessidades que não seriam descobertas sem o entendimento do processo fim a fim. Isto permite a identificação de aspectos que talvez não fossem identificados apenas com as técnicas de elicitação de requisitos. O desenho da modelagem de negócio geralmente é feito com o apoio de uma dessas duas linguagens:

- **UML** (ERIKSSON, 2000): O *Unified Modeling Language* (UML) é linguagem voltada para a modelagem visual de software orientado a objeto amplamente adotado no mundo do desenvolvimento. Pode ser utilizada para desenhar processos de negócio através do uso de diagramas de atividades.
- **BPMN** (WHITE, 2008): O *Business Process Model and Notation* (BPMN) é uma representação gráfica para especificar processos de negócios.

2.3.1.4.2 Elicitação de Requisitos

Segundo o *Rational Software* (2001), o objetivo desta atividade é descobrir e descrever o que o sistema deve fazer, e que, ao fim desta etapa, os desenvolvedores e o cliente concordem com essa descrição - a especificação do sistema. Para tanto, é preciso entender como funciona o negócio do cliente e quais são as suas necessidades.

Karl Wieggers e Joy Beatty (2013) descrevem as principais técnicas que podem ser empregadas com este propósito. A escolha da técnica é dada com base no tipo do projeto e nas características das partes envolvidas. De forma resumida, as principais técnicas são:

- **Entrevistas:** A técnica mais comum para a coleta de requisitos. Consiste em sentar frente a frente com as principais fontes de requisitos. Geralmente, os indivíduos escolhidos são aqueles que terão o maior contato com o sistema ou que serão mais afetados pela implantação do mesmo, não se limitando a eles.

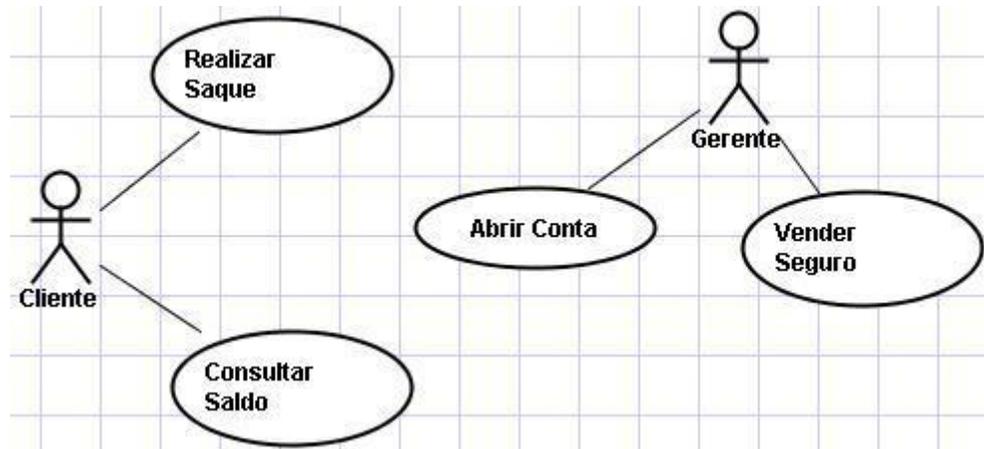
As perguntas da entrevista podem ser livres (não estruturada) ou seguir um roteiro. Ao adotar um roteiro, a entrevista ainda pode seguir dois formatos:

- I. Estruturada:** as perguntas se limitam às definidas no roteiro;
- II. Semi-estruturada:** são definidas algumas perguntas bases para guiar a entrevista, mas a ideia é que novas perguntas surjam ao longo da entrevista;
- **Análise da Legislação:** Diversas leis podem modificar o comportamento do sistema ou impor restrições que se não cumpridas inviabilizam o desenvolvimento ou a implantação do sistema. Portanto, a depender do projeto, a consulta à legislação relacionada ao domínio do problema pode ser uma das principais fontes de requisitos.
 - **Prototipagem:** A criação de um rascunho de aplicação para verificar se o entendimento da equipe de desenvolvimento é consonante ao do cliente e estimular a criação de novas ideias.

Com apoio de uma ou algumas dessas técnicas, é feita a descoberta das necessidades do cliente e, em seguida, um trabalho de refinamento destas necessidades a fim de mapeá-las para funcionalidades e restrições do sistema. O produto deste refinamento dá origem ao artefato Documento de Requisitos do Sistema - que pode ser também chamado de especificação do sistema.

Na próxima tarefa desta atividade, os atores são identificados, representando os usuários, e qualquer outro sistema que possa interagir com o sistema que está sendo desenvolvido. Então, um Diagrama de Casos de Uso UML (W3TECHS, 2016) pode ser criado para representar, de forma superficial, o comportamento do sistema, como os atores interagem com as partes do sistema e casos de uso (ou funcionalidades). A Figura 4 ilustra um Diagrama de Casos de Uso de um sistema fictício, onde os atores são: “Cliente” e “Gerente”; e as funcionalidades: “Realizar Saque”, “Consultar Saldo”, “Abrir Conta” e “Vender Seguro”.

Figura 4 - Ilustração de um diagrama de casos de uso.



Fonte : (MADEIRA, 2007)

Em seguida, cada Caso de Uso pode ser descrito com mais detalhes, de forma a criar-se um novo artefato: o Documento de Casos de Uso - que mostra os detalhes associados a cada Caso de Uso, bem como sua relevância para o cliente do software.

Este artefato, juntamente com os demais artefatos mencionados nesta seção, será utilizado como base para o desenvolvimento do sistema nas próximas fases, sobretudo na fase de “Análise e Projeto” e “Teste”. Inclusive, o próprio esforço de elicitação de requisitos pode ser validado após a geração destes artefatos, uma vez que estes (os artefatos) são estrategicamente compreensíveis ao cliente que, ao encontrar incoerências, pode acusá-las à equipe de desenvolvimento que fará as devidas adequações.

2.3.1.4.3 Análise e Projeto

De acordo com o *Rational Software* (2001), os esforços desta fase concentram-se em torno da definição da arquitetura do *software* que está sendo desenvolvido. Esta arquitetura basicamente consiste na definição dos componentes que compõem o software, bem como a comunicação entres tais componentes. Ao fim desta fase, estes componentes devem estar

detalhados o suficiente para que na fase de Implementação, o código-fonte da aplicação seja escrito, basicamente, com base nas especificações definidas na arquitetura.

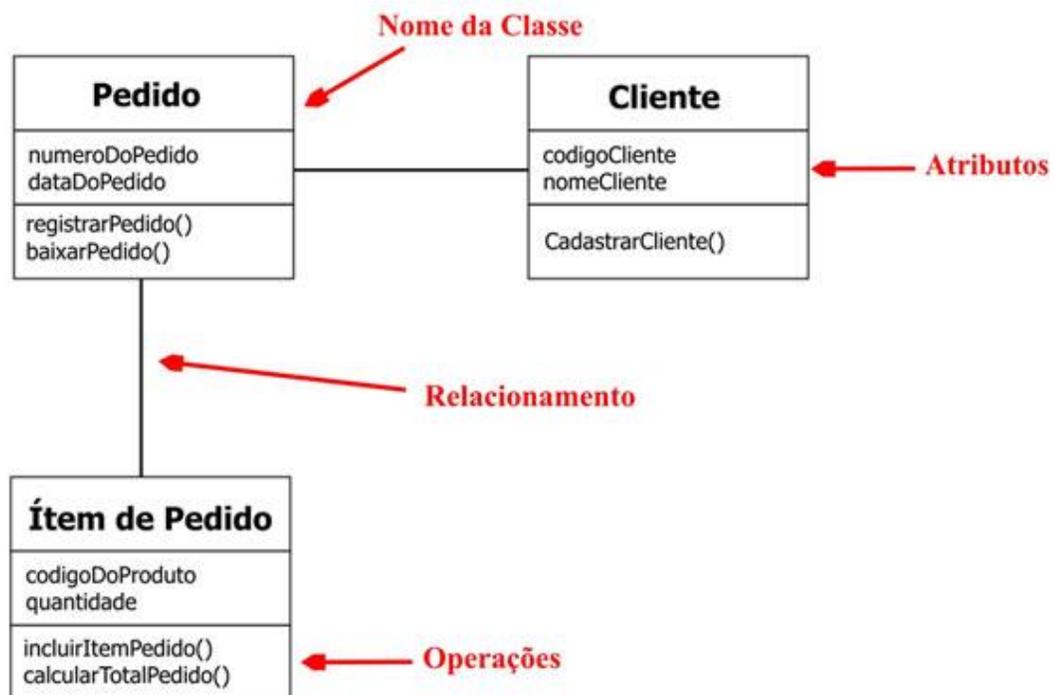
Estes componentes, na verdade são agrupamentos de entidades que colaboram para produzirem funcionalidades para o sistema. Aproximando esta definição de um mundo Orientado a Objeto (OO), as entidades são nada mais que classes e objetos, que nesta fase são definidas da forma mais fiel possível, de forma que, na fase de Implementação, os esforços se concentrem apenas na realização dessas classes e objetos.

Nesta fase, pode ser criado um diagrama de classes (MICROSOFT, 2015) que representa todas as entidades (classes) existentes no sistema e como estas estão relacionadas. Este diagrama é utilizado então pelo desenvolvedor da aplicação para concretizar (escrever o código) que define estas classes, dando origem ao sistema de fato. Além disso, este diagrama pode ser utilizado para definir o banco de dados que armazena de forma persistente os dados referentes a estas entidades.

Figura 5 ilustra um simples diagrama de classe. Tal diagrama é composto por:

- Classes: representam as entidades que fazem parte do sistema;
- Relacionamentos: indicam como as entidades estão associadas;
- Atributos: representam propriedades/qualidades das entidades;
- Operações: são operações que podem ser executadas dentro da aplicação sobre aquela entidade.

Figura 5- Ilustração de um diagrama de classes.



Fonte (ESPÍNDOLA, 2016)

2.3.1.4.4 Implementação

O objetivo da implementação é definir a organização do código-fonte do software, bem como sua escrita de fato, que se concretiza através da implementação dos componentes definidos

na fase de Análise e Projeto. Após a construção dos componentes ou parte dele, o código criado pela equipe de desenvolvimento é testado e, quando estável, integrado à aplicação dando sempre origem a um sistema incrementado e executável - que pode ser implantado e utilizado no ambiente do usuário.

Segundo o *Rational Software* (RATIONAL SOFTWARE, 2001), esta divisão do sistema em componentes confere uma série de vantagens, como: facilitação da manutenção do sistema e possibilidade de reutilização de componentes existentes. Ainda mais, embora todos componentes possam ser implementados nesta fase pela equipe de desenvolvimento, os componentes podem ser adquiridos de terceiros, cabendo a equipe apenas os esforços de integração ao sistema em construção.

2.3.1.4.5 Teste

Embora testes sejam executados maciçamente, na fase implementação, durante a criação do código-fonte da aplicação, eles não se limitam apenas a este momento (RATIONAL SOFTWARE, 2001). Muito pelo contrário, os testes também são importantes para:

- Verificar a integração adequada de todos os componentes do software.
- Verificar a todo instante se todos os requisitos foram implementados corretamente.
- Identificar e garantir que os defeitos gerados nas fases anteriores do desenvolvimento não sejam propagados às próximas fases.

O Rational Unified Process propõe uma abordagem iterativa para todas suas etapas, inclusive esta, o que significa que você testará todo o projeto. Isso permite que você encontre defeitos o mais cedo possível, reduzindo radicalmente o custo de consertá-los.

2.3.1.4.6 Implantação

O propósito desta fase é produzir e entregar novos incrementos de software aos usuários finais. Isto cobre uma série de tarefas como (RATIONAL SOFTWARE, 2001):

- Produção de novas versões do software: consiste no empacotamento dos novos recursos desenvolvidos com a versão mais estável do software, até então liberada para o usuário.
- Distribuição do software: consiste na entrega do incremento do software ao cliente.
- Instalação do software: disponibilização do software na estrutura física que comportará o mesmo - servidores web ou nas máquinas do cliente.
- Treinamento dos usuários: torna os usuários do sistema aptos à utilização das novas funcionalidades disponibilizadas.
- Planejamento e realização de testes beta: verifica se o incremento de software possui alguma falha ou o sistema regrediu ao incorporar tal incremento. O teste é executado no ambiente de trabalho do usuário.
- Aceitação formal do cliente: posicionamento do cliente indicando que o software cumpre com suas expectativas.

É importante reiterar que, assim como as outras atividades, a Implantação ocorre de forma incremental, sendo revisitada várias vezes à medida que novas funcionalidades vão sendo criadas para o sistema na fase de Implementação.

2. 3.1.5 RUP - Conclusão

Vale ressaltar que o RUP possui uma série de especificações além das descritas até então: papéis, outras atividades e artefatos mais complexos. No entanto, elas não foram consideradas relevantes para este trabalho. Sendo os princípios versados até então suficientes para o entendimento de como foi abordado o processo de desenvolvimento do sistema proposto neste trabalho.

2.3.2 Valores Das Metodologias Ágeis

De acordo com Ambler (2002), as metodologias ágeis surgiram como uma alternativa às abordagens tradicionais (como RUP, por exemplo) com o objetivo de aperfeiçoar o processo como se dá o desenvolvimento de um produto de software, eliminando práticas desnecessárias que muitas vezes são um desperdício de tempo e recursos preciosos para o projeto.

Valendo-se desse princípio, várias metodologias ágeis surgiram como XP, Crystal, SCRUM, DSDM (DYBÅ, 2008), entre outras. Todas baseadas nos 4 valores fundamentais do manifesto ágil (SLIGE e BRODERICK, 2008):

- “**software funcional** mais do que uma documentação completa”;
- “**a colaboração do cliente** mais do que a negociação de um contrato”;
- “**resposta às mudanças** mais do que seguir um plano”.
- “**as pessoas** acima de procedimentos e ferramentas”

Estes valores são suficientes para entendermos de onde surgiram as inspirações para se propor uma metodologia de desenvolvimento de software criada a partir da simplificação do RUP. Tal metodologia será utilizada ao longo de todo o desenvolvimento do sistema proposto neste trabalho e é apresentada na seção seguinte.

2.3.2.1 RUP Simplificado: RUP + Valores Ágeis

Embora os conceitos apresentados pelo RUP estejam muito consolidados, há uma série de práticas, artefatos, papéis, entre outros detalhes, que podem deixar a metodologia bastante laboriosa quando seguida à risca em sua totalidade. Felizmente, de acordo com o *Rational® Software* (RATIONAL SOFTWARE, 2001), o RUP é um processo de desenvolvimento configurável, ou seja, é possível modificá-lo de acordo com as necessidades do projeto. Diante disso, ao conhecer os valores das metodologias ágeis (BECK e KENT, 2001), podemos unir os dois mundos a fim de criar uma metodologia que ao mesmo tempo em que é “leve” (eliminando

desperdícios), é bem consolidada. Desta forma, o desenvolvimento do *software* deste trabalho será baseado no RUP, valendo-se de alguns valores Ágeis da seguinte forma:

- “**software funcional** mais do que uma documentação completa”: será desenvolvida apenas a documentação considerada necessária ao desenvolvimento do projeto e no nível de detalhe suficiente. Partes do software prontas para o uso serão entregue ao cliente de forma contínua, de forma incremental - em pequenas partes utilizáveis
- “**colaboração do cliente** mais do que a negociação de um contrato”: um canal de comunicação sempre ativo e rápido é estabelecido entre o desenvolvedor e o cliente. O objetivo desta abordagem é garantir que dúvidas sejam sanadas o mais rápido possível, eliminando todas as barreiras que possam interromper o processo de desenvolvimento; e até mesmo gerar inconsistências no entendimento que podem gerar um produto diferente do esperado pelo cliente.
- “**resposta às mudanças** mais do que seguir um plano”: todas as etapas do desenvolvimento são feitas em pequenas partes e ao final de cada pequena parte o produto e/ou artefatos gerados são imediatamente apresentados ao cliente a fim de validá-los. Esta abordagem permite que inconsistências possam ser encontradas o mais rápido possível e impedidas de se propagarem às demais fases do desenvolvimento.

Dado que não há tempo e prazo suficiente (o projeto tem apenas 5 meses de duração), não seria possível implementar todos os recursos de uma metodologia tradicional como o RUP. Por conta disso, a proposta dessa junção permite que o autor possa focar na entrega contínua de funcionalidades do sistema (software funcional) ao cliente, despreendendo-se de burocracias e documentação extensa. Esta abordagem colabora para a entrega do sistema dentro do prazo estabelecido e de um sistema que obedece às especificações criadas pelo autor deste trabalho em conjunto com as partes interessadas no sistema.

2.4 SISTEMAS GERENCIADORES DE CONTEÚDO (SGC) E *DRUPAL*

Para o entendimento do que se trata os sistemas gerenciadores de conteúdo, assim como o próprio Drupal, é necessário o prévio conhecimento da definição de conteúdo, inserida no contexto deste trabalho. Segundo Deane Barke (2016), conteúdo é a informação criada, editada e publicada por seres humanos para consumo por parte de outros seres humanos. Sendo assim, percebemos que conteúdo é um elemento bastante presente no mundo das aplicações de software, sobretudo nas disponibilizadas através da internet.

A exemplo, podemos citar aplicações bem populares, da atualidade, voltadas ao gerenciamento maciço de conteúdo, como: *Facebook*, *Twitter*, *YouTube*, *Google Maps* e *LinkedIn*, entre outras. Estas aplicações permitem que seus usuários criem/modifiquem conteúdos de forma que os outros usuários possam consumi-los, sendo o conteúdo apresentado nas mais variadas formas, como por exemplo: vídeos, músicas, artigos, páginas web, comentários, formulários, mapas, gráficos, currículos, entre outros.

O papel dos sistemas gerenciadores de conteúdo (SGC), em suma, está na automatização das tarefas necessárias ao gerenciamento eficiente deste tipo de conteúdo (BARKE, 2016). A ferramenta permite que múltiplos usuários criem/modifiquem e interajam com o conteúdo armazenado em algum repositório, seja ele um banco de dados e/ou um sistema de arquivos. Uma das maiores vantagens deste tipo de sistema é a capacidade de customização: é possível modificá-lo a fim de construir as mais diversas aplicações, não estando a aplicação proposta neste trabalho fora desta lista. Tecnicamente, o SGC tem a capacidade de se comportar como um framework voltado para desenvolvimento de software, sendo possível usá-lo como um conjunto de ferramentas para apoiar o desenvolvimento de aplicações tradicionais.

Os SGCs, assim como os frameworks, resolvem vários problemas bem comuns ao mundo das aplicações de software. Funcionalidades que geralmente estão presentes na maioria dos softwares vêm previamente embutidas nos SGCs e podem ser ajustadas para atender a real necessidade do projeto. Esta característica permite que o desenvolvedor da aplicação direcione seus esforços à construção das funcionalidades específicas ao negócio do cliente. Consequentemente, o cliente perceberá que a aplicação agrega mais valor ao seu negócio.

2.4.1 Tipos de SGC

Os gerenciadores de conteúdo podem ser divididos em 4 grandes grupos (BARKE, 2016), sendo esta classificação importante para entendermos em quais categorias melhor se encaixa a aplicação proposta neste trabalho:

- **Gerenciamento de Conteúdo Web (GCW):** A gestão de conteúdos destina-se principalmente à entrega em massa através de um *web* site.
- **Gerenciamento de Conteúdo Corporativo (GCC):** A gestão do conteúdo da empresa de uma forma geral, não necessariamente destinada à entrega ou consumo em massa (por exemplo, currículos dos funcionários, relatórios de incidentes, memorandos, etc.). Era, tradicionalmente, conhecido como "gerenciamento de documentos". O **GCC** dá um maior enfoque a colaboração (gerenciamento da concorrência pelo conteúdo), controle de acesso (definição de diferentes permissões para cada usuário) e gerenciamento de arquivos.
- **Gerenciamento de Ativos Digitais (GAD):** A gestão e manipulação tem um destaque maior de ativos digitais como imagens, áudio e vídeo.
- **Gerenciamento de Registros (GR):** Gerenciamento de informações transacionais e outros registros criados como subproduto de operações comerciais (por exemplo, registros de vendas, registros de acesso, contratos, etc.).

Sendo assim, é possível entender quais grupos melhor comportam o sistema SAS UFRPE e quais as características que serão herdadas desses grupos por esta aplicação:

- **Gerenciamento de Conteúdo Web:** O conteúdo do SAS será disponibilizado através de uma interface *web*. Basicamente, toda interação com o conteúdo disponível no sistema será através da Internet. Não podemos considerar que a entrega será em massa pois a aplicação será acessada apenas por algumas centenas de usuários, sendo a maior parte desse acesso apenas uma vez no mês.

- **Gerenciamento de Conteúdo Corporativo:** Este é o tipo de SGC que mais se adequa à realidade do SAS - UFRPE e, portanto, recebe maior enfoque. Isso deve-se ao fato do SAS tratar, prioritariamente, da troca de documentos (que comprovam o gasto com o plano de saúde) entre os beneficiários do Auxílio Saúde e os gestores do benefício.

Outra característica muito importante deste tipo de SGC, que precisa ser incorporada ao SAS, é o controle de concorrência e acesso, uma vez que o sistema será utilizado por centenas de usuários e nem todos usuários têm a permissão para visualização de todo conteúdo. Por exemplo, o usuário beneficiário do Auxílio Saúde não deve visualizar os comprovantes de pagamento de outros usuários.

As características apresentadas até então são fundamentais para o desenvolvimento de diversas aplicações e na maioria das aplicações convencionais são suficientes. Para reafirmar a capacidade dos SGCs na construção deste tipo de aplicação, Deane Barke (2016) apresenta exemplos de várias aplicações que já foram desenvolvidas com o apoio de algum Sistema Gerenciador de Conteúdo:

- Uma pequena plataforma *online* para divulgação de um consultório odontológico;
- Um *web site* para um jornal multinacional que publica centenas de artigos por dia;
- Um blog para um simples autor que escreve sobre tecnologia;
- Um portal com os materiais e programas referentes aos cursos de uma universidade;
- Uma comunidade *online* ou rede social;
- Entre outras;

O sistema proposto neste trabalho, o SAS - UFRPE, não foge a esta realidade, pois, assim como as aplicações acima, ele é perfeitamente factível com o apoio de um sistema gerenciador de conteúdo. Para este trabalho, a escolha do SGC recaiu sobre o *Drupal*, o qual será abordado com mais detalhes na próxima seção.

3.4.2 DRUPAL

O *Drupal* é um sistema gerenciador de conteúdo voltado para a internet e é, portanto, utilizado para construção de *websites* e aplicações *web*. O SGC está entre os mais populares da atualidade. A empresa ¹*W3Techs* divulgou uma pesquisa sobre a fatia de mercado dos principais SGC (*W3TECHS*). Nesta pesquisa, o *Drupal* figurou em terceiro lugar com 4,8%, perdendo para o *Joomla* - 6,7% - e *WordPress* - 58,7%. O restante do mercado ficou distribuído entre as outras centenas de SGCs.

Apesar da discrepância em relação ao gerenciador de conteúdo *WordPress*, o *Drupal* ainda é a preferência de muitas instituições públicas, a exemplo de instituições públicas de grande porte, como: Oregon State University e University of Minnesota; e o governo da Austrália. Até mesmo empresas privadas renomadas adotaram o gerenciador de conteúdo como: a Warner e a Honda Brasil. O motivo pelo qual estas instituições optam pela ferramenta pode ser justificado pelas vantagens que são apresentadas mais abaixo.

A primeira delas, segundo Todd Tomlinson e John K. VanDyk (2010), é a capacidade do gerenciador de conteúdo permitir a criação - rápida e de forma simplificada - de sites/aplicações simples e complexas. Um simples blog, site corporativo ou uma rede social são possíveis de ser implementados contanto que se dedique tempo e esforço suficiente. Além disso, como se trata de uma plataforma de código aberto, está em constante aperfeiçoamento, pois todos os dias pessoas de todo lugar do mundo contribuem com melhorias de segurança e performance da ferramenta.

Outra vantagem muito atrativa para os desenvolvedores de aplicações apoiadas pelo *Drupal* é a colaboração da comunidade (TOMLINSON e VANDYK, 2010)- os membros são bastante ativos, disponibilizando bastante tutoriais (escritos e em forma vídeos). Esta cooperação se estende a fóruns dedicados a sanar dúvidas e resolver erros encontrados durante o desenvolvimento. Isto contribui para um desenvolvimento fluido da aplicação, uma vez que a causa/solução de muitas dificuldades, encontradas durante a construção do sistema, serão identificadas ao consultar os fóruns especializados.

¹*W3Techs*: Instituto responsável pela coleta de informação sobre o uso de vários tipos de tecnologias *web*.

Trazendo o foco para o contexto deste trabalho, vale mencionar que a Universidade Federal Rural de Pernambuco (UFRPE), ao reconhecer os casos de sucessos mencionados mais acima, bem como as vantagens do *Drupal*, optou por criar uma política de implantação de sistemas que determina que todos sistemas - implantados por terceiros sob o domínio da universidade - devem utilizar o SGC *Drupal* para apoiar o seu desenvolvimento.

Alguns exemplos de sistemas/sites desenvolvidos com o apoio do SGC e implantados sob o domínio da UFRPE são: o site principal da instituição; o portal do JEPEX, o qual permite a divulgação de notícias da jornada científica, submissão e acompanhamento dos trabalhos referentes ao evento; portal de concursos da UFRPE, o qual permite o acompanhamento de concurso para o preenchimento do quadro de docentes da referida universidade, bem como as inscrições do certame.

Desta forma, torna-se indispensável o aprofundamento na ferramenta para que seja possível a concretização do SAS - UFRPE. Portanto, a seção abaixo apresenta os conceitos do *Drupal* mais relevantes, cujo entendimento permitirá a construção do sistema proposto.

2.4.2.1 Arquitetura do *Drupal*

O *Drupal* é um SGC modular. As funcionalidades são adicionadas ao habilitar módulos.

Estes módulos são nada mais que arquivos de código-fonte escritos na linguagem de programação PHP e são classificados em:

- **Módulos do Núcleo:**

O *Drupal* é distribuído com estes módulos por padrão. Eles fornecem as funcionalidades básicas para o funcionamento do sistema como, por exemplo: gerenciamento de usuários, controle de sessão, mecanismos para criação/edição/publicação de conteúdo.

- **Módulos de Contribuidores:**

Estes módulos são construídos pela comunidade e disponibilizados para download. Eles permitem a construção de novas funcionalidades com menos

esforço. Geralmente, quando algum conceito novo é lançado na *web* a comunidade se mobiliza para habilitá-lo no *Drupal* em alguns dias. Para tanto, criam um módulo que estende as funcionalidades do SGC.

- **Módulos Customizados:**

Esses módulos são escritos pelo desenvolvedor da aplicação. Os módulos do núcleo do *Drupal* não podem ser modificados. Portanto, quando o desenvolvedor percebe que algum módulo do núcleo ou de contribuidores não possui o comportamento desejado, é preciso sobrescrever o comportamento de todo o módulo ou de uma parte dele. Isso é feito através da escrita de módulos customizados que são invocados assim que os eventos a serem sobrescritos são disparados. Os módulos customizados são escritos na linguagem PHP.

Ao iniciar a construção de uma aplicação web, o desenvolvedor na verdade inicia uma espécie de um quebra-cabeça onde as peças são os diferentes módulos do *Drupal*. Ciente das funcionalidades que são necessárias à concretização da aplicação, o trabalho inicial do desenvolvimento gira em torno da pesquisa por módulos que realizam exatamente ou parcialmente aquilo que a funcionalidade exige.

Neste último caso, o módulo de precisa ser modificado para se encaixar perfeitamente à realidade da aplicação. Para isso, o desenvolvedor precisa escrever seu próprio módulo para sobrescrever todo ou parte do comportamento referente ao módulo. Após todos os módulos estarem configurados, habilitados e corretamente integrados, a aplicação construída com o apoio do *Drupal* está pronta para operar.

2.5 LINGUAGEM DE PROGRAMAÇÃO PHP

De acordo com Rasmus Lerdorf e Kevin Tatroe (2013), PHP é uma linguagem de programação *server-side* (executada num servidor web) desenvolvida para a geração de páginas

web dinâmicas, ou seja, que têm o seu conteúdo modificado à medida que novos dados são inseridos no sistema.

A linguagem foi criada em 1994 e desde então passou por várias atualizações que elevou a ferramenta a um estado de confiança e maturidade considerável. Para se ter uma noção da dimensão da tecnologia, a organização *W3Techs*, que coleta dados dos sites mais utilizados no mundo, durante o desenvolvimento desta pesquisa, afirma que a linguagem PHP era utilizada por 82.2% dos sites varridos pela ferramenta de monitoramento desenvolvida pela organização como linguagem *server-side*, sendo a versão 5 utilizada em 97,2% dos casos (W3TECHS, 2016).

Este fato deve-se a sua versatilidade. Pois mesmo sendo uma linguagem simples, é bastante robusta, uma vez que suporta a criação de aplicações tanto de médio quanto grande porte. Tal robustez pode ser comprovada ao analisar o desempenho de grandes serviços como: *Facebook*, *Wikipedia* e *Twitter* que utilizam a tecnologia e conseguem atender a uma enorme demanda.

Do ponto de vista do desenvolvedor de aplicações *web*, a tecnologia possui uma série de vantagens frente a outras linguagens do gênero. A primeira delas é a facilidade de aprendizado: parte desta facilidade é devida a similaridade com outras linguagens de programação muito bem estabelecidas como C e Java; bem como uma vasta documentação disponível, através de livros e *web sites*. Isto acaba contribuindo para que muitos iniciantes no desenvolvimento *web* adotem PHP como sua primeira linguagem *server-side*.

Além destas facilidades, a linguagem possui código aberto e uma comunidade ativa, o que permite que milhares de colaboradores contribuam tanto em fóruns dedicados a sanar dúvidas dos desenvolvedores quanto em comunidades responsáveis pelo aprimoramento da própria ferramenta. Essas duas realidades colaboram, respectivamente, para que os desenvolvedores *web* consigam evoluir na construção dos seus projetos quando dúvidas surgirem e permite que a tecnologia amadureça sempre através de melhorias de segurança e desempenho.

Estas, dentre outras vantagens são responsáveis por tamanha popularidade da linguagem PHP. Como mencionado mais detalhadamente nas seções anteriores, o gerenciador de conteúdo *Drupal* utiliza a tecnologia como base para construção de todos os seus módulos: do núcleo, de contribuidores e customizados. Vale ressaltar a importância desta tecnologia para este trabalho, já que todo o desenvolvimento do SAS foi realizado com o apoio desta linguagem. Seja nos

“bastidores” (utilizando módulos do núcleo e de contribuidores), seja explicitamente com o desenvolvimento próprio de módulos customizados.

CAPÍTULO III – DESENVOLVIMENTO

Este capítulo apresenta os detalhes da definição de uma sistemática para automatização de processos dentro do setor público. Em seguida, é apresentada a aplicação desta sistemática para o desenvolvimento do novo processo de gestão do auxílio saúde, automatizado por um sistema de informação.

Ainda é importante ressaltar que o desenvolvimento do novo processo de gestão do auxílio saúde envolve a construção de sua peça principal: o sistema de informação responsável pela automatização. Para o sucesso do processo, é importante que o desenvolvimento deste sistema obedeça às boas práticas preconizadas pela Engenharia de Software. Portanto, fundamentamos o seu desenvolvimento na metodologia de desenvolvimento de software RUP, simplificada com base nos valores ágeis.

Sendo assim, as demais seções também detalham a execução de todas as atividades realizadas a fim de se concretizar o sistema supracitado, desde a descoberta do problema, passando pela modelagem de negócio (definição do processo), elicitação de requisitos, construção da arquitetura do sistema, implementação e implantação. Vale a pena ressaltar, que ao sistema foi dado o nome de Sistema Auxílio Saúde (SAS UFRPE).

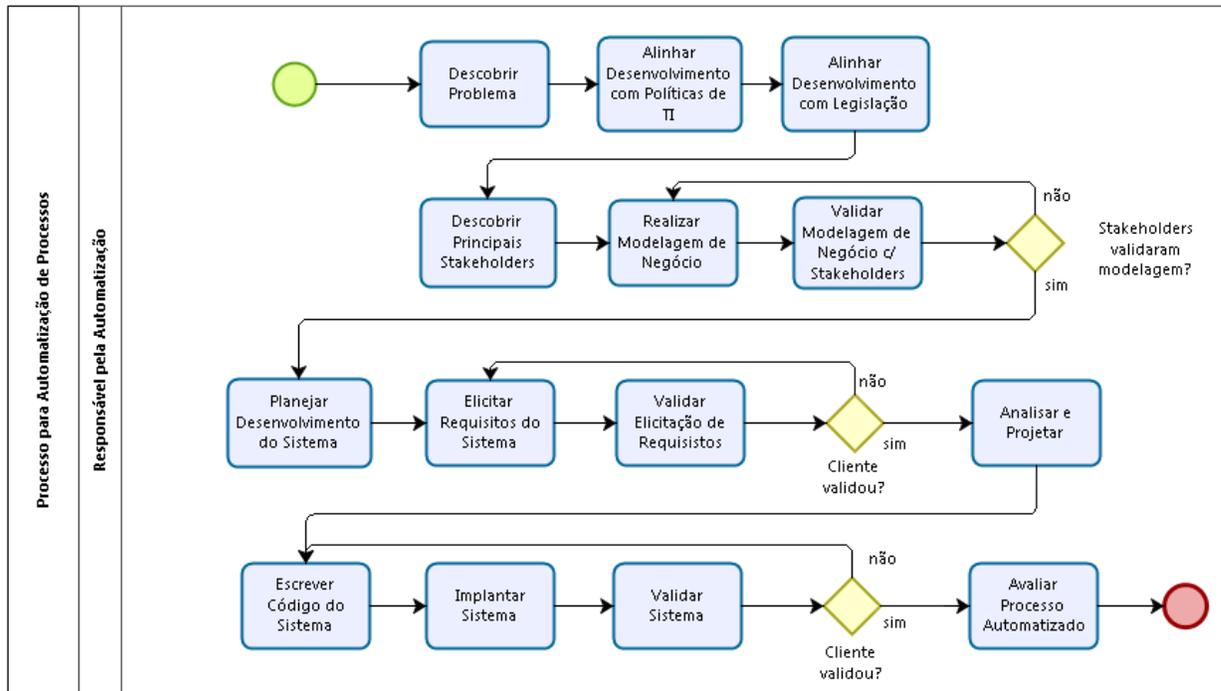
3.1 SISTEMÁTICA PARA AUTOMATIZAÇÃO DE PROCESSOS

Com o intuito de generalizar a automatização de processos através de *software* orientados a objetos executada neste trabalho, definimos uma sistemática que pode ser replicada em outros contextos para automatização de processos dentro da própria SUGEP ou no setor público, de uma forma geral, independentemente das tecnologias requeridas para o processo.

Esta sistemática consiste num conjunto de atividades que devem ser executadas a fim de permitir o entendimento do problema a ser solucionado, bem com a especificação, construção e validação do processo automatizado. A ideia é que ao fim da execução destas atividades tenha-se um processo automatizado por um sistema de informação. Tais atividades e fases foram

inspiradas na metodologia de desenvolvimento de software RUP. A Figura 6 ilustra as atividades desta sistemática e como elas se relacionam. As seções seguintes detalham cada atividade para que se tenha um entendimento melhor destas e seja possível pô-las em prática.

Figura 6 Sistemática para automatização de processos.



Fonte: Elaborada pelo autor.

Etapas da sistemática para automatização juntamente com sua descrição:

- **Descobrir problema:** Inicialmente, é descoberto um processo crítico, cuja necessidade de automatização por software é importante e aumentaria significativamente sua eficiência.
- **Alinhar Desenvolvimento com Políticas de TI:** Após a descoberta do problema e decisão em automatizá-lo, é necessário iniciar o estudo as políticas de TI da instituição na qual a solução automatizada será implantada, a fim de verificar possíveis restrições tecnológicas, as quais o *software* desenvolvido precisará se adequar.

- **Alinhar Desenvolvimento com Legislação:** Antes de iniciar a elaboração do novo processo é importante identificar e estudar a legislação específica em torno do processo, a fim de identificar possíveis restrições impostas pela lei, as quais o novo processo também deve se adequar.
- **Descobrir principais *stakeholders*:** Em seguida, é importante descobrir quais são as partes interessadas/afetadas pela automatização: os principais e eventuais usuários do software e organizações que participam de algum forma do processo. Estes *stakeholders* possuem o domínio do negócio e, portanto, serão os principais contribuidores com a modelagem de negócio e elicitação de requisitos que seguirão. Neste momento, deve-se estabelecer contato com os diversos envolvidos, com o propósito de marcar reuniões presenciais ou remotas para que seja possível entender o problema em profundidade.
- **Realizar Modelagem de Negócio:** Junto aos *stakeholders*, deve-se desenhar processo como é atualmente. Neste momento, pode ser realizadas entrevistas e questionários para que seja possível entender todos problemas e identificar possibilidades de automatização. A partir deste entendimento, deve-se propor um novo processo, automatizado por um sistema de software. Todo processo pode ser automatizada ou parte dele.

É importante ressaltar que esta atividade só deve se encerrar após a validação desta modelagem com os *stakeholders*: é preciso que os artefatos gerados nesta fase estejam em total consonância com a realidade, de forma que o sistema construído na próxima fase atenda a todas necessidades do problema abordado.

- **Planejar o Desenvolvimento:** Uma vez que a modelagem de negócio foi validada, deve-se iniciar o planejamento do desenvolvimento do sistema responsável por apoiar o processo. Nesta fase, é preciso decidir e iniciar o estudo das tecnologias utilizadas para o desenvolvimento do sistema: linguagens de programação, *frameworks*, sistemas gerenciadores de conteúdo, servidores de aplicação, entre outras.

Paralelo a este estudo deve-se iniciar a priorização de quais partes do processo são mais importantes e serão o foco principal no tocante à automatização. Esta priorização permitirá que durante as próximas atividades os problemas mais críticos - que mais prejudicam as partes envolvidas - sejam resolvidos o mais rápido possível

- **Elicitar Requisitos:** Esta atividade consiste no levantamento de todas as funcionalidades que o sistema deve conter para automatização do processo. Os esforços de modelagem de negócio são cruciais para que a partir das oportunidades de automatização descobertas seja possível vislumbrar tais funcionalidades. Os *stakeholders* identificados devem ser consultados para que o sistema atenda completamente às suas necessidades. A apresentação de um protótipo do sistema é importante para que os futuros usuários do mesmo possam identificar mais facilmente se aquilo que está prestes a ser construído é de fato o que é desejado. Só deve-se prosseguir para a próxima fase quando as funcionalidades que se pretende construir foram validadas pelos *stakeholders*.
- **Analisar e Projetar:** Nesta fase, a arquitetura do sistema deve ser construída. Aqui, propomos uma arquitetura básica, onde os componentes são basicamente as classes e o relacionamento entre elas. Estas classes são fundamentais para geração das entidades e dos relacionamento entre elas, tanto no banco de dados quanto na aplicação, e deve ser representadas através de uma diagrama de classes, para facilitar a compreensão da arquitetura desenvolvida por terceiros.
- **Escrever Código do Sistema:** O software que apoiará o novo processo deve ser concretizado. Para isso, é necessária a preparação do ambiente onde o software será implantado e a escrita do código através das tecnologias escolhidas.
- **Implantar Sistema:** O sistema desenvolvido na fase anterior deve ser então disponibilizado para que seja utilizado pelo usuário final. Além disto, é necessário treinar o usuário na utilização da sua nova ferramenta de trabalho. Este momento é crucial para que o usuário verifique se aquilo que foi entregue era de fato aquilo que atende às suas reais necessidades. Caso não, o software precisa ser reescrito para se adequar.
- **Avaliar Processo Automatizado:** Após a implantação de todo sistema, ou parte dele, é preciso verificar se o novo processo de fato resolve os problemas apresentados pelo antigo processo ou se pelo menos os mitiga. Para isso, o software pode ser colocado numa situação real de uso para que seja feita esta confirmação.

3.2 APLICAÇÃO DA SISTEMÁTICA À REALIDADE DA SUGEP

3.2.1 Concepção

3.2.1.1 Descoberta do problema da SUGEPE

Em conversas com beneficiários do auxílio saúde foi relatada a dificuldade em realizar a entrega do comprovante mensal de forma física. Eles alegaram dois problemas: era preciso ir pessoalmente até a unidade da SUGEP, interrompendo suas atividades; e nenhum *feedback* era fornecido quando havia algum problema com a documentação. Desta forma, sugeriu-se por parte de um dos beneficiários que um sistema poderia automatizar o processo a fim de solucionar estes problemas. Assim, solicitamos o e-mail de um dos servidores da SUGEP responsáveis pela gestão do auxílio saúde, para estabelecermos um canal de comunicação e entendermos melhor o problema.

Ao entramos em contato, um dos servidores relatou de forma bastante superficial os problemas que existiam no processo de gestão do auxílio saúde, mostrando que as dificuldades no processo de gestão do auxílio iam bem mais além do que as dificuldades apontadas pelos beneficiários. A gestora ressaltou, sobretudo, quão dispendiosa eram as principais atividades envolvidas em tal processo. A mesma relatou que o processo de gestão era extremamente mecânico, propenso a erros, além de descrevê-lo como estressante. Tal processo alocava três servidores e requeria várias horas para ser concluído, quando o mesmo poderia ser drasticamente simplificado se apoiado por um sistema operacional de propósito específico.

Foi então que acordamos com a servidora em nos reunir para estudarmos melhor todas as os problemas apresentados pelo processo antigo com o propósito de propor e implementar um novo processo de gestão do auxílio saúde, automatizado por um sistema computacional. Então, a servidora se mostrou muito interessada na proposta e concordou com a automatização.

Após a decisão de automatizar o processo de gestão do auxílio saúde, um canal de comunicação foi estabelecido entre nós e os servidores da SUGEP, através de um dos servidores. A fim de se criar uma comunicação rápida e efetiva, foram trocados os mais diversos contatos. Este estabelecimento foi baseado no valor ágil colaboração do cliente que preconiza a

aproximação do cliente do software (neste caso, os funcionários responsáveis pela gestão do Auxílio Saúde) com a equipe de desenvolvimento.

A ideia por trás deste princípio é que o entendimento do que está sendo construído seja o mais claro possível, possibilitando que maus entendimentos ou dúvidas a respeito do sistema sejam sanados o mais rápido possível, nas fases mais iniciais do desenvolvimento. A não adoção desta prática poderia fazer com que falhas não detectadas precocemente fossem descobertas tardiamente, demandando bastante carga em retrabalho, o que seria inadmissível dado o curto prazo para desenvolvimento do trabalho.

Além disto, esse canal de comunicação foi fundamental para marcação de reuniões para as próximas etapas de desenvolvimento, as quais ocorreram presencialmente e remotamente. Ao todo, foram quatro reuniões presenciais e uma remota com tela compartilhada. As reuniões aconteceram em diversas etapas do desenvolvimento e serão detalhadas ao longo das seções seguintes.

Além destas reuniões com o cliente do sistema, realizamos reunião com o Núcleo de Tecnologia da Informação (NTI) da UFRPE para acordar questões técnicas sobre o desenvolvimento do sistema, já que o mesmo fica hospedado em servidores deste órgão. Nesta reunião foram repassadas as seguintes restrições:

- Utilizar o gerenciador de conteúdo Drupal, na sua versão 7.x;
- Utilizar o ambiente de desenvolvimento disponível no domínio público "<http://www.sugep.ufrpe.br/auxiliosaude/>";
- Solicitações de mudança no ambiente deveriam ser solicitadas por e-mail a equipe do NTI.

3.2.1.2 Modelagem de negócio

Com o objetivo de iniciar as atividades de modelagem de negócio, realizamos uma entrevista semiestruturada com a equipe da SUGEP responsável pelo auxílio saúde. O objetivo do encontro foi entender como funcionava fim a fim o processo de gestão do Auxílio Saúde para que fosse possível identificar as possibilidades de automatização do mesmo.

Este encontro nos proporcionou um entendimento do processo fim a fim num nível de detalhe considerável. Após entrevista, em um outro momento, novas dúvidas foram surgindo. Para sanar essas dúvidas, ao longo de todo o desenvolvimento, nós utilizamos os canais de comunicação estabelecidos.

A partir deste encontro, foi possível entender e modelar o processo de gestão do auxílio saúde. O objetivo desta modelagem foi desenhar o funcionamento do processo de gestão do auxílio saúde do começo ao fim. A **Figura 2** ilustra este processo e foi apresentada com detalhes no Capítulo I.

O entendimento deste processo possibilitou a identificação de vários problemas na abordagem atual, bem como a possibilidade de automatização e ainda melhoria de algumas atividades do processo - a atividade poderia passar a ser executada de forma diferente/mais eficiente a partir da implantação do sistema. Este entendimento nos permitiu a elaboração de um novo processo de gestão do auxílio saúde. Tal processo e suas vantagens são apresentadas detalhadamente na Seção 1.3.1. A definição deste novo processo foi insumo direto para as atividades de elicitação de requisitos e modelagem do sistema que ocorreram nas fases seguintes.

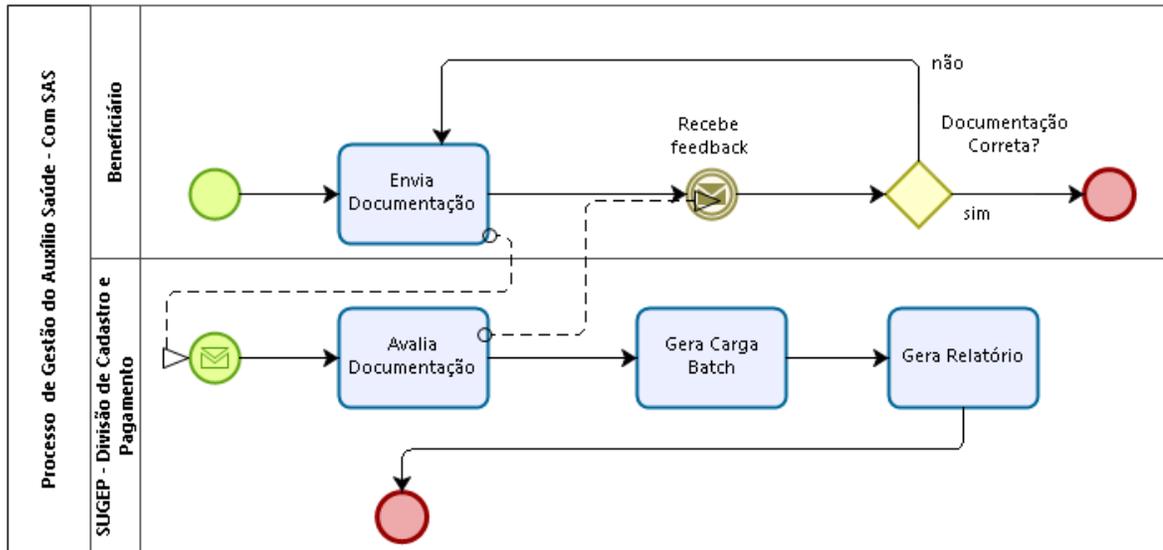
3.2.1.2.1 Processo Antes x Processo Depois

Com o objetivo de destacar as vantagens advindas da implantação de um sistema para apoiar a gestão do auxílio saúde, nós apresentamos a proposta de um novo processo - executado com o apoio do SAS – ao mesmo tempo que são destacadas as vantagens frente ao processo manual - sem o auxílio do sistema.

A implantação do SAS visa promover um processo de gestão do auxílio saúde mais humanizado, ao promover a automação das atividades mais laboriosas do processo. De forma prática, o SAS permite que as tarefas mais “pesadas” sejam realizadas pelo computador e não pelo ser humano.

O diagrama BPMN da Figura 6 ilustra o processo de gestão do auxílio saúde após a implantação do sistema. Mais abaixo, segue a explicação do diagrama e a apresentação das vantagens advindas da nova abordagem para gestão do benefício.

Figura 7 Ilustração do processo de gestão do auxílio saúde automatizado pelo SAS.



Fonte: Elaborada pelo autor

Atividades do processo de gestão do auxílio saúde automatizado pelo SAS:

- 1. Envia Documentação:** O beneficiário acessa o SAS e submete a documentação mensal necessária ao recebimento do benefício.
- 2. Avalia Documentação:** O gestor do auxílio, acessando o sistema, verifica a documentação e dá um parecer, informando se a documentação apresenta, ou não, problemas. Caso presente, o gestor informa, através do sistema, qual o problema encontrado.
- 3. Recebe *Feedback*:** O beneficiário recebe o parecer realizado pelo gestor do auxílio saúde informando se a documentação foi aceita ou apresenta algum problema. Caso

haja algum problema, o beneficiário pode reenviar a documentação, obedecendo aos prazos legais de reenvio, a tempo de receber o benefício ainda no mês seguinte.

4. **Gera Carga Batch:** Ao final do mês, o gestor do auxílio, através do sistema, gera um arquivo no formato de *Carga Batch* (vide Apêndice I - Documento de Requisitos) referente ao mês corrente. Tal arquivo é transferido pelo gestor do auxílio para o Sistema de folha de pagamento do Governo Federal (SIAPE). Sendo este sistema responsável pelo acréscimo na folha de pagamento do valor devido ao servidor por conta do benefício.
5. **Gera Relatório:** Após gerar a Carga Batch, ativando todos beneficiários, o gestor do auxílio saúde pode acessar o sistema e gerar uma planilha contendo um relatório de todos beneficiários que enviaram a documentação e foram ativados naquele mês.

Esta nova abordagem confere, tanto aos gestores quanto aos beneficiários do auxílio saúde, uma série de vantagens e soluções frente aos problemas apresentados pelo processo manual de gestão do auxílio saúde. Estas vantagens são apresentadas abaixo e estão agrupadas de acordo com as etapas do novo processo.

- **Envia Documentação:**

Percebe-se que o beneficiário não precisa mais interromper suas atividades para deslocar-se pessoalmente até a SUGEP, podendo direcionar seus esforços para outras finalidades. No caso dos beneficiários que são servidores ativos, a UFRPE será favorecida diretamente já que os servidores em questão pertencem à universidade.

Os dados referentes aos comprovantes de pagamento do plano de saúde serão armazenados digitalmente, dispensando o armazenamento de documentos físicos nas instalações do órgão e impedindo a perda em casos de situações

catastróficas, já que os servidores de armazenamento cuidarão da manutenção de *backups* dos documentos digitais.

- **Avalia Documentação:**

O sistema permite que os comprovantes possam ser facilmente recuperados e avaliados, quando necessário, através dos mecanismos de busca e filtros que estão disponíveis no sistema – por exemplo, busca por: todos comprovantes referentes ao mês atual que ainda não foram avaliados; comprovante de servidor que possui determinado CPF, entre outras. Esta abordagem diminui as chances de documentos ficarem de fora da avaliação, o que poderia impedir o beneficiário de receber o benefício no mês seguinte (comum na abordagem antiga - sem o sistema).

Além disso, o sistema pode fazer verificações prévias, no momento do envio da documentação para que o gestor não seja sobrecarregado. Por exemplo: o sistema pode verificar se o comprovante de pagamento enviado para determinado mês, já foi enviado e, portanto, barrar o envio e avisar ao beneficiário.

A avaliação permite que o gestor dê um *feedback* ao beneficiário. Este recebe um e-mail informando se a documentação contém algum erro ou está dentro do conformes. Vale ressaltar que o gestor não é sobrecarregado, uma vez que o sistema dispõe de justificativas pré-definidas, não sendo necessário ao gestor a redação de um texto para cada avaliação.

- **Gera Carga *Batch* e Gera Relatório:**

Após a avaliação de toda documentação, um intenso trabalho manual, que era realizado com a abordagem manual dá lugar a apenas alguns cliques dentro do sistema. O gestor do auxílio passa a navegar até a seção de carga *Batch* e Relatórios dentro do sistema. Então, o sistema recupera todos os dados referentes aos beneficiários e suas submissões de documentação. Com estes dados o sistema gera um arquivo no formato de *Carga Batch* ou o Relatório Mensal. Então, o gestor do

auxílio simplesmente faz a transferência do arquivo referente a carga Batch para o Sistema de folha de pagamento do Governo Federal (SIAPE) e todos servidores que tiveram a documentação aprovada são ativados de uma só vez. Desta forma, os beneficiários recebem o valor referente ao benefício na próxima folha de pagamento.

Na abordagem antiga estes poucos cliques no sistema e importação de apenas um arquivo era uma das atividades mais desgastante do processo. O gestor do auxílio saúde precisava navegar entre várias telas do SIAPE. Para cada beneficiário o gestor navegava entre algumas telas para informar que aquele devia receber o benefício. Estas tarefas juntas duravam alguns dias para serem executadas e precisavam ser divididas entre aproximadamente três servidores para que a mesma fosse finalizada em tempo hábil. Além disto, eram atividades extremamente mecânicas e suscetíveis a erros. Felizmente, com a implantação do sistema, o esforço foi reduzido drasticamente, passando a alocar apenas um servidor e sendo executada em aproximadamente 5 minutos.

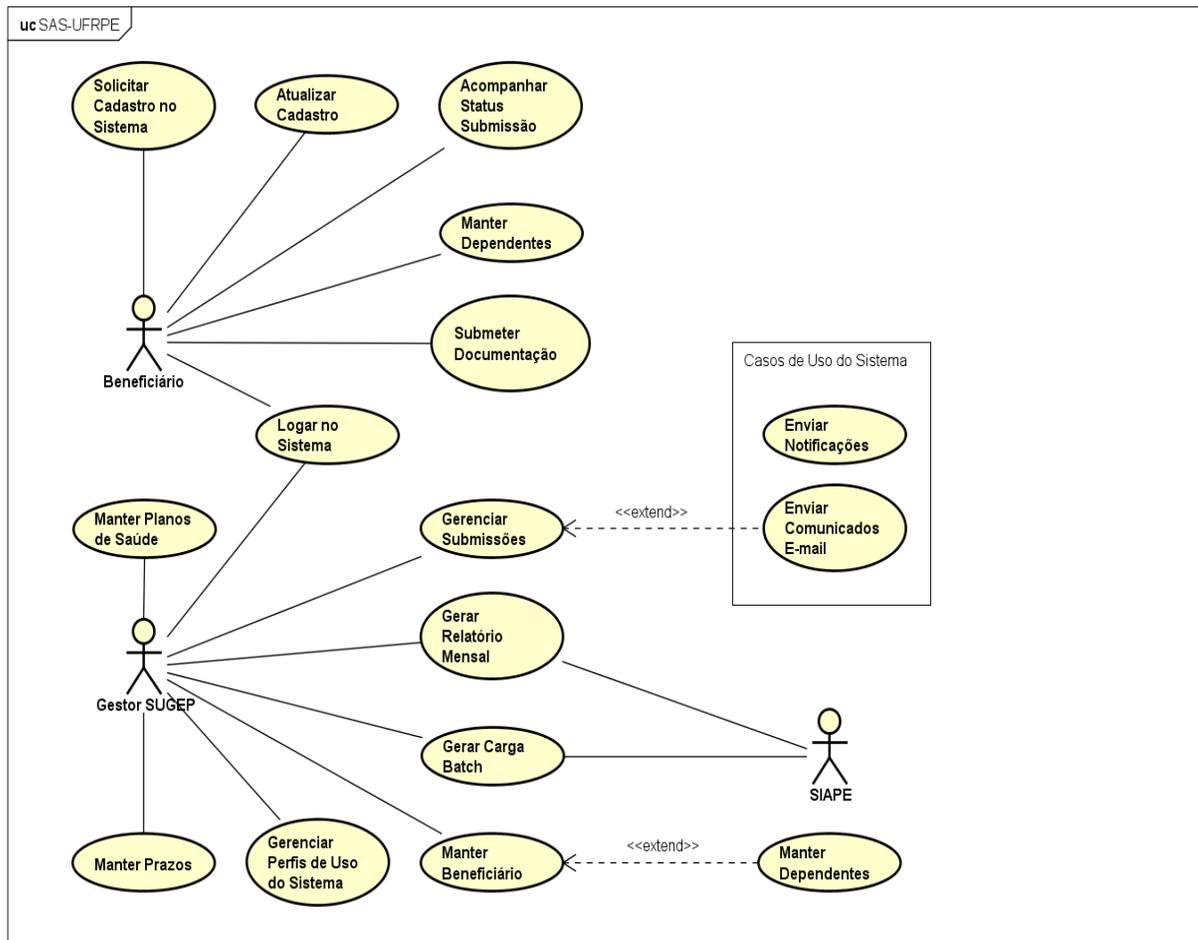
De uma forma geral, podemos sintetizar as vantagens advindas da implantação do processo de gestão do auxílio saúde, apoiado pelo SAS: redução drástica o tempo gasto fim a fim com processo; melhora da comunicação do processo, já que o beneficiário fica ciente do estado da sua documentação; diminui consideravelmente a possibilidade de falhas no processo; e por fim, permite que os servidores da UFRPE, sejam da SUGEP ou beneficiários que são servidores ativos, concentrem seu tempo disponível e energia em atividades que podem gerar ainda mais valor para a referida universidade.

3.2.1.3 Elicitação de requisitos – inicial

As anotações do primeiro encontro - juntamente com a modelagem de negócio - foram a base para o início das atividades de Elicitação de Requisitos. Esta atividade gerou uma modelagem inicial contendo as principais funcionalidades do sistema. O diagrama de Casos de Uso da Figura 8 apresenta, além destas funcionalidades, os atores que interagem com as mesmas - o

beneficiário, o gestor do auxílio saúde e o sistema SIAPE. Vale lembrar, que estas funcionalidades são apresentadas com mais detalhes no Apêndice I – Documento de Requisitos - e de forma sucinta na próxima seção.

Figura 8 - Diagrama de casos de uso para o SAS.



powered by Astah

Fonte : Elaborada pelo autor

O propósito do diagrama apresentado na Figura 8 foi comunicar ao cliente a proposta de um sistema que estava sendo idealizado pelo autor com base nas necessidades percebidas. A ideia era verificar se a concepção que nós tivemos do sistema cumpria com as expectativas do cliente.

Este documento foi apresentado aos clientes num segundo encontro justamente para validá-lo. O propósito desse segundo encontro foi refinar a modelagem inicial e os requisitos identificados e será descrito com mais detalhes na fase de elaboração.

3.2.2 Elaboração

3.2.2.1. Elicitação de Requisitos – Refinamento

Após o primeiro encontro, realizado na fase de iniciação, iniciou-se um processo de refinamento das necessidades que foram identificadas e das funcionalidades que foram levantadas de forma superficial na fase de Iniciação. O objetivo deste esforço foi de mapear as necessidades descobertas para os requisitos finais do sistema. Esses requisitos ou especificação são nada mais que funcionalidades que o sistema pode ter e restrições que ele deve obedecer.

Após uma análise minuciosa das necessidades, tal especificação do sistema foi compilada num documento de requisitos e pode ser analisado com mais detalhes no Apêndice I deste trabalho. O documento foi construído de forma a:

- Retomar o diagrama de casos de uso criado na atividade anterior (Modelagem Inicial);
- Identificar e detalhar a partes envolvidas na construção do sistema;
- Apresentar e descrever todas as funcionalidades identificadas;
- Priorizar as funcionalidades: quais são essenciais, importantes e desejáveis;
- Apresentar as restrições do sistema descobertas durante a reunião com o NTI: tecnológicas (linguagem de programação, frameworks), de segurança, confiabilidade, entres outras;

Embora as funcionalidades elicítadas sejam apresentadas com mais detalhes no Apêndice I, segue uma descrição breve sobre as mais importantes:

- **Submeter documentação:** permite ao beneficiário do auxílio saúde enviar comprovantes de pagamento referentes ao seu plano de saúde;
- **Acompanhar status submissão:** permite ao beneficiário acompanhar o status do envio da sua documentação, se foi aprovado ou reprovado;
- **Gerenciar submissões:** permite ao gestor do auxílio saúde verificar todas as submissões de documentação, podendo aprová-las ou reprová-las;
- **Gerar carga batch:** permite a criação de um arquivo codificado para importação no SIAPE. Tal arquivo contém a relação de beneficiários que devem receber o benefício;
- **Gerar relatório mensal:** permite a criação de uma relação contendo os beneficiários e respectivos dependentes, bem como o valor pago ao beneficiário por conta do benefício;

É importante entender que o esforço maior para se criar este documento se deu na primeira iteração desta fase. Entretanto, ao longo de todo o projeto esta atividade foi revisita, de forma que em cada uma dessas iterações o documento foi atualizado de acordo com o *feedback* do cliente para se adequar a real necessidade dele. Este artefato foi fundamental para guiar a elaboração e construção do sistema nas próximas fases.

3.2.2.2 Protótipo

Objetivando a descoberta de novas funcionalidades e problemas não identificados com os esforços para criação do documento de requisitos, o autor optou por usar a técnica de prototipação, a qual é descrita no Capítulo II.

Então, com base na especificação do sistema, nós desenvolvemos um protótipo não funcional da aplicação. No entanto, ele não continha nenhuma funcionalidade pronta, que pudesse ser utilizada pelo usuário. O propósito do protótipo foi mostrar de forma visual aos

clientes as funcionalidades que levantamos e verificar se esse entendimento era consonante com o que o cliente havia imaginado.

Além disto, a prototipação poderia captar aspectos que passaram despercebidos, estimulando a criatividade dos clientes para funcionalidades que poderiam melhorar ainda mais o processo de gestão do auxílio saúde, mas que não haviam sido pensadas até então.

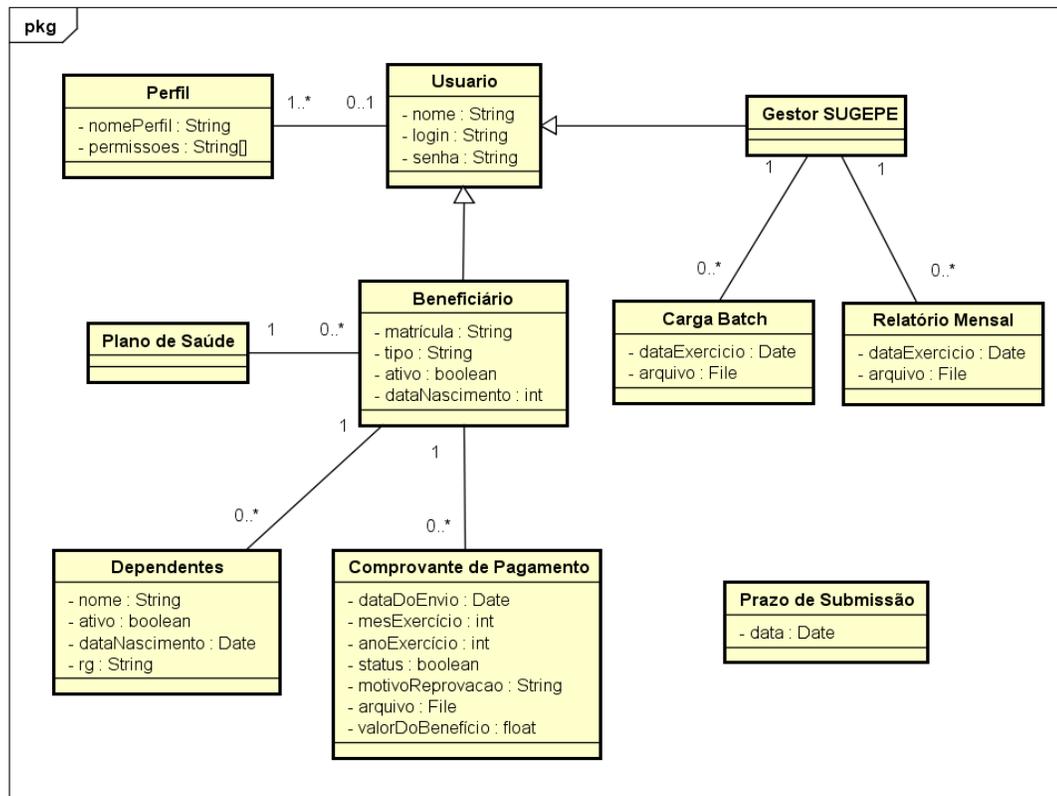
Após a construção do documento de requisitos e o protótipo, tais documentos foram validados pela equipe da SUGEP, numa nova visita à unidade. Dúvidas que surgiram ao longo das demais fases do desenvolvimento foram sanadas através de contatos por telefone e e-mail.

Sempre que as dúvidas eram sanadas, o Documento de Requisitos foi atualizado para refletir as mudanças. Este artefato foi atualizado durante todo o projeto. Nas fases seguintes, a construção de todas as funcionalidades do sistema foi baseada neste documento.

3.2.2.3 Análise e Projeto

As informações colhidas nas fases anteriores foram refinadas para dar origem a uma arquitetura conceitual do sistema. Já que o desenvolvimento do sistema foi planejado para ser orientado a objetos, a arquitetura básica do sistema foi representada através de um diagrama de classes. A Figura 9 ilustra o diagrama conceitual de classes gerado com base na especificação levantada nas fases anteriores.

Figura 9 - Diagrama de classes conceitual para o SAS.



powered by Astah

Fonte : Elaborada pelo autor.

De forma sucinta, o diagrama apresenta as seguintes entidades:

- **Perfil:** representa as permissões dos usuários para acessar recursos dentro do sistema;
- **Usuário:** representa as informações comuns aos gestores da SUGEP e os beneficiários do auxílio saúde;
- **Gestor SUGEPE:** representa um usuário do sistema que é o gestor do auxílio saúde;
- **Carga Batch:** representação do arquivo de carga batch e informações associadas;
- **Relatório Mensal:** representação do arquivo de relatório mensal e informações associadas;

- **Plano de Saúde:** representa as informações de determinado plano de saúde;
- **Beneficiário:** representa um usuário do sistema que é o beneficiário do auxílio saúde;
- **Dependente:** representa os indivíduos que dependem do beneficiário do auxílio saúde;
- **Comprovante de Pagamento:** representação do arquivo de comprovação de pagamento de plano de saúde, juntamente com informações associadas;
- **Prazo de submissão:** representa as informações a respeito do prazo de submissão do comprovante de pagamento referente ao auxílio saúde;

Sobre os relacionamentos entre as entidades do diagrama:

- Cada **Usuário** possui um ou mais **Perfis** de uso do sistema;
- O **Gestor SUGEP** pode gerar várias **Cargas Batch** ou **Relatórios**;
- Cada **Beneficiário** pode estar associado a apenas um **Plano de Saúde**;
- Cada **Beneficiário** pode ter vários **Dependentes** e fazer várias submissões de **Comprovantes de Pagamento**;

Nas fases seguintes, esse diagrama foi utilizado como base para a construção do sistema, sobretudo durante a construção da base de dados e a criação dos relacionamentos entre as entidades da aplicação.

3.2.3 Construção

Nesta fase os esforços se voltaram prioritariamente à combinação dos módulos do Drupal que coordenadamente se uniram para dar corpo ao sistema. Um grande esforço de pesquisa por módulos de terceiros pré-existentes (do núcleo e de contribuidores) foi realizado. Para tanto, foi analisada documentação disponibilizada para o módulo no site oficial do Drupal (www.drupal.org), bem como vídeos na plataforma Youtube (www.youtube.com) que demonstravam o módulo em funcionamento.

Após a descoberta dos módulos de terceiros, os módulos customizados foram escritos para realizar tarefas específicas dentro do sistema que não eram realizadas por outros módulos, ou porque simplesmente tal módulo de terceiro não foi encontrado durante a pesquisa.

3.2.3.1 Ambiente de desenvolvimento

Conforme foi acertado com o NTI, um ambiente de desenvolvimento foi disponibilizado para o desenvolvimento do sistema. Este ambiente era nada mais do que uma instância do Drupal que ficou acessível através da internet sob o endereço: <http://www.sugep.ufrpe.br/auxiliosaude/>. Através deste endereço era possível acessar uma interface web para a customização do sistema, por parte do desenvolvedor, bem como a utilização do mesmo por parte do cliente.

Além da interface web disponibilizada, foi concedido acesso remoto ao servidor que hospedava a aplicação - através de uma Virtual Private Network (VPN) e do protocolo SSH. Sem esse acesso não seria possível construir e testar, na instância do Drupal acima, os módulos customizados desenvolvidos.

3.2.3.2 Módulos do núcleo utilizados

Abaixo estão listados apenas os módulos do núcleo encontrados na pesquisa e habilitados no *Drupal* para utilização no projeto, junto com seus respectivos papéis/aplicações no SAS (A lista com todos módulos do núcleo, inclusive os listados abaixo, pode ser acessada em: <<https://www.drupal.org/node/1283408>>):

- **Actions:** Executa tarefas quando eventos específicos são acionados dentro do sistema. No sistema deste trabalho, é utilizado para disparar um e-mail após a avaliação da documentação comprovando gasto com plano de saúde;
- **CKEditor:** Permite a criação de um editor WYSIWYG. Este editor permite a pré-visualização da aparência final do texto que está sendo editado/formatado;
- **Datetime:** Permite a criação de campos em formulário que suportam data e hora;
- **Email:** Define um tipo de campo de e-mail simples;
- **Entity:** API para permitir a criação e gerenciamento de entidades. No desenvolvimento SAS - SUGEP foi utilizado para criar as entidades: Beneficiário, Dependentes, Plano de Saúde, Prazo, Submissão (Comprovante de Pagamento), Carga Batch e Relatório Mensal;
- **Entity Reference:** Permite a criação de um campo numa entidade que faz referência a outras entidades. No SAS - UFRPE foi utilizado para relacionar as entidades. Os relacionamentos foram criados com base no diagrama de classes conceitual construído na fase de Elaboração - Análise e Projeto;
- **Field:** Permite a adição de campos a entidades;
- **Field UI:** Interface de usuário para modificar os campos criados com o módulo Field;
- **Views:** Construtor de consultas que permite a criação de relatórios e listas complexos. No SAS - SUGEP foi utilizado para criar páginas com a listagem das informações das entidades;
- **Views UI:** Cria uma interface gráfica para gerenciamento das Views criadas;

- **Filter:** Permite a criação de filtros para os dados que estão sendo mostrado na visualização feita pelo módulo Views;
- **Block:** Permite a criação de Views dentro blocos que podem ser reutilizados em várias partes do sistema;
- **File:** Define um tipo de campo de arquivo;
- **Locale:** Adiciona a funcionalidade de manipulação de idioma e permite a tradução da interface do usuário para idiomas que não o inglês.;
- **Menu:** Permite aos administradores personalizar o menu de navegação do site;
- **Menu Link:** Permite que os usuários criem links de menu;
- **Options:** Define caixa de seleção e *widgets* de botão de opção para campos de texto e numéricos;
- **Path:** Permite a criação de URLs amigáveis;
- **Upload:** Permite aos usuários carregar e anexar arquivos. Utilizado no SAS - SUGEP para permitir a submissão dos comprovantes de pagamento referentes ao plano de saúde;
- **User:** Gerencia o sistema de registro e login do usuário;

3.2.3.3 Módulos de contribuidores utilizados

À medida que os módulos necessários a construção do sistema foram sendo descobertos o gestor de infraestrutura do NTI era notificado por e-mail, recebendo os *links* para *download* da versão correta do módulo, bem como um breve passo-a-passo para a instalação do mesmo.

Esse gestor, então, habilitava e instalava os módulos solicitados, uma vez que não foi concedida permissão de instalação de novos módulos ao desenvolvedor. Abaixo estão listados os módulos dos contribuidores encontrados na pesquisa, instalados e habilitados no *Drupal*, para utilização no projeto, junto com seus respectivos papéis/objetivos no sistema.

- **EntityForm:** Permite a construção de formulários para criação e modificação das entidades criadas com o módulo *EntityAPI*;

- **EntityRules:** Permite a criação de eventos que são disparadas assim que uma entidade é modificada. Utilizado para disparar a ação de envio de e-mail quando a Submissão de Comprovante é avaliada pelo gestor do auxílio saúde;
- **UniqueField:** Permite a criação de campos que não podem ter valores repetidos em diferentes registros;
- **Devel:** Utilizado para exibir mensagens de depuração mais elaboradas;
- **Datepicker:** Cria um *widget* mais elaborado que permite o usuário selecionar uma data, para um campo de um formulário, ao clicar num calendário;
- **Exposed Filter:** Permite que os filtros criados pelo módulo Filter sejam modificados pelo usuário. Utilizado na aplicação para permitir que o gestor do auxílio saúde defina seus próprios filtros. Por exemplo: beneficiário com determinada matrícula; submissões enviadas até determinado dia;
- **SMTP Authentication Support:** Responsável pela conexão do Drupal com servidor de e-mail SMTP. No SAS - SUGEP possibilita o envio de comunicados por e-mail referentes a aprovação/reprovação de comprovantes de pagamento, criação/ativação/cancelamento de contas e redefinição/lembrete de senha;

3.2.3.4 Dificuldades com módulos de contribuidores

Em todos os casos, os módulos dos contribuidores não realizavam toda a tarefa. Foi então, que o autor precisou escrever seus próprios módulos para concluir a implementação das funcionalidades.

Problemas com a pesquisa de módulos de contribuidores: descoberta tardia de que um módulo não conseguiria suportar completamente algumas funcionalidades.

Um exemplo desta dificuldade, ocorreu com um módulo muito importante: o módulo *ContentType* que permite a criação de entidades (por exemplo, Beneficiários, Dependentes, Relatórios, Carga Batch) foi adotado inicialmente para o projeto.

No entanto, após todas as entidades estarem criadas e parte do sistema estar implementado, descobriu-se tardiamente que o módulo não permitiria a criação de formulários de submissão de forma fácil para essas entidades. Praticamente, isso significa que não seria possível fazer operações de CRUD com as entidades, na própria interface que seria disponibilizada para os usuários (por exemplo, atualização dos dados cadastrais de um beneficiário do auxílio saúde).

Por conta disso, foi preciso encontrar um novo módulo que permitisse a representação das entidades e oferece uma interface *web* (que seria disponibilizado para todos usuários da aplicação) para edição e modificação das entidades. Felizmente, foi encontrado o módulo *EntityForm* que se propõe a fazer exatamente o que estava sendo procurado. Desta forma, toda aplicação precisou ser refeita, o que atrasou bastante o desenvolvimento.

3.2.3.5 Módulos customizados escritos

Uma vez que tanto os módulos do núcleo quanto os de contribuidores não eram suficientes para implementar as funcionalidades do SAS em sua totalidade, sobretudo as de geração de Carga Batch e Relatórios, foi preciso proceder com a escrita de módulos customizados. Abaixo estão listados todos os módulos customizados criados e a função de cada um deles dentro do sistema desenvolvido.

- **Mensagem de Prazo:** Permite a exibição de um bloco de texto ao longo de todo o sistema. Bloco este que exhibe o prazo de entrega (para o mês vigente) referente ao comprovante de pagamento do seu plano de saúde, com base no prazo que foi cadastrado para o mês vigente;
- **Gerar Carga Batch:** Responsável por recuperar os dados dos beneficiários que tiveram a documentação aprovada no mês vigente e gerar o arquivo de Carga Batch que é aceito pelo sistema de folha de pagamento do Governo Federal (SIAPE);
- **Gerar Relatório:** Módulo responsável por recuperar e exibir os dados a respeito do pagamento do plano de saúde e dos beneficiários que submeteram comprovantes de pagamento dentro do prazo para o mês informado;

- **Redirecionamento EntityForm:** Os formulários de criação/alteração de entidades sempre direcionam o usuário para uma interface do administrador do site, a qual destoa da interface da aplicação. Portanto, foi criado um módulo que redireciona o usuário para uma página que exibe todas as entidades do tipo criado/alterado.

Por exemplo, um usuário ao preencher os dados de um novo Dependente e salvá-lo, o sistema passa a redirecioná-lo para uma tela customizada que exibe todos Dependentes;

- **DAO Dependente:** Responsável por recuperar os dados referentes aos dependentes;
- **DAO Beneficiário:** Responsável por recuperar os dados referentes aos dependentes;
- **Carga de Usuários:** Responsável por fazer o cadastro em lote de vários usuários. Permite a inserção de vários beneficiários do auxílio saúde na base de dados do *Drupal*;

As telas, desenvolvidas nesta fase, são apresentadas no Apêndice II deste documento.

3.2.4 Transição

Esta seção detalha os detalhes da entrega do sistema aos clientes, bem como o treinamento dos usuários para utilização adequada do **produto** entregue.

3.2.4.1 Implantação

A implantação do software ocorreu sempre no ambiente de desenvolvimento disponibilizado pelo NTI, o qual foi apresentado na fase de iniciação. Da mesma forma que preconiza o RUP, a implantação do *software* ocorreu de forma iterativa: revisitada várias vezes e incremental: integrando uma nova parte funcional ao software principal.

À medida que novas funcionalidades ficavam prontas os usuários eram notificados para poder validá-las e testá-las. Objetivo era verificar se a funcionalidade cumpria com suas expectativas e até mesmo encontrar falhas não detectadas na fase de construção. Sempre que havia alguma incoerência naquilo que estava sendo entregue, os clientes tomavam notas e nos

apresentavam de maneira mais informal (através de e-mail, telefone ou em aplicativos de mensagens de texto para celular) ou mais formal ao redigir um documento contendo os ajustes a serem realizados no sistema para que o mesmo se adequasse às suas expectativas/necessidades.

Após uma dessas entregas nós realizamos uma reunião remota com um dos clientes, na qual ambos compartilharam a mesma tela com objetivo de compreender de forma mais clara e direta algumas observações do cliente a respeito da entrega. O cliente aproveitou e compilou todas suas observações em um documento. Tais observações alimentavam a iteração seguinte da fase de construção. Foi quando nós, além de construirmos as novas funcionalidades da próxima entrega do *software*, modificamos o sistema para que o mesmo se adequasse a todas as observações elencadas pelo cliente.

É importante mencionar que a implantação do sistema aconteceu num ambiente de testes. A estrutura montada neste ambiente foi primordial para colocarmos o sistema à prova através de um projeto piloto. Este piloto permitiu que usuários reais utilizassem o sistema numa situação real enquanto monitoramos o comportamento do sistema em busca de pontos de falhas e melhorias. Após correção de erros e execução das melhorias, esta estrutura implantada num ambiente de testes será migrada para um ambiente de produção, onde todos os usuários poderão utilizar o sistema. A Seção 1 do Capítulo IV aborda com detalhes o projeto piloto executado a fim realizar a validação do sistema.

3.2.4.2 Treinamento

Após cada entrega, o cliente era notificado por e-mail para testar as novas funcionalidades disponibilizadas. Estrategicamente, nós pedíamos para que o mesmo executasse aquela funcionalidade sem nossa intervenção. O propósito desta abordagem era justamente verificar se a funcionalidade esta tão intuitiva a ponto do usuário não precisar de nossa intervenção para executá-la.

Felizmente todas elas foram muito bem compreendidas, de forma que em nenhum momento foi necessária à nossa intervenção a fim de realizar um treinamento formal com o usuário. Nas poucas ocasiões em que foi necessária nossa intervenção, o cliente nos notificava

por e-mail ou via ligação telefônica para sanar pequenas dúvidas a respeito da utilização das funcionalidades recém disponibilizadas. Então, após breves observações, as funcionalidades em questão ficaram claras.

É importante apontar que apenas os gestores do auxílio saúde acompanharam o desenvolvimento e receberam o software de forma incremental. Desta forma, eles foram sendo instruídos a utilizar o software. Já os beneficiários, que não tiveram esta oportunidade, foram instruídos a utilizar o software através da divulgação de um pequeno roteiro a ser seguido, o qual foi divulgado por e-mail para os atuais beneficiários e para os próximos que vierem a aderir ao benefício no futuro.

3.3 LIÇÕES APRENDIDAS (SEÇÃO SECUNDÁRIO)

Os esforços para elaboração de um novo processo, assim como a construção de um sistema para apoiá-lo, nos ensinou algumas lições, as quais devem ser consideradas em futuros projetos com o mesmo propósito deste trabalho: automatização de processos dentro da UFRPE. Estas lições são listadas abaixo.

O uso do *Drupal* foi imposto pelo NTI da UFRPE. Sua utilização por desenvolvedores que nunca tiveram contato com o gerenciador de conteúdo pode gerar uma dificuldade considerável nas fases iniciais. Portanto, caso não seja possível utilizar outra tecnologia com qual o desenvolvedor possui mais afinidade, é recomendado que o estudo do *Drupal* seja iniciado o quanto antes para que os prazos de entrega do sistema não sejam ultrapassados.

Também percebemos a importância de iniciarmos e validarmos a modelagem antes de iniciarmos a codificação. Desta forma, raras vezes tivemos a necessidade de refazer alguma funcionalidade porque a mesma não tinha sido entendida na fase de modelagem. Esta prática permitiu que entregássemos no sistema, na maioria das vezes, o que satisfazia as reais necessidades do problema.

Com relação autenticação de usuário no sistema, percebemos que o cenário ideal é aquele onde há a autenticação através de uma base única da instituição, neste caso a UFRPE. É inviável que cada nova aplicação implantada sob o domínio da universidade gere novas credenciais de acesso para os usuários. É preciso utilizar, em parceria com o NTI da UFRPE, o mecanismo de

serviços integrados disponibilizado pela universidade para que todas as aplicações possuam um mecanismo único de autenticação, evitando assim uma explosão de credenciais de uso, que a longo prazo seria impraticável.

Enfim, com relação ao cadastro de usuários em aplicações que envolvem servidores federais, é importante que haja um mecanismo de integração com sistemas do governo que armazenam informações sobre estes servidores. Desta forma, os dados já cadastrados podem ser reaproveitados, de forma que os dados estejam sempre consistentes.

CAPÍTULO IV - AVALIAÇÃO DOS RESULTADOS

O objetivo deste capítulo é verificar se o processo de gestão do auxílio saúde, automatizado pelo SAS, possui um desempenho satisfatório, bem como identificar possibilidades de melhorias não identificadas durante as iterações de desenvolvimento. Para isso, idealizamos um projeto piloto com intuito de por o sistema que apoia o processo à prova, pois o sistema é a peça chave do processo. Entendemos que a boa avaliação do sistema é um passo primordial para a boa avaliação do processo como um todo. As seções abaixo descrevem as etapas do processo de avaliação realizado.

4.1 AVALIAÇÃO DO SISTEMA

A Seção 1.1 detalha as estratégias definidas para avaliação a fim de se obter o resultado mais fiel possível. A Seção 1.2 define a técnica definida para se obter os critérios de qualidade referente ao SAS. A Seção 1.3 define a concretização do plano estabelecido na fase de Planejamento, juntamente com a execução da técnica. A Seção 1.4 apresenta os resultados desta avaliação. A Seção 1.5 discute o que foi inferido a partir dos resultados obtidos.

4.1.1 Planejamento

Para fundamentar esta avaliação baseamo-nos nas diretrizes de qualidade definidas na ISO/IEC 9126-1 (LÚCIO ANDRÉ MENDONÇA DOS ANJOS, 2009) para definirmos um processo de avaliação que colocasse sob teste os aspectos mais relevantes associados à qualidade do sistema. Desta forma, com base na ISO planejamos a avaliação para que esta verificasse os seguintes aspectos do processo e do SAS:

- **Funcionalidade:** O conjunto de funções que atendem às necessidades dos usuários. O SAS possui todas as funcionalidades que são necessárias à resolução do problema?
- **Confiabilidade:** Evidencia a capacidade do produto de manter seu desempenho ao longo do tempo e em condições estabelecidas. O SAS é imune a falhas? O SAS faz o que deve da maneira correta?
- **Usabilidade:** Facilidade para a utilização do produto. O SAS é fácil de usar?

Para identificar os aspectos acima descritos, decidimos colocar o SAS em condições reais de operação, com um conjunto reduzido de usuários. A ideia foi realizar um projeto piloto para verificar o comportamento do sistema numa situação mais próxima possível da realidade que o mesmo iria enfrentar ao ser colocado em operação de verdade.

- **Participantes da avaliação:**

Inicialmente, planejamos a escolha das pessoas que participariam do teste. Neste momento, definimos que seria importante a escolha de diferentes perfis de usuários, a fim de tornar os resultados da avaliação o mais próximo possível da realidade. Desta forma, decidimos que o teste contemplaria, além dos gestores do auxílio saúde, algumas dezenas de beneficiários, sendo estes beneficiários distribuídos de forma proporcional entre servidores ativos, servidores aposentados e pensionistas.

A SUGEP fez a escolha dos participantes, pois mantinha o contato de vários possíveis candidatos à avaliação, além de conhecer o perfil e disponibilidade dos mesmos. Além disso, a SUGEP optou por limitar bastante a quantidade de participantes, o motivo: a ativação dos servidores de forma manual é muito laboriosa e leva um tempo considerável para ser finalizada. Caso a ativação automática dos beneficiários, realizada pelo SAS através da Carga Batch, falhasse, se tivéssemos escolhido muitos participantes, correríamos o risco de não conseguirmos efetuar-los dentro do prazo, uma vez que a avaliação ocorreria próxima a data de fechamento da folha de pagamento.

- **Local de realização do teste:**

Como o SAS é um sistema *web* disponível através de um domínio público, achamos interessante que cada participante fizesse a avaliação do sistema no local de sua preferência, mais uma vez para aproximar o projeto piloto da realidade, uma vez que os beneficiários normalmente farão a utilização do sistema em suas casas ou em qualquer outro lugar, sem a interferência dos avaliadores.

4.1.2 Metodologia

A fim de se obter as métricas de qualidade do sistema, optamos por realizar um projeto piloto do sistema, como se o mesmo já estivesse em operação. Para isso, primeiramente, definimos um roteiro a ser seguido pelos participantes dentro do sistema a fim de testá-lo. Considerando que o sistema possui dois perfis de usuário - beneficiário e gestor do auxílio saúde - criamos dois roteiros, uma para cada perfil.

O primeiro roteiro contém as principais atividades a serem executadas pelos beneficiários dentro do sistema:

- Preencher cadastro de dados pessoais;
- Cadastrar dependentes (quando houver);
- Enviar comprovante de pagamento;
- Acompanhar o status do comprovante.

Já o segundo roteiro, o fluxo de atividades mais comum a ser realizado pelos gestores do auxílio saúde:

- Avaliar os comprovantes de pagamento enviados pelos beneficiários;
- Gerar Carga Batch;
- Gerar Relatório;

- Importar arquivo de Carga Batch para SIAPE.

Em seguida, definimos um questionário a ser respondido pelos participantes após a execução da avaliação. Este questionário foi definido estrategicamente para que ao final da avaliação as medidas referentes aos critérios de qualidade perseguidos (definidos pela ISO 9126-1) fossem mensuradas, bem como outras características desejáveis ao sistema.

A Tabela 2 apresenta as perguntas elaboradas, juntamente com o propósito da pergunta. Os termos em negrito remetem aos critérios da ISO.

Tabela 2 - Questionário de avaliação do SAS.

Perguntas	Alternativas	Propósito
Você é um(a):	-Docente (Aposentado) -Docente (Ativo) -Técnico (Aposentado) -Técnico (Ativo) -Pensionista	Ajuda a identificar a distribuição das respostas entre os diferentes tipos de beneficiários.
Você considera o seu nível de experiência com sistemas computacionais como:	-Experiente -Intermediário -Básico -Nenhum	Usabilidade: Esta pergunta ajuda a filtrar os diferentes perfis de usuários. Desta forma, podemos identificar, por exemplo, que a dificuldade de algum usuário em utilizar o SAS deve-se à pouca experiência do usuário e não ao sistema em si.
O SAS propõe-se a fazer o que é apropriado.	-Concordo -Concordo	Funcionalidade: Verifica se o sistema dispõe de todas as

	parcialmente -Parcialmente -Discordo Parcialmente -Discordo	ferramentas que são necessárias às execuções das tarefas.
O SAS faz o que foi proposto de forma correta.	-Concordo -Concordo Parcialmente -Parcialmente -Discordo Parcialmente -Discordo	Funcionalidade: As ferramentas dispostas no SAS funcionam.
O SAS está de acordo com as normas, leis etc.	-Concordo -Concordo Parcialmente -Parcialmente -Discordo Parcialmente -Discordo -Não se aplica	Utilizado para que o gestor do auxílio saúde informasse se o SAS atende a legislação.
O SAS evita acesso não autorizado aos dados.	-Concordo -Concordo Parcialmente -Parcialmente -Discordo Parcialmente -Discordo	Confabilidade: Verificar se os usuários sentem-se seguros ao utilizar o SAS.
É fácil entender o conceito e	-Concordo	Usabilidade: Verificar se o

a aplicação do SAS.	-Concordo Parcialmente -Parcialmente -Discordo Parcialmente -Discordo	propósito de cada funcionalidade dentro do SAS é claro.
É fácil aprender a usar o SAS.	-Concordo -Concordo Parcialmente -Parcialmente -Discordo Parcialmente -Discordo	Usabilidade: Verificar se o usuário consegue aprender a utilizar o SAS rapidamente.
É fácil de operar e controlar o SAS.	-Concordo -Concordo Parcialmente -Parcialmente -Discordo Parcialmente -Discordo	Usabilidade: Verificar a facilidade de utilização do sistema.
Em comparação com o antigo processo de solicitação de auxílio saúde, dê uma nota para o SAS (1 - pior nota, 10 - melhor nota).	1 a 10	Definida para que o participante ajude a destacar os ganhos advindos da implantação do SAS. Esta pergunta não afirma o SAS como a melhor solução possível, apenas ajuda a verificar se realmente houve melhorias com relação ao processo

		antigo. Esta mesma pergunta pode ser reformulada e refeita no futuro para comparar uma nova versão do SAS com sua versão anterior.
--	--	--

4.1.3 Execução

Após a definição da metodologia, demos início ao processo de avaliação. Para isso, juntamente com os gestores do auxílio saúde, enviamos um e-mail para alguns beneficiários contendo: o convite para participação da avaliação; a explicação do propósito da avaliação; e o roteiro a ser seguido em caso de aceite. Então, a maioria, dezessete, deles aceitou participar da avaliação. Após o cumprimento do roteiro, tínhamos dezessete comprovantes de pagamento referente ao auxílio saúde submetidos no sistema.

Em seguida, foi solicitado a um dos gestores que seguisse seu roteiro. Desta forma, o mesmo realizou a avaliação de todos comprovantes. É importante salientar que apesar do caráter de teste deste projeto piloto, a documentação enviada pelos beneficiários foi real e, portanto, avaliada de verdade. Sendo assim, após a avaliação, o gestor gerou a Carga *Batch* e aqueles beneficiários que entregaram a documentação correta foram ativados de fato no SIAPE, recebendo assim o valor referente ao auxílio saúde no mês seguinte.

Ao final de toda avaliação, apenas oito participantes responderam ao questionário que foi solicitado. A fim de se gerar um resultado mais satisfatório, o e-mail com o *link* do questionário foi reenviado, pedindo a colaboração dos participantes que ainda não haviam respondido. Após o e-mail, mais três participantes responderam ao questionário, totalizando onze respostas. De posse de uma taxa de respostas de aproximadamente 60%, optamos em partir para a avaliação dos resultados.

4. 1.4 Resultados

Todos os envios de comprovante de pagamento foram realizados com sucesso no SAS, com exceção de um candidato que ultrapassou o prazo de envio, desta a forma esta submissão será considerada apenas no mês seguinte. Em seguida, o gestor do auxílio saúde realizou a avaliação destes comprovantes. Após a avaliação, o gestor gerou o arquivo de Carga Batch. Então, este arquivo foi importado para o SIAPE, através da funcionalidade ativação em lote, para que todos beneficiários fossem ativados de uma só vez.

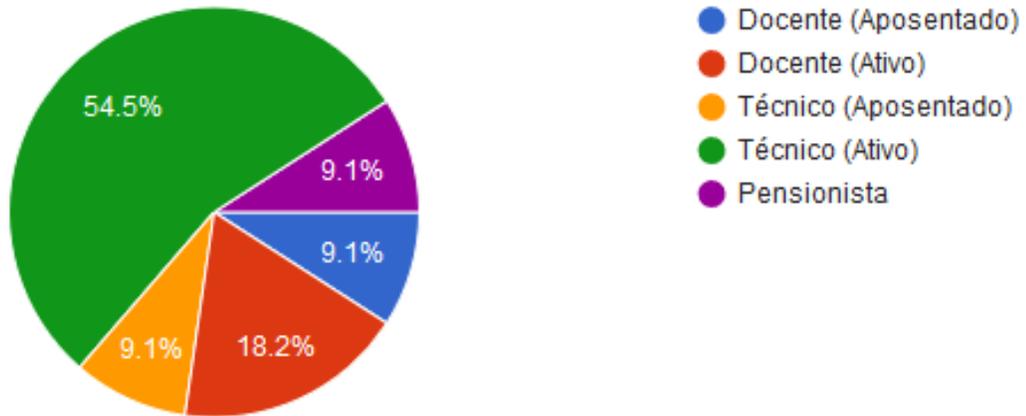
Como o SIAPE acusa o resultado da carga apenas no dia seguinte, o gestor acessou o SIAPE, no outro dia e verificou que a importação via Carga *Batch* havia funcionado perfeitamente. Sendo, assim todos beneficiários foram ativados e receberam o valor referente ao auxílio saúde na folha de pagamento do mês seguinte.

4.1.4.1 Resultado do Questionário

A seguir nós sumarizamos as respostas de cada uma das perguntas do questionário em termos percentuais.

1 Você é um(a):

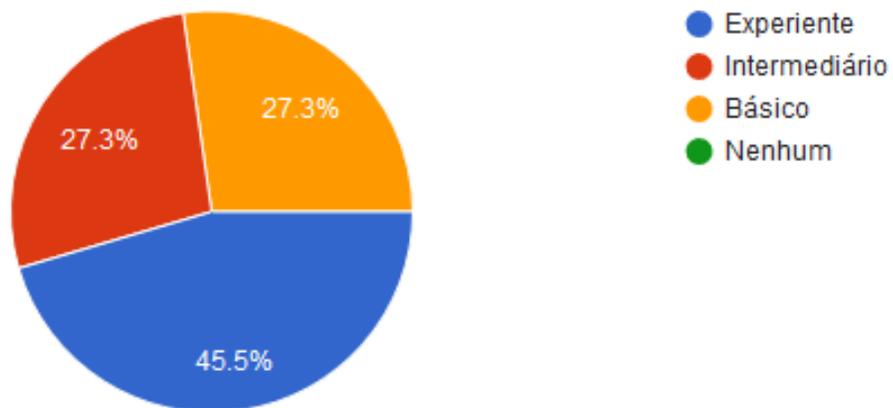
Figura 10- 1ª Pergunta do questionário de avaliação.



Fonte : Elaborada pelo autor.

2 Você considera o seu nível de experiência com sistemas computacionais como:

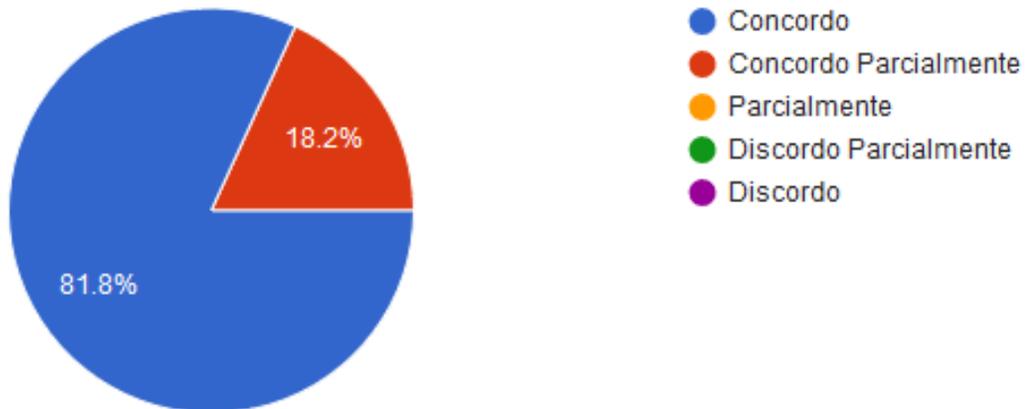
Figura 11- 2ª Pergunta do questionário de avaliação.



Fonte : Elaborado pelo autor.

1 O SAS propõe-se a fazer o que é apropriado.

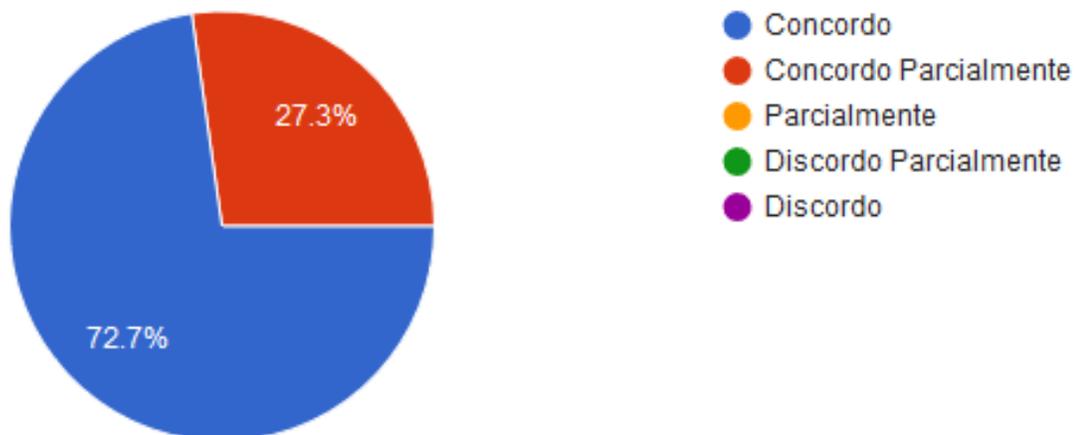
Figura 12- 3ª Pergunta do questionário de avaliação.



Fonte : Elaborada pelo autor.

4 O SAS faz o que foi proposto de forma correta.

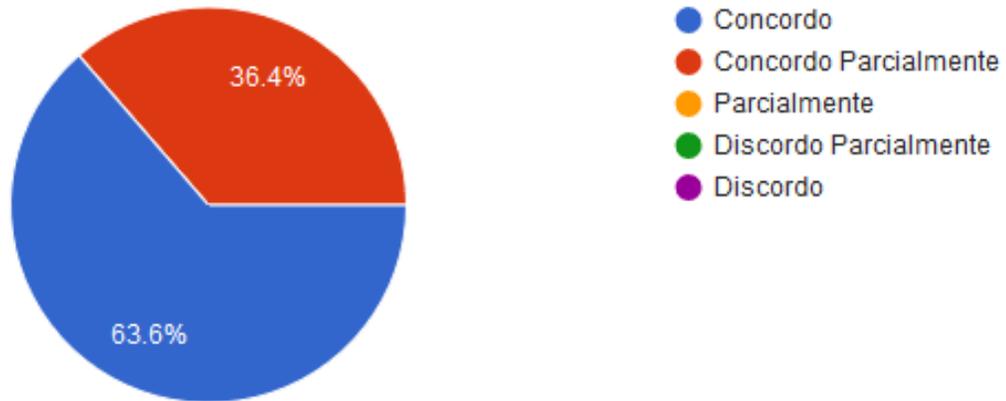
Figura 13- 4ª Pergunta do questionário de avaliação.



Fonte : Elaborada pelo autor.

5 O SAS está de acordo com as normas, leis etc.

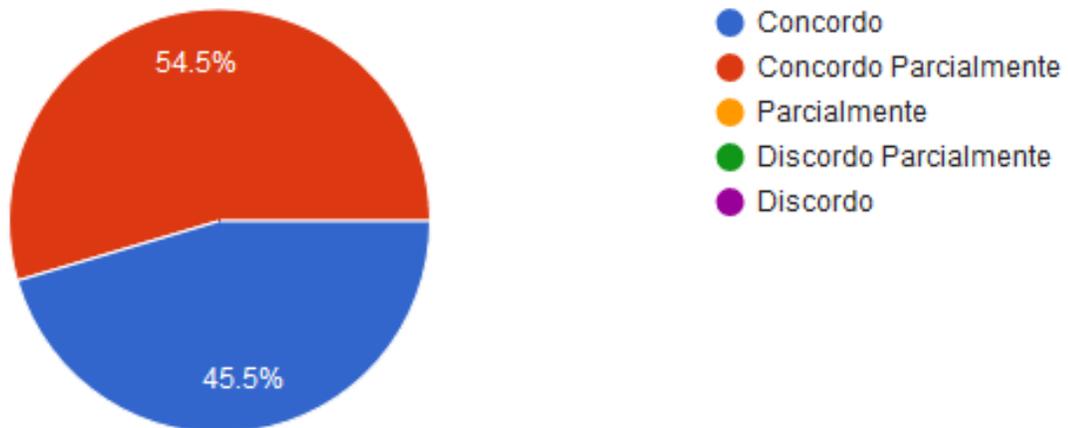
Figura 14- 5ª Pergunta do questionário de avaliação.



Fonte : Elaborada pelo autor.

6 O SAS evita acesso não autorizado aos dados.

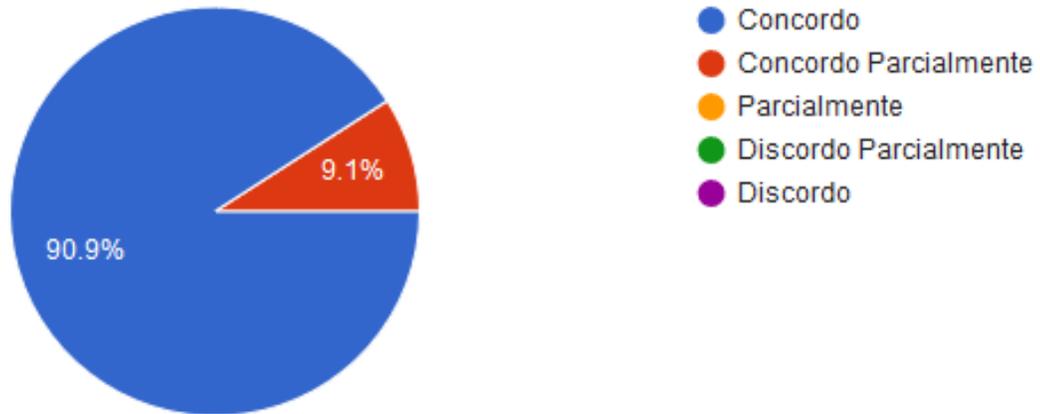
Figura 15- 6ª Pergunta do questionário de avaliação.



Fonte : Elaborada pelo autor.

7 É fácil entender o conceito e a aplicação do SAS.

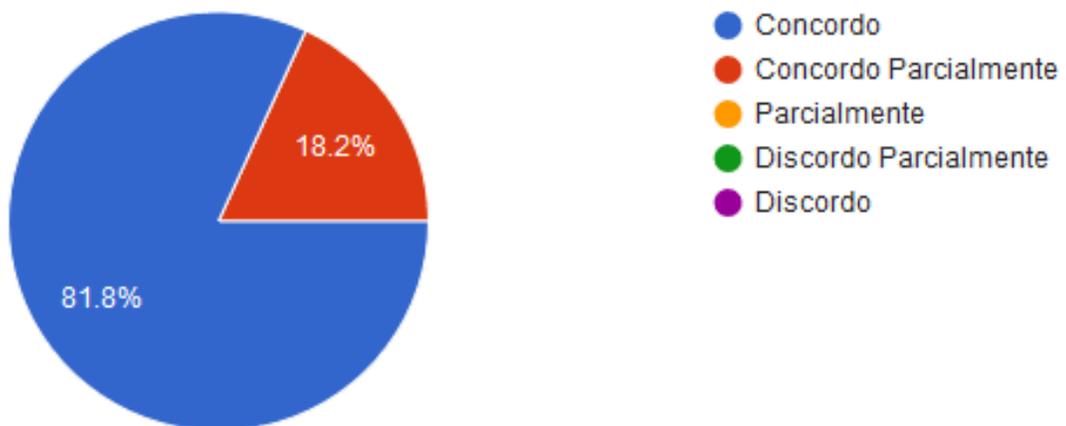
Figura 16- 7ª Pergunta do questionário de avaliação.



Fonte : Elaborada pelo autor.

8 É fácil aprender a usar o SAS.

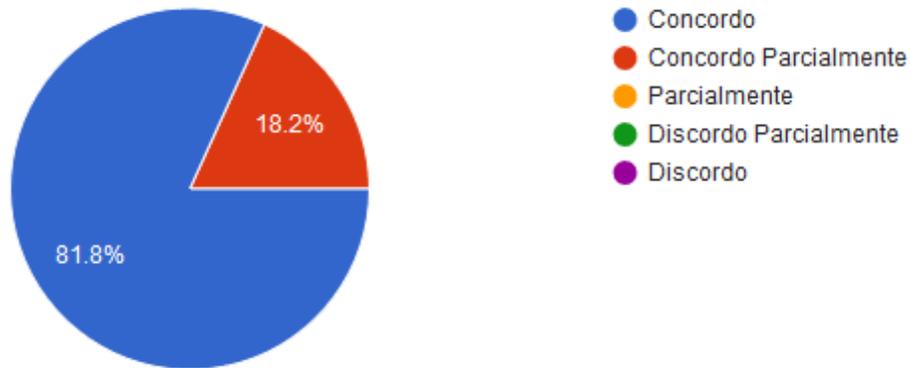
Figura 17- 8ª Pergunta do questionário de avaliação.



Fonte : Elaborada pelo autor.

9 É fácil de operar e controlar o SAS.

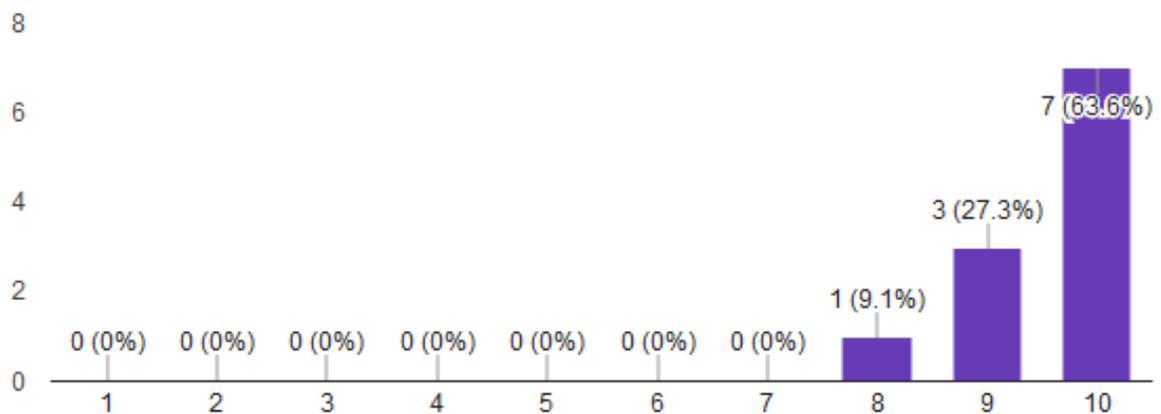
Figura 18- 9ª Pergunta do questionário de avaliação.



Fonte : Elaborada pelo autor.

10 Em comparação com o antigo processo de solicitação de auxílio saúde, dê uma nota para o SAS (1 - pior nota, 10 - melhor nota).

Figura 19 -10ª Pergunta do questionário de avaliação.



Fonte : Elaborada pelo autor.

4. 1.4.2 Dúvidas enviadas por e-mail

Apenas um participante enviou uma dúvida por e-mail. O questionamento foi com relação ao cadastro do dependente, pois o participante não sabia que opção escolher para o campo Tipo de Dependente.

4.1.5 Conclusões

Com base nos resultados do projeto piloto, respostas dos questionários e *feedback*/questionamentos por e-mail, chegamos as seguintes conclusões referentes a vários aspectos que foram colocados sobre avaliação. Dado, que a quantidade de participantes não foi tão expressiva, não podemos entender os resultados abaixo como verdades, mas sim tendências:

- **Usabilidade:** a maioria dos beneficiários demonstrou facilidade na utilização do sistema. Chegamos esta conclusão, uma vez que a maioria dos participantes não precisou de nossa ajuda para realizar as operações dentro do sistema e enviar a documentação. Além disso, as respostas do questionário referentes à usabilidade indicam que o sistema tende a ter uma facilidade de uso relativamente alta.
- **Funcionalidade:** o projeto piloto permitiu que as funcionalidades essenciais do sistema fossem colocadas à prova. O sistema se saiu muito bem, tendo cumprido com toda especificação necessária a realização da avaliação. Além disso, através das respostas do questionário, percebe-se que a maior parte dos participantes demonstrou que o SAS dispõe de todas as ferramentas para executar o que lhes fora solicitado e que estas funcionam adequadamente. Desta forma, observamos a tendência do SAS em se mostrar perfeitamente funcional.
- **Confiabilidade:** acreditamos que algumas pessoas concordaram parcialmente com segurança do sistema, pois estão cientes de que não é possível prever todas as possibilidades de invasão e acesso indevido a dados. Além disto, não conhecem a estrutura interna do sistema e, desta forma, não sabem se o mesmo está preparado. No entanto, acreditamos que elas não discordaram, pois percebem que realmente

não é possível acessar informações cujo acesso não lhes foi concedido. Assim, concluímos que o SAS deve ser trabalhado a fim de manter a continuidade neste sentimento de confiabilidade, para que ao longo do tempo todos usuários concordem plenamente com a segurança do sistema.

- **Melhorias:** identificamos oportunidades de melhorias, sobretudo nas funcionalidades que causaram confusão aos participantes. Essas melhorias foram anotadas pelo gestor da SUGEP e repassadas para nós. Algumas das melhorias apontadas já foram concretizadas numa nova versão do SAS, outras serão implementadas nas próximas iterações do desenvolvimento. Ainda vale lembrar que a manutenção do sistema não encerra com a entrega deste trabalho. As melhorias apontadas estão compiladas no Apêndice I deste trabalho, algumas delas são: modificação de texto das opções de tipos de dependentes para torná-lo mais intuitivo; limitação da quantidade de caracteres do campo de matrícula do beneficiário para evitar erros na geração da Carga *Batch*;

Outro reflexo da melhoria advinda da implantação do SAS para automatizar o processo pode ser visto nas respostas da pergunta 10 (dez) do questionário, a qual pede para que seja feita a avaliação do sistema em contraposição à abordagem antiga. Neste momento, os participantes avaliaram positivamente o uso do SAS em comparação com o processo anterior. Acreditamos por parte dos beneficiários que esse comportamento se deve à comodidade que o sistema trouxe aos mesmos ao permitir que a entrega da documentação seja feita de forma rápida e a partir de qualquer lugar.

Já por parte dos gestores do auxílio saúde, imaginamos que esta avaliação positiva se deu por conta da diminuição drástica percebida por eles no tempo despendido com a ativação dos beneficiários.

Enfim, esta avaliação foi primordial para verificarmos se a solução proposta para automatização do processo de gestão do auxílio saúde tende a ser satisfatória. De uma forma geral, dada a quantidade limitada de participante, percebemos que o sistema tende a se comportar bem, cumprindo de forma justa com o propósito para o qual foi projetado. Ademais, o sistema

demonstrou uma tendência na sua facilidade de utilização e suficiência para atender a todas as necessidades dos usuários, pelo menos nas atividades que foram propostas para a avaliação, além de cumprir com seu principal objetivo: diminuir drasticamente o tempo gasto com a ativação dos usuários.

O tempo gasto com a ativação de todos beneficiários foi reduzida drasticamente de várias horas para pouquíssimos minutos. Com a implantação do SAS, o gestor do auxílio saúde precisa apenas acessar o SAS e clicar em alguns botões para gerar um arquivo de Carga Batch. Em seguida, dentro do SIAPE este arquivo é importado num intervalo de tempo bem pequeno, enfim concluindo a ativação de todos beneficiários. Esta nova abordagem diminui drasticamente o trabalho mecânico e propenso a erros que era executado anteriormente. Após observação do sucesso desta nova abordagem, os gestores do auxílio saúde nos demonstraram sentimento de muito agradecimento e otimismo com relação à proposta do novo processo de gestão do auxílio saúde apoiado pelo SAS.

CAPÍTULO V – CONCLUSÃO

Este capítulo aponta as contribuições deste trabalho, bem como suas limitações. Além disto, são apresentados os trabalhos futuros que podem ser encabeçados a partir destas contribuições. Por fim, as dificuldades encontradas são elencadas.

5.1 CONSIDERAÇÕES FINAIS

Inicialmente, levantamos o seguinte questionamento: “é possível melhor a eficiência do processo de gestão do auxílio saúde?”. Após a elaboração de um novo processo, automatizado por um sistema de informação, avaliação e análise dos resultados da avaliação, percebemos que o processo proposto possui um grande potencial de eliminar os problemas apresentados pelo antigo processo.

O novo processo eliminou drasticamente o tempo despendido na realização de uma das atividades mais custosas, “Registro do Reembolso no SIAPE”. Esta atividade que antes levava várias horas para ser executada por três servidores da SUGEP agora leva aproximadamente cinco minutos para ser finalizada. Embora a funcionalidade de “Verifica Valor Reembolsado ao Beneficiário” ainda não tenha sido completamente automatizada, uma vez que a integração dos SAS com SIAPE se mostrou perfeitamente funcional, esta atividade, assim como a mencionada anteriormente, também será realizada em apenas 5 minutos após a conclusão de sua automatização.

Estas vantagens, advindas da implantação do novo processo, além da diminuição drástica do tempo: permite a aplicação mais eficiente da força de trabalho dos servidores da SUGEP, os quais podem utilizar das suas outras atribuições do cargo para gerar mais valor para universidade; melhora a comunicação do processo de comunicação, pois os beneficiários são avisados com antecedência sobre erros nos comprovantes de pagamento; evita a perda de documentos, muito comum no processo antigo e que prejudicava o recebimento do benefício.

Além do processo em si, este trabalho acabou gerando alguns subprodutos que podem ser reaproveitados por terceiros para futuras colaborações com o problema abordado por esta pesquisa, são eles:

- **Modelagem do processo:** esta modelagem está presente no Apêndice I. Nela, desenhamos o processo antigo e a nova proposta de processo. Desta forma, é possível que terceiros entendam o problema a ponto de detalhar ainda mais a modelagem ou até mesmo elaborar uma especificação para o sistema completamente independente da criada por neste trabalho.
- **Especificação inicial do sistema (Apêndice I):** foi gerada de forma que outros interessados, inclusive o NTI, possam construir um sistema inteiramente novo (por exemplo, com outra tecnologia diferente do *Drupal*: Java EE ou PHP puro) utilizando como base esta especificação inicial. Ainda é possível evoluir tal especificação de forma a atender os anseios da Universidade.
- **Sistemática para Automatização de Processos:** uma vez que a sistemática utilizada por este trabalho se mostrou eficiente para automatização do processo, esta sistemática pode ser utilizadas em outros contextos, para automatização de outros processos dentro do setor público.
- **Sistema de Informação:** um sistema construído com o apoio do sistema gerenciador *Drupal* foi construído com suas principais funcionalidades. Este pode ser evoluído de forma a implementar as novas funcionalidades que venham a existir, caso a especificação mude. As telas do sistema podem ser consultadas no Apêndice II.

5.2 LIMITAÇÕES E TRABALHOS FUTUROS

Apesar das partes principais do processo estarem automatizadas, uma atividade ainda não foi automatizada completamente: “Verifica Valor Reembolsado ao Beneficiário”. Esta atividade permite a criação de um relatório mensal contendo a relação de valores reembolsado a cada beneficiário.

Vale lembrar, mais uma vez, que do processo automatizado não irá encerrar com a entrega deste trabalho. Ainda com o nosso apoio, o sistema desenvolvido deve evoluir de acordo tanto com as necessidades institucionais, como também para a entrega da resolução das limitações citadas anteriormente e ajustes que precisem ser feitos. Além disto, realizaremos a transição da responsabilidade de manutenção do sistema para o NTI. Demais esforços, seguintes a finalização deste trabalho, inclui:

- **Verificação de necessidade de autenticação de documentos:** é necessário estudar a legislação para verificar se apenas um documento do tipo *pdf* é suficiente para o envio dos comprovantes de pagamento ou é necessário adicionar ao sistema algum mecanismo de autenticação a fim de confirmar a veracidade da documentação enviada.
- **Análise de viabilidade de integração com SIGEPE:** é importante avaliar a existência de algum mecanismo disponibilizado pelo SIGEPE para recuperação de dados referentes aos servidores federais. Caso exista, será possível criar no sistema desenvolvido por nós um cadastro automatizado de usuários que são servidores da UFRPE, eliminando assim a necessidade de replicação de dados em dois sistemas distintos.
- **Suporte a auditorias:** como o sistema envolve movimentações financeiras é importante que o sistema dê suporte futuras investigações sobre os valores que foram reembolsados. Desta forma, é de suma importância o estudo e implantação de um mecanismo que permita auditorias dentro do sistema.
- **Integração com serviços integrados do SIG@:** identificamos a necessidade de unificar os mecanismos de autenticação dos diversos sistemas implantados sob o domínio da UFRPE. A integração com os serviços integrados do Sig@ permitirá que haja apenas uma credencial de acesso para todos os sistemas da universidade, evitando assim uma explosão de credenciais por conta dos diversos serviços que podem ser ofertados.

Após a conclusão dos principais ajustes, o sistema deverá ser migrado do ambiente de desenvolvimento, no qual foi realizada a avaliação, para o ambiente de produção, onde será utilizado definitivamente para automatização do processo de gestão do auxílio saúde.

5.3 DIFICULDADES ENCONTRADAS

A maior dificuldade encontrada durante o desenvolvimento deste trabalho foi o desconhecimento total da tecnologia adotada para o desenvolvimento do sistema - *Drupal*. Após descobrir a obrigatoriedade da utilização do *Drupal* como tecnologia de desenvolvimento, descobrimos que seria necessário aprender a tecnologia para concretizar o SAS.

A curva de aprendizado inicial do *Drupal* não se estabilizou tão rapidamente, já que o desenvolvedor precisa passar por uma quebra de paradigma. Basicamente, é necessário passar a pensar o sistema em módulos e intercepções de eventos executados por estes módulos. Esta quebra dificulta um pouco a implementação até de funcionalidades mais básicas como um CRUD, por exemplo. Felizmente, após a estabilização da curva de aprendizado o desenvolvimento passou a fluir e as principais funcionalidades foram sendo concretizadas.

Outro fator que atrapalhou um pouco a construção de algumas funcionalidades foi a falta de controle total do ambiente de desenvolvimento. Sempre que um novo módulo de contribuidor precisava ser instalado ou alguma configuração modificada no servidor, era preciso aguardar um membro do NTI receber a solicitação e realizar a mudança.

O problema é que às vezes o módulo não era adequado e só descobríamos isso um dia ou mais, tempo que o módulo levava para ser instalado. Então, era preciso aguardar o novo módulo solicitado ser instalado para verificar se este satisfazia às necessidades do projeto. Isto acabou atrasando bastante o desenvolvimento. Principalmente, quando a necessidade de um novo módulo surgia no final de semana, quando o NTI não funcionava, apesar de ser este o período que tínhamos mais disponibilidade para desenvolver o projeto.

REFERÊNCIAS

- AMBLER, S. **Agile Modeling: Effective Practices for eXtreme Programming and the Unified Process**. [S.l.]: [s.n.], 2002.
- ANJOS, L. A. M. D.; MOURA, H. P. D. Um Modelo para Avaliação de Produtos de Software, Recife, 2009.
- BARBOSA, S. D. J.; SILVA, B. S. D. **Interação Humano-Computador**. Rio de Janeiro: Elsevier, 2010.
- BARKE, D. **Web Content Management: Systems, Features, and Best Practices**. 1st edition. ed. [S.l.]: [s.n.], 2016.
- BECK, K. Manifesto for agile software development., 2001.
- BLOG IPROCESS. Exemplo de processo em bpmn. **Blog Iprocess**, 2012. Disponível em: <http://blog.iprocess.com.br/wp-content/uploads/2012/11/blog_da_iprocess-exemplo_de_processo_em_bpmn.png>. Acesso em: 10 jan. 2016.
- DINIZ, V. A história do uso da tecnologia da informação na gestão pública brasileira através do CONIP – Congresso de Informática Pública. **X Congresso Internacional del CLAD sobre la Reforma del Estado y de la Administración Pública**, Santiago, Chile., p. 18-21, Oct. 2005.
- DYBÅ, T. A. T. D. Empirical studies of agile software development: A systematic review. **Information and software technology**, 2008. 833--859.
- ERIKSSON, H. E.; PENKER, M. Business modeling with UML. **New York**, 2000.
- ESPÍNDOLA, E. C. A importância do Modelagem de Objetos no Desenvolvimento de Sistemas. **Linha de código**, 2016. Disponível em: <http://www.linhadecodigo.com.br/artigos/img_artigos/evandro_camarini/modelagem_1.jpg>. Acesso em: 06 jan. 2017.
- GEORGAKOPOULOS, D.; HORNICK, M.; SHETH, A. An Overview of Workflow Management: From Process Modeling to Workflow Automation Infrastructure. **Distributed and Parallel Databases**, Boston, n. 3, p. 119-153, 1995.
- HIRSCH, M. Making RUP agile. **OOPSLA 2002 Practitioners Reports**, p. 1--ff, 2002.
- KRUCHTEN, P. **The Rational Unified Process: An Introduction**. 3rd. ed. [S.l.]: [s.n.], 2003.
- LERDORF, R.; TATROE, K. **Programming PHP**. 3ª. ed. [S.l.]: [s.n.], 2013.

- LIMA, G. A. F.; NETO, A. F. R. **Sistemas Institucionais Integrados da UFRN**. [S.l.]: Superintendência de Informática UFRN, 0000.
- MADEIRA, M. D. M. Casos de uso exemplo. **https://celodemelo.files.wordpress.com**, 2007. Disponível em: https://celodemelo.files.wordpress.com/2007/03/casos_de_uso_exemplo.JPG?w=550>. Acesso em: 15 jan. 2017.
- MICROSOFT. Diagramas de classe UML: referência. **Microsoft Developer Network**, 2015. Disponível em: <https://msdn.microsoft.com/pt-br/library/dd409437.aspx>>. Acesso em: 5 dez. 2016.
- MICROSOFT. Diagramas de classe UML. **MSDN Microsoft**, 2017. Disponível em: <https://msdn.microsoft.com/pt-br/library/dd409437.aspx>>. Acesso em: 05 jan. 2017.
- PERNAMBUCO, G. D. E. D. Delegacia pela internet. **Delegacia pela internet**, 2016. ISSN <http://servicos.sds.pe.gov.br/delegacia/>. Acesso em: 23 ago. 2016.
- PRESSMAN, R. S. **Engenharia de Software**. 6ª. ed. [S.l.]: McGrawHill, 2006.
- QUINTELLA, R. H.; JUNIOR, J. S. S. Sistemas de Apoio à Decisão e Descoberta de Conhecimento em Bases de Dados: Um Aplicação Potencial em Políticas Públicas. **o&s** , v. 10, p. 83-97, Setembro/Dezembro 2003. ISSN 28.
- RATIONAL SOFTWARE. Rational Unified Process Best Practices for Software Development Teams, nov. 2001. Disponível em: https://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251_bestpractices_TP026B.pdf>. Acesso em: 15 Novembro 2016.
- SANT'ANA, R. C. G. Tecnologia e gestão pública municipal. **Cultura Acadêmica**, São Paulo, 2009.
- SLIGE, M.; BRODERICK, S. **The Software Project Manager's Bridge to Agility**. 1st Edition. ed. [S.l.]: [s.n.], 2008.
- SOMMERVILLE, I. **Engenharia de Software**. 8ª. ed. [S.l.]: PEARSON EDUCATION, 2007.
- TOMLINSON, T.; VANDYK, J. K. **Pro.Drupal.7.Development**. 3rd edition. ed. [S.l.]: Apress, 2010.

UFPA, P. -. Sistema Auxílio Saúde. **SDGP - UFPA**, 2016. Disponível em: <<https://sdgp.ufpa.br/sgp/publico/login/loginServidorAS.action?acesso=servidorAS>>. Acesso em: 17 set. 2016.

UFRN. SIGRH - Sistema Integrado Gestão de Recursos Humanos, 2016. Disponível em: <<https://sigrh.ufrn.br/sigrh/public/home.jsf>>. Acesso em: 17 set. 2016.

W3TECHS. Usage statistics and market share of PHP for websites. **W3Techs**, 2016. Disponível em: <<https://w3techs.com/technologies/details/pl-php/all/all>>. Acesso em: 03 jan. 2016.

W3TECHS. Usage of content management systems for websites. **W3Techs**. Disponível em: <https://w3techs.com/technologies/overview/content_management/all>. Acesso em: 16 ago. 2016.

WESKE, M. **Business Process Management: Concepts, Languages, Architectures**. 2. ed. Potsdam, Germany: [s.n.], 2012.

WHITE, S. A. **BPMN modeling and reference guide: understanding and using BPMN**. [S.l.]: Future Strategies Inc., 2008.

WIEGERS, K.; BEATTY, J. **Software Requirements**. 3^a. ed. [S.l.]: Microsoft Press, 2013.

WIKIPÉDIA. IBM Rational Unified Process. **Wikipédia, a enciclopédia livre**, 2012. Disponível em: <https://upload.wikimedia.org/wikipedia/pt/0/07/Fases_do_RUP_-_portugues.jpg>. Acesso em: 12 jan. 2017.

APÊNDICE I – DOCUMENTO DE REQUISITOS

DOCUMENTO DE REQUISITOS - SISTEMA AUXÍLIO SAÚDE, SUGEP

Versão 1.0

Responsável

Márdeni Ferreira

1 INTRODUÇÃO

1.1 Objetivo do Documento

O objetivo principal deste documento é apresentar uma especificação inicial para um sistema que automatiza o processo manual de submissão e análise da documentação referente ao auxílio saúde, gerenciado pela SUGEP - órgão interno da UFRPE.

1.2 Glossário

Esta seção explica o conceito de alguns termos importantes que serão mencionados no decorrer deste documento.

Termo	Descrição
SAS	Sistema Auxílio Saúde. Sistema proposto como solução aos problemas da SUGEP.
Ano de exercício	Ano a que se refere o auxílio.
Mês de exercício	Mês a que se refere o auxílio. É entregue na folha do mês seguinte. Valores de meses de exercício anteriores não solicitados podem ser solicitados a qualquer momento, contanto que sejam de meses do atual ano de exercício.
Escopo	Parte da solução que se pretende entregar
Stakeholders	Partes envolvidas/interessadas no projeto
UFRPE	Universidade Federal Rural de Pernambuco
SUGEP	Superintendência de Gestão e Desenvolvimento de Pessoas, órgão interno da UFRPE responsável por gerir os recursos humanos da universidade.

1.3 Prioridades dos Requisitos

Ao estabelecer a prioridade dos requisitos foram adotadas as denominações “essencial”, “importante” e “desejável”. A prioridade dos requisitos é utilizada no gerenciamento do escopo das etapas do projeto e na definição das prioridades durante o desenvolvimento do sistema.

- **Essencial:** requisito sem o qual o sistema não entra em funcionamento. Requisitos essenciais são requisitos imprescindíveis, devem ser implementados desde as primeiras implantações do sistema.
- **Importante:** requisito sem o qual o sistema entra em funcionamento, mas de forma não satisfatória. Requisitos importantes devem ser implantados o mais rápido possível, mas se não forem parte do sistema ainda continuará funcional.
- **Desejável:** requisito que não compromete as funcionalidades básicas do sistema, isto é, o sistema pode funcionar de forma satisfatória sem ele. Requisitos desejáveis são requisitos que podem ser implantados por último, sem comprometer o funcionamento do sistema.

1.4 Visão Geral do Sistema

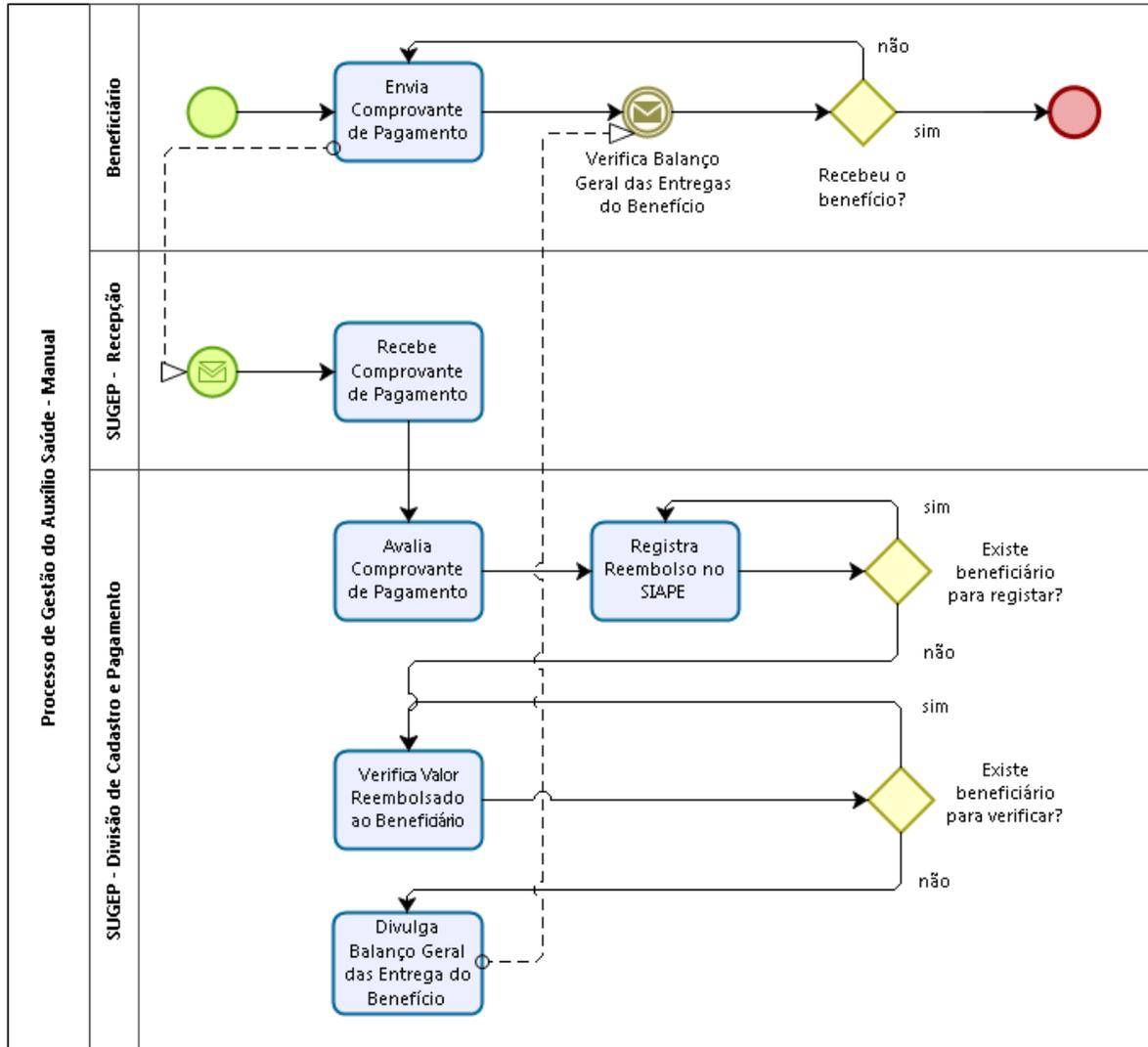
Esta seção apresenta o problema que deve ser solucionado e uma descrição macro da solução (o sistema a ser implantado).

1.4.1 Problema

A SUGEP gerencia a entrega do auxílio saúde aos servidores da universidade. Este auxílio cobre parte ou todo valor gasto pelo servidor com seu plano de saúde individual ou familiar e está fundamentada nas leis 8.112/1990 e 9.656/1998.

O diagrama da Figura 2 ilustra o processo de gestão do auxílio saúde manual, ou seja, sem o apoio de um sistema computacional customizado. Em seguida, o diagrama é explicado para que se possa entender bem o processo e os problemas desta abordagem.

Figura 20 - Ilustração do processo de gestão do auxílio saúde sem o apoio de um sistema específico.



Fonte: Elaborada pelo autor

Etapas/Atividades do processo de gestão do auxílio saúde manual:

- 8. Envia Comprovante de Pagamento:** Mensalmente, mais de mil beneficiários (as pessoas que fazem jus ao benefício) entregam aos servidores da SUGEP seus respectivos boletos e comprovantes de pagamento de planos de saúde, referentes

ao mês atual e/ou meses anteriores, além de um formulário padrão do auxílio preenchido manualmente, os quais são entregues pessoalmente na SUGEP.

- 9. Recebe Comprovante de Pagamento:** A recepção da SUGEP acolhe os comprovantes enviados e repassa para o setor de Divisão de Cadastro e Pagamento.
- 10. Avalia Comprovante de Pagamento:** Para cada documentação entregue, os gestores do auxílio saúde verificam se tudo está dentro dos padrões. Então, se tudo estiver correto, o beneficiário que entregou tal documentação é anotado numa planilha, para que posteriormente possa ser ativado, recebendo o benefício no mês seguinte.
- 11. Registra Reembolso no SIAPE:** Para cada beneficiário que entregou a documentação corretamente, o gestor do auxílio saúde acessa o sistema do Governo Federal referente à folha de pagamento (SIAPE) e informa se o beneficiário deve ser ativado no benefício, ou seja, deve receber o reembolso do valor gasto com o plano de saúde.
- 12. Verifica Valor Reembolsado ao Beneficiário:** Esta atividade acontece após a ativação dos beneficiários. Para cada beneficiário verifica-se o valor que o SIAPE creditou em sua folha de pagamento por conta do benefício. Estes valores são utilizados para geração do relatório mensal que precisa ser redigido pelo gestor do auxílio saúde.
- 13. Divulga Balanço Geral das Entregas do Benefício:** Enfim, depois de gerar o relatório, a SUGEP divulga o balanço geral de todos valores pagos aos beneficiários.
- 14. Verifica Balanço Geral das Entregas do Benefício:** Logo após o a divulgação do balanço geral, o beneficiário verifica se o benefício foi adicionado a sua folha de pagamento. Caso não, o beneficiário realiza o reenvio da documentação.

Esta abordagem gera, tanto aos gestores quanto aos beneficiários do auxílio saúde, uma série de problemas. Tais problemas são apresentados abaixo e estão agrupados de acordo com as etapas/atividades do processo.

- **Envia Comprovante de Pagamento:** O beneficiário precisa interromper suas atividades para se deslocar até o órgão, a fim de entregar a documentação. A UFRPE acaba sendo prejudicada quando o beneficiário é servidor ativo, pois o mesmo deixa de prestar os seus serviços à universidade mesmo que temporariamente.

Uma grande quantidade de pastas precisa ser mantida dentro da SUGEP para armazenar toda a documentação entregue. Além da restrição do espaço físico de trabalho que esta condição tem causado ao longo do tempo, sabe-se que os documentos físicos estão sujeitos a desgaste, catástrofes naturais/criminosas ou acidentes como incêndios, por exemplo.

- **Avalia Comprovante de Pagamento:** Existe certa limitação referente à busca de comprovantes para que seja feita a análise, justamente por causa da forma de armazenamento. Para a realização da análise ou criação de relatórios, os documentos precisam ser buscados manualmente em um sistema de arquivo físico, o que geralmente demanda um tempo considerável, principalmente quando se trata de centenas de documentos. Fatalmente, muitos erros acontecem por conta do “fator humano” e, eventualmente, alguns beneficiários não têm sua documentação avaliada já que a documentação está sujeita à perda.

Como se não bastasse, o beneficiário recebe tardiamente o *feedback* sobre o status do comprovante de pagamento submetido. Quando algum comprovante se perde, eventualmente o beneficiário não consegue reenviá-lo a tempo.

- **Registra Reembolso no SIAPE:** Aqui é feito um trabalho extremamente maçante e cansativo. É preciso navegar várias vezes entre as telas do sistema de folha de pagamento para informar, um a um, os beneficiários (mais de mil) que devem ser ativados no benefício. Esta atividade é uma das mais desgastantes, quando se trata de consumo de energia e tempo do pessoal da SUGEP.

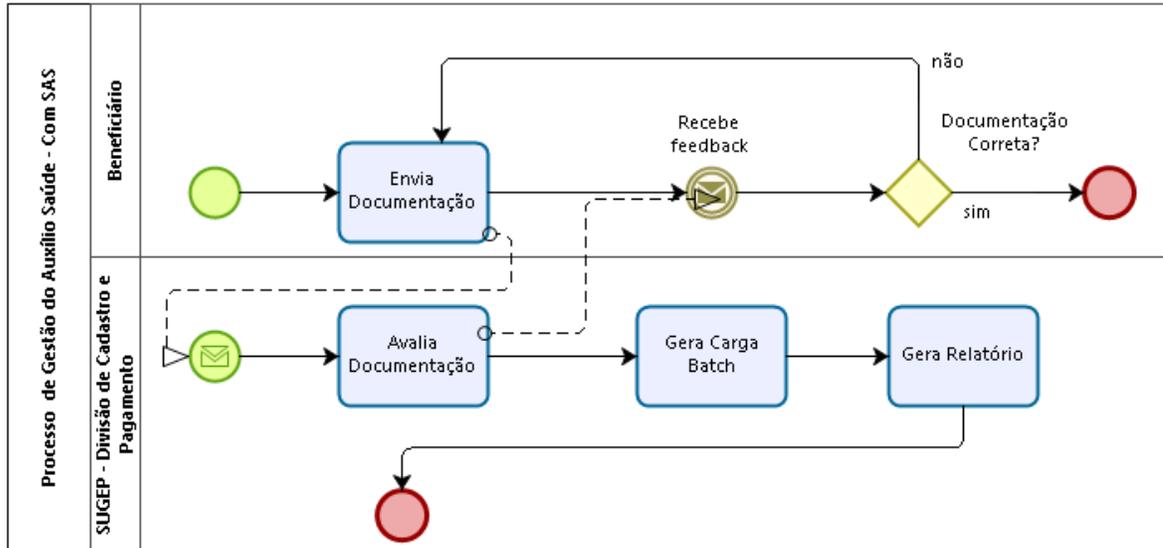
- **Verifica Valor Reembolsado ao Beneficiário:** Após a finalização da ativação dos beneficiários, o sistema do Governo Federal de folha de pagamento calcula o valor devido aos beneficiários por conta do auxílio saúde, de acordo com suas regras. Então, os gestores do auxílio saúde precisam navegar até a página de cada um dos beneficiários para recuperar esses valores. Em seguida, com bases nesses valores, geram um relatório contendo a relação de todos beneficiários e seus dependentes, juntamente com o valor pago a cada um dos beneficiários.

Esta atividade, juntamente com a atividade “Registra Reembolso no SIAPE”, compõe a parte mais desgastante de todo o processo. Juntas, estas atividades alocam aproximadamente três servidores e levam alguns dias para serem finalizadas. Estas duas atividades são responsáveis por tornar o processo de gestão do auxílio saúde extremamente mecânico e propenso a erros, além de comprometer as outras atribuições dos gestores do auxílio saúde.

1.4.2 O Sistema/Solução

O sistema proposto visa acabar com os problemas apresentados na seção anterior. O diagrama abaixo ilustra o processo de gestão do auxílio saúde após a implantação do sistema. Mais abaixo, segue a explicação do diagrama e a apresentação das vantagens advindas da nova abordagem para gestão do benefício.

Figura 21 Ilustração do processo de gestão do auxílio saúde automatizado pelo SAS.



Fonte: Elaborada pelo autor

Atividades do processo de gestão do auxílio saúde automatizado pelo SAS:

6. **Envia Documentação:** O beneficiário acessa o SAS e submete a documentação mensal necessária ao recebimento do benefício.
7. **Avalia Documentação:** O gestor do auxílio, acessando o sistema, verifica a documentação e dá um parecer, informando se a documentação apresenta, ou não, problemas. Caso presente, o gestor informa, através do sistema, qual o problema encontrado.
8. **Recebe *Feedback*:** O beneficiário recebe o parecer realizado pelo gestor do auxílio saúde informando se a documentação foi aceita ou apresenta algum problema. Caso haja algum problema, o beneficiário pode reenviar a documentação, obedecendo aos prazos legais de reenvio, a tempo de receber o benefício ainda no mês seguinte.

9. Gera Carga Batch: Ao final do mês, o gestor do auxílio, através do sistema, gera um arquivo no formato de *Carga Batch* (vide Apêndice I - Documento de Requisitos) referente ao mês corrente. Tal arquivo é transferido pelo gestor do auxílio para o Sistema de folha de pagamento do Governo Federal (SIAPE). Sendo este sistema responsável pelo acréscimo na folha de pagamento do valor devido ao servidor por conta do benefício.

10. Gera Relatório: Após gerar a Carga Batch, ativando todos beneficiários, o gestor do auxílio saúde pode acessar o sistema e gerar uma planilha contendo um relatório de todos beneficiários que enviaram a documentação e foram ativados naquele mês.

Esta nova abordagem confere, tanto aos gestores quanto aos beneficiários do auxílio saúde, uma série de vantagens e soluções frente aos problemas apresentados pelo processo manual de gestão do auxílio saúde. Estas vantagens são apresentadas abaixo e estão agrupadas de acordo com as etapas do novo processo.

- **Envia Documentação:**

Percebe-se que o beneficiário não precisa mais interromper suas atividades para deslocar-se pessoalmente até a SUGEP, podendo direcionar seus esforços para outras finalidades. No caso dos beneficiários que são servidores ativos, a UFRPE será favorecida diretamente já que os servidores em questão pertencem à universidade.

Os dados referentes aos comprovantes de pagamento do plano de saúde serão armazenados digitalmente, dispensando o armazenamento de documentos físicos nas instalações do órgão e impedindo a perda em casos de situações catastróficas, já que os servidores de armazenamento cuidarão da manutenção de *backups* dos documentos digitais.

- **Avalia Documentação:**

O sistema permite que os comprovantes possam ser facilmente recuperados e avaliados, quando necessário, através dos mecanismos de busca e filtros que estão disponíveis no sistema – por exemplo, busca por: todos comprovantes referentes ao mês atual que ainda não foram avaliados; comprovante de servidor que possui determinado CPF, entre outras. Esta abordagem diminui as chances de documentos ficarem de fora da avaliação, o que poderia impedir o beneficiário de receber o benefício no mês seguinte (comum na abordagem antiga - sem o sistema).

Além disso, o sistema pode fazer verificações prévias, no momento do envio da documentação para que o gestor não seja sobrecarregado. Por exemplo: o sistema pode verificar se o comprovante de pagamento enviado para determinado mês, já foi enviado e, portanto, barrar o envio e avisar ao beneficiário.

A avaliação permite que o gestor dê um *feedback* ao beneficiário. Este recebe um e-mail informando se a documentação contém algum erro ou está dentro do conformes. Vale ressaltar que o gestor não é sobrecarregado, uma vez que o sistema dispõe de justificativas pré-definidas, não sendo necessário ao gestor a redação de um texto para cada avaliação.

- **Gera Carga *Batch* e Gera Relatório:**

Após a avaliação de toda documentação, um intenso trabalho manual, que era realizado com a abordagem manual dá lugar a apenas alguns cliques dentro do sistema. O gestor do auxílio passa a navegar até a seção de carga Batch e Relatórios dentro do sistema. Então, o sistema recupera todos os dados referentes aos beneficiários e suas submissões de documentação. Com estes dados o sistema gera um arquivo no formato de *Carga Batch* ou o Relatório Mensal. Então, o gestor do auxílio simplesmente faz a transferência do arquivo referente a carga Batch para o Sistema de folha de pagamento do Governo Federal (SIAPE) e todos servidores que tiveram a documentação aprovada são ativados de uma só vez. Desta forma, os

beneficiários recebem o valor referente ao benefício na próxima folha de pagamento.

Na abordagem antiga estes poucos cliques no sistema e importação de apenas um arquivo era uma das atividades mais desgastante do processo. O gestor do auxílio saúde precisava navegar entre várias telas do SIAPE. Para cada beneficiário o gestor navegava entre algumas telas para informar que aquele devia receber o benefício. Estas tarefas juntas duravam alguns dias para serem executadas e precisavam ser divididas entre aproximadamente três servidores para que a mesma fosse finalizada em tempo hábil. Além disto, eram atividades extremamente mecânicas e suscetíveis a erros. Felizmente, com a implantação do sistema, o esforço foi reduzido drasticamente, passando a alocar apenas um servidor e sendo executada em aproximadamente 5 minutos.

2. STAKEHOLDERS IDENTIFICADOS

Esta seção lista os principais *stakeholders* - parte afetada/interessada pelo sistema - por ordem de prioridade, os mais importantes aparecem primeiro. O critério de priorização é apresentado na Seção 2.1.

Gestor do Auxílio Saúde	Funcionário responsável por examinar a documentação e entregar o benefício aos beneficiários do auxílio saúde. Principal usuário/beneficiado do/pelo sistema. Sua carga de trabalho manual decairá drasticamente a partir da automatização do processo.
Beneficiário do Auxílio Saúde	Pessoa que faz jus ao auxílio saúde. Eventual usuário do sistema. Será eliminada

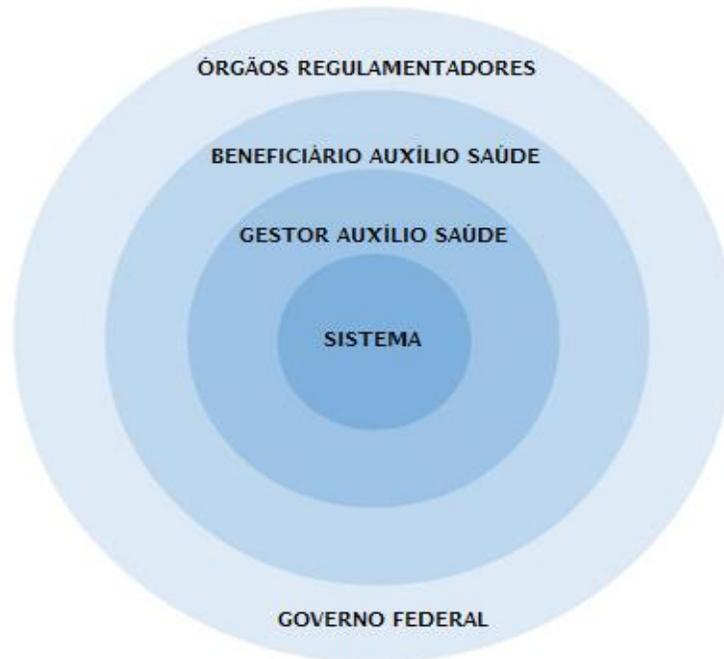
	a necessidade de seu deslocamento até a SUGEP para entrega da documentação mensal exigida para o recebimento do benefício. Podem ser servidores da UFRPE (ativos ou aposentados) ou pensionistas destes servidores.
Governo Federal	A entidade não utiliza este sistema (SAS). Mas alguns dados gerados precisam ser importados para o sistema de folha de pagamento do Governo Federal (SIAPE), respeitando o padrão imposto por ele.
Órgãos Regulamentadores	Há duas leis que fundamentam o benefício: 8.112/1990 e 9.656/1998. O sistema deve se adequar a estas leis. Esta legislação precisa ser consultada antes da implementação do sistema.

2.1. Modelo Cebola

O diagrama de cebola,

Figura 22, ilustra o nível de interação dos *stakeholders* com o sistema. Os *stakeholders* mais próximos da camada central possui uma relação mais forte com o sistema, sendo a principal fonte de requisitos, principais usuários e beneficiados pela implantação do sistema.

Figura 22 Modelo Cebola.



Fonte: O autor.

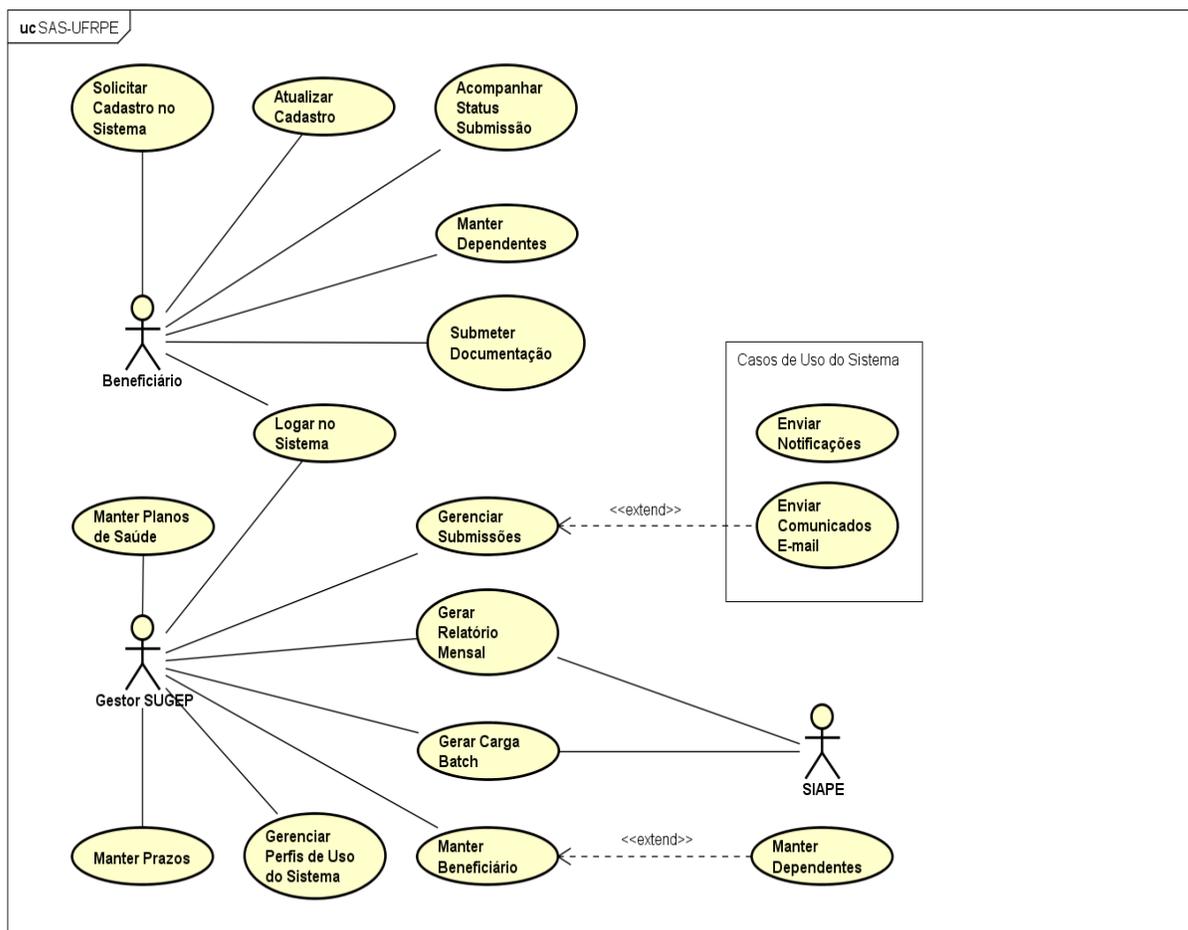
3. REQUISITOS FUNCIONAIS

Esta seção apresenta todos os casos de uso levantados para o sistema. Estes casos de uso podem ser entendidos como requisitos funcionais ou, simplesmente, funcionalidades do sistema.

3.1. Diagrama de Casos de Uso

Esta seção apresenta os requisitos funcionais na forma de um diagrama de casos de uso apresentado na **Figura 23**.

Figura 23 Diagrama de casos de uso.



powered by Astah

Fonte: O autor.

3.2. Descrição dos Casos de Uso

Abaixo são apontados os requisitos funcionais elencados para o sistema, acompanhados de sua descrição e prioridade. Os requisitos estão agrupados da seguinte forma:

- **Submissão:** casos de uso associados a envio e avaliação de comprovantes de pagamento referentes a planos de saúde;
- **Importação/Exportação:** casos de uso associados a troca de informação entre este sistema e o SIAPE;
- **Cadastro:** casos de uso associados a operações de inserção, leitura, modificação e exclusão de entidades dentro do sistema;
- **Autorização e Acesso:** casos de uso associados à concessão de acesso ao sistema e criação de permissões;
- **Comunicação:** casos de uso associados à troca de mensagens entre o usuário e o sistema.

3.2.1 Submissão

RF01 - Submeter Documentação	Prioridade: Essencial	Ator: Beneficiário
-------------------------------------	------------------------------	---------------------------

Pré-condição: Ter seção ativa no sistema.

Responsável por permitir que o servidor que já aderiu ao benefício, envie o comprovante de pagamento referente ao seu plano de saúde para o mês vigente e/ou meses anteriores, dentro do ano de exercício.

RF02 – Acompanhar Status da Submissão	Prioridade: Importante
--	-------------------------------

Responsável por permitir que o beneficiário tenha acesso a todas suas submissões. O beneficiário pode ser o status da documentação aprovada ou reprovada - de forma que possa corrigi-las antes do encerramento do prazo de envio.

RF03 - Gerenciar Submissões	Prioridade: Essencial
------------------------------------	------------------------------

Responsável por permitir que o gestor do benefício visualize as submissões de todos beneficiários, avalie e informe quais os problemas identificados para que o beneficiário possa enviá-los dentro do prazo, de forma que o recebimento do benefício ocorra no mês de exercício seguinte ao da submissão. Após avaliação da documentação um e-mail deve ser enviado pelo sistema ao beneficiário.

3.2.1 Importação/Exportação

RF04 – Gerar Carga <i>Batch</i>	Prioridade: Importante
--	-------------------------------

Responsável por permitir que o gestor do auxílio gere um arquivo de Carga Batch. Este arquivo é então exportado para o sistema de folha de pagamento do governo federal (SIAPE), a fim de iniciar a ativação de uma só vez de todos beneficiários que enviaram a documentação correta.

Este arquivo deve seguir o padrão definido pelo manual de Carga Batch. Mais informações sobre a integração de sistemas de gestão com o SIAPE através de arquivos *batch* podem ser encontradas neste endereço: <http://www.tesouro.fazenda.gov.br/arquivos-batch>.

RF05 – Gerar Relatório Mensal	Prioridade: Importante
--------------------------------------	-------------------------------

Responsável por permitir que o gestor do auxílio gere uma relação contendo os beneficiários e respectivos dependentes, bem como o valor pago ao beneficiário por conta do auxílio num dado mês. O sistema deve construir este relatório, a partir de um arquivo *batch* gerado pelo SIAPE.

3.3. Cadastro

RF06 – Atualizar Cadastro	Prioridade: Importante
----------------------------------	-------------------------------

Responsável por permitir que o beneficiário atualize suas informações pessoais.

RF07 - Manter Gestores do Auxílio	Prioridade: Importante
--	-------------------------------

Responsável por permitir que o administrador do sistema crie, modifique, consulte ou desative, no sistema, o cadastro de um gestor do auxílio saúde.

RF08 - Manter Beneficiário**Prioridade: Importante**

Responsável por permitir que o gestor do auxílio saúde crie, modifique, consulte ou desative, no sistema, o cadastro de um beneficiário.

RF09 - Manter Dependentes**Prioridade: Importante**

Responsável por permitir que o gestor do auxílio saúde e os beneficiários criem e desativem dependentes associados a um beneficiário. Os gestores podem modificar qualquer dependente, já os beneficiários pode manter apenas os seus respectivos dependentes.

RF10 - Manter Prazos**Prioridade: Importante**

Responsável por permitir que o gestor do auxílio saúde modifique os prazos para envio dos comprovantes de pagamento. Além disso, o sistema usará esses prazos para notificar o cliente no próprio sistema através do **RF15**.

RF11 - Manter Planos de Saúde**Prioridade: Importante**

Responsável por permitir que o gestor do auxílio saúde adicione, exclua e modifique os planos de saúde no sistema. Depois de registrados, estes planos de saúde podem ser associados ao cadastro dos beneficiários.

3.4. Autorização e Acesso**RF12 – Solicitar Cadastro no Sistema****Prioridade: Essencial**

Responsável por permitir que o beneficiário realize o cadastro no sistema. O beneficiário precisará ser homologado pelo gestor do auxílio saúde. Após estes procedimentos, o beneficiário poderá acessar o sistema através do **RF13**.

RF13 - Logar no Sistema**Prioridade: Essencial**

Responsável por permitir que o usuário do sistema receba autorização para acessar os recursos do sistema. Poderá acessar o sistema: administrador do sistema, gestor do auxílio saúde e servidor beneficiário.

RF14 – Gerenciar Perfis de Uso	Prioridade: Importante
---------------------------------------	-------------------------------

Responsável por permitir que os usuários com determinado perfil de uso acessem apenas os recursos que lhes são autorizados. Neste caso temos dois perfis de uso:

- Administrador: o autor gestor do auxílio saúde;
- Usuário comum: o autor beneficiário;

Cada autor tem acesso apenas as funcionalidades com as quais interage. Esta interação pode ser conferida no diagrama de casos de uso apresentado pela **Figura 23**. Estes diferentes perfis de uso podem ser concedidos ou revogados a qualquer usuário do sistema a qualquer momento.

3.5. Comunicação

RF15 – Enviar Notificação	Prioridade: Desejável
----------------------------------	------------------------------

Responsável por notificar o beneficiário sobre o prazo de envio da documentação para recebimento do auxílio naquele mês. Os prazos cadastrados no sistema devem ser recuperados e uma mensagem avisando sobre tal prazo deve ser exibida ao longo de todo sistema.

RF16 - Envio de Comunicados E-mail	Prioridade: Desejável
---	------------------------------

Responsável pelo envio de e-mail ao beneficiário no momento no qual o gestor do auxílio saúde aprova ou reprovava os comprovantes enviados. Desta forma, o beneficiário pode ser tranquilizado ou tomar as medidas cabíveis para reenviar a documentação correta.

4. REQUISITOS NÃO FUNCIONAIS

Esta seção agrupa os requisitos não funcionais associados a Requisitos do produto (Usabilidade; Desempenho; Segurança; Distribuição; Portabilidade).

RNF01 - Disponibilidade

O sistema deve estar disponível durante o horário de expediente dos técnico-administrativos da UFRPE, de segunda a sexta-feira. E durante os prazos legais para reenvio de documentação que apresentou algum problema.

RNF02 - Usabilidade

O sistema deve apresentar uma interface gráfica leve (poucos elementos), intuitiva e de fácil memorização da localização das funcionalidades. Como a documentação de centenas de beneficiários precisa ser analisada todo mês, para que o trabalho não se torne exaustivo, as tarefas dentro do sistema precisam ser concluídas com poucos cliques e transições de tela.

RNF03 - Segurança

Os dados referentes aos beneficiários, como valor pago no plano de saúde, número de conta bancária, entre outras informações pessoais, devem ser invioláveis, acessíveis apenas pelo pessoal autorizado da SUGEP.

RNF04 – Desempenho

O tempo de resposta do sistema deve pode ser flexível para as operações realizadas pelos beneficiários, que usam o sistema eventualmente. No caso das operações realizadas pelos gestores do auxílio, o sistema deve responder em menos de 5 segundos para que as atividades que já demandam um tempo considerável, não se tornem impraticáveis.

5. ESCOPO NÃO CONTEMPLADO

- **Entrada no Benefício:** Responsável por permitir que servidores deem entrada/acompanhe o processo inicial que o habilita a receber o auxílio saúde.
 - **Motivo:** O cliente optou por não automatizar esta parte do processo por questões internas à organização.

- **Requisição de Valores do Exercício Anterior:** Responsável por permitir que os beneficiários do auxílio saúde solicitem o benefício não solicitado no ano de exercício anterior.
 - **Motivo:** O cliente informou que o recebimento desse valor ocorre por meio de um processo independente e que fugiria do propósito deste sistema.

APÊNDICE II – TELAS

Abaixo são apresentadas as telas criadas para o Sistema Auxílio Saúde.

Figura 24 Tela de *login*.



Auxílio Saúde SUGEP/UFRPE

INÍCIO

Login do usuário

Usuário *

Senha *

[Criar nova conta](#)
[Recuperar senha](#)

ENTRAR

Procedimento de Inclusão

Enviado por auxiliosauade em qua, 23/11/2016 - 09:19

1- A solicitação de inclusão, alteração ou exclusão do Auxílio Saúde poderá ser feita a qualquer tempo e deverá ser efetuada por meio de requerimento próprio e em seguida formalizado processo no Setor de Comunicação desta Universidade.

[Leia mais](#)

Atenção - Prazo de Envio de Comprovante

Prezado Servidor/Pensionista,

V.Sa. tem até o 6/12 para enviar a documentação comprovando o gasto com o plano de saúde, com data de vencimento referente ao mês de novembro.

Figura 25 Tela inicial - visão do beneficiário do auxílio saúde.



The screenshot shows the user interface for the 'Auxílio Saúde SUGEP/UFRPE' system. At the top center is the logo of the UFRPE (Universidade Federal de Pernambuco), featuring a shield with a star and the motto 'VIVAMUS SEMPER'. Below the logo, the title 'Auxílio Saúde SUGEP/UFRPE' is displayed in a bold, black font. A horizontal navigation bar contains several menu items: 'INÍCIO' (highlighted in red), 'MEU CADASTRO', 'ADICIONAR DEPENDENTE', 'MEUS DEPENDENTES', 'ENVIAR COMPROVANTE', and 'MEUS COMPROVANTES'. The main content area is divided into two columns. The left column features a section titled 'Procedimento de Inclusão' with a sub-header 'Enviado por auxiliossaude em qua, 23/11/2016 - 09:19'. Below this, there is a paragraph of text explaining the inclusion process and a 'Leia mais' link. The right column has a section titled 'Atenção - Prazo de Envio de Comprovante' with a sub-header 'Prezado Servidor/Pensionista,' and a paragraph of text regarding the deadline for submitting documentation. At the bottom of the page, there is a section titled 'Prazo de Envio da Documentação Mensal' and a small upward-pointing arrow icon.

Figura 26 Tela inicial – visão do gestor do auxílio saúde.



The screenshot shows the user interface for the 'Auxílio Saúde SUGEP/UFRPE' system from the manager's perspective. At the top center is the logo of the UFRPE (Universidade Federal de Pernambuco), featuring a shield with a star and the motto 'VIVAMUS SEMPER'. Below the logo, the title 'Auxílio Saúde SUGEP/UFRPE' is displayed in a bold, black font. A horizontal navigation bar contains several menu items: 'INÍCIO' (highlighted in red), 'MEU CADASTRO', 'GERENCIAR BENEFICIÁRIOS', 'ADICIONAR DEPENDENTE', 'MEUS DEPENDENTES', 'TODOS DEPENDENTES', 'ENVIAR COMPROVANTE', 'MEUS COMPROVANTES', 'AVALIAÇÃO DE COMPROVANTES', 'GERENCIAR PRAZOS', 'CARGAS BATCH', 'GERAR CARGA BATCH', 'RELATÓRIOS', and 'PLANOS DE SAÚDE'. The main content area is divided into two columns. The left column features a section titled 'Procedimento de Inclusão' with a sub-header 'Enviado por auxiliossaude em qua, 23/11/2016 - 09:19'. Below this, there is a paragraph of text explaining the inclusion process and a 'Leia mais' link. The right column has a section titled 'Atenção - Prazo de Envio de Comprovante' with a sub-header 'Prezado Servidor/Pensionista,' and a paragraph of text regarding the deadline for submitting documentation. At the bottom of the page, there is a small upward-pointing arrow icon.

Figura 27 Tela de cadastro de beneficiário do auxílio saúde - parte 1.

Cadastro de Beneficiário Titular

Permite o cadastro/atualização dos dados pessoais do Beneficiário Titular.

Ativo

Categoria de Beneficiário *

Ativo
 Aposentado
 Pensionista

Nome *

Márdeni Ferreira

CPF *

0980123

Data de Nascimento *

Date

01/10/2013
E.g., 13/12/2016

Data da Inclusão

Date

10/2016
E.g., 12/2016

E-mail *

mardeni.ferreira@gmail.com

Figura 28 Tela de cadastro de beneficiário do auxílio saúde - parte 2.

Matrícula *

12356

Ocultar pesos das linhas

Processo	Ordem
6983456	0
0567	1
	2

ADICIONAR OUTRO ITEM

Telefone

81 985981238

Plano de Saúde *

ALIANÇA UFRPE

Convênio *

Nenhum

Matrícula Instituidor (Apenas para Pensionista)

CPF Instituidor (Apenas para Pensionista)

Figura 29 Tela de cadastro de beneficiário do auxílio saúde - parte 3.

Convênio *

Nenhum

Matrícula Instituidor (Apenas para Pensionista)

CPF Instituidor (Apenas para Pensionista)

Valor do Plano

SALVAR

Figura 30 Tela para gerenciamento de todos beneficiários cadastrados no sistema.

Gerenciar Beneficiários

Matrícula CPF Nome APPLY

Tipo de Beneficiário	CPF	Matrícula	Nome	Data de Nascimento	Email	Visualização	Remoção	Dependente
Ativo	0980123	12356	Márdeni Ferreira	01/10/2013	mardeni.ferreira@gmail.com	ver	excluir	dependente
Aposentado	37233289468	383519	MARIA DE FATIMA DA SILVA	02/12/2016	MARIA.FATIMA.2015@BOL.COM.BR	ver	excluir	dependente
Ativo	43991386453	385057	maria leda da silva coelho	24/11/1965	leda.coelho@gmail.com	ver	excluir	dependente
Pensionista	23155442468	3074404	NERISE DAS GRAÇAS CAVALCANTI S.	27/02/1955	Nerisesales@hotmail.com	ver	excluir	dependente
Ativo	01054802408	1870275	BARTIRA WANDERLEY ALVES LEITE	13/03/1981	bartiraw@yahoo.com.br	ver	excluir	dependente
Ativo	03806432430	1148509	HAROLDO SABINO DO NASCIMENTO JUNIOR	15/09/1981	haroldosabino@ufrpe.br	ver	excluir	dependente
Ativo	04354592444	1754248	ANDREA KALINKA DE FREITAS MENDONCA	15/05/1982	deamendonc@gmail.com	ver	excluir	dependente
Ativo	05979799443	1612261	JULIANA MARÇAL	25/06/1986	JULIMARCAL25@GMAIL.COM	ver	excluir	dependente
Ativo	08444325473	1903798	THIAGO HENRIQUE ANDRADE DE LUCENA	21/09/1991	thiagope21@hotmail.com	ver	excluir	dependente
Ativo	14844613472	0383125	Elias Manoel de Lima	16/05/1956	eliaslimam3@gmail.com	ver	excluir	dependente
Ativo	43194451472	1038823	MÔNICA MARIA CESAR FONSECA DA CRUZ	18/02/1965	mmcesar@gmail.com	ver	excluir	dependente

Figura 31 Tela Adicionar Dependente – parte 1.

Dependente

Indivíduo que vive sob a dependência do Servidor Ativo ou Aposentado ⚙

Ativo

Nome *

Data de Nascimento *

Date

E.g., 19/12/2016

Data da Inclusão

Date

E.g., 12/2016

Ocultar pesos das linhas

Processo **Ordem**

0 ⬆

Figura 32 Tela Adicionar Dependente - parte 2.

Ocultar pesos das linhas

Processo **Ordem**

0 ▼

ADICIONAR OUTRO ITEM

Valor do Plano

Vínculo

- Nenhum - ▼

SALVAR

Figura 33 Tela para gerenciamento dos dependentes associados à pessoa que está acessando o sistema.

Início / Meus Dependentes / Meus Dependentes

Meus Dependentes ⚙️

Nome	Data de Nascimento	Modificação	Vínculo
Rosa Bione	11/10/2016	modificar	FILHO(A) ESP
Pedrinho José	11/10/2016	modificar	ENTEADO(A) ATÉ 24 ANOS, SE ESTUDANTE
Flora Maria	11/10/2016	modificar	FILHO(A) ESP

Figura 34 Tela para gerenciamento de todos dependentes cadastrados no sistema.

Todos Dependentes ⚙

Visualização	Modificação	Nome	Data de Nascimento
ver	modificar	Joaquim	domingo, 9 Outubro, 2016
ver	modificar	Rosa Bione	terça-feira, 11 Outubro, 2016
ver	modificar	Pedrinho José	terça-feira, 11 Outubro, 2016
ver	modificar	Flora Maria	terça-feira, 11 Outubro, 2016
ver	modificar	yuri silva coelho	segunda-feira, 31 Outubro, 1994
ver	modificar	João Vitor	quinta-feira, 15 Dezembro, 2016
ver	modificar	yuri silva coelho	segunda-feira, 31 Outubro, 1994
ver	modificar	caue silva coelho	sábado, 4 Janeiro, 2003
ver	modificar	Maria de Fátima Neves de Lima	sexta-feira, 25 Novembro, 1955

1 2 próximo » fim »

Figura 35 Tela de envio de comprovante de pagamento.

Comprovante de Pagamento

Submissão da documentação mensal referente ao auxílio saúde.

Comprovante de Pagamento *

Nenhum arquivo selecionado.

Arquivos devem ter menos que **2 MB**.
 Tipos de arquivos permitidos: **pdf jpeg jpg png mp4**.

Valor do Pagamento R\$

Valor gasto com o plano de saúde.

Data de Vencimento *

Date

E.g., 20/12/2016

Figura 36 Tela para acompanhamento dos comprovantes enviados.

Início / Meus Comprovantes / Meus Comprovantes

Meus Comprovantes

Visualização	Comprovante de Pagamento	Status	Data de Envio	Data de Vencimento	Remoção	Modificação
ver	 HP_Deskjet_2000(1).jpg	Não avaliado	20/12/2016 - 23:02	22/12/2016	apagar	modificar
ver	 HP_Deskjet_2000.jpeg	Aprovado	20/12/2016 - 23:04	21/12/2016	apagar	modificar

Figura 37 Tela para gerenciamento/avaliação dos comprovantes enviados – parte 1.

Avaliação de Comprovantes

Data de Envio: Vigência: Ordenar Por: Ordem:

ID do Usuário	Visualização	Comprovante de Pagamento	Data de Envio	Data de Modificação	Status	Data de Vencimento	Avaliação	Remoção
57	ver	 cassi-jeisa-domingues-nov16.pdf	07-12-2016 01:59:47	07-12-2016 09:31:11	Não avaliado	30/11/2016	avaliar	excluir
28	ver	 Protocolo Auxilio Saude 11.16.jpg	06-12-2016 18:48:06	06-12-2016 19:33:07	Aprovado	08/11/2016	avaliar	excluir
39	ver	 HAPVIDA-CIDA-NOV.pdf	06-12-2016 16:15:08	06-12-2016 16:18:37	Aprovado	10/11/2016	avaliar	excluir
38	ver	 comprovantemarcelo.jpg	06-12-2016 16:15:08	06-12-2016 16:28:52	Aprovado	30/11/2016	avaliar	excluir
36	ver	 Comprovante Unimed Novembro (paguei atrasado - vence dia 25-11) JHMD.pdf	06-12-2016 16:01:21	06-12-2016 16:03:47	Aprovado	25/11/2016	avaliar	excluir
38	ver	 comprovanteclaudia.PDF	06-12-2016 15:21:45	06-12-2016 15:21:45	Não avaliado	25/11/2016	avaliar	excluir
32	ver	 Comprovante (2).pdf	06-12-2016 14:46:56	06-12-2016 14:50:58	Aprovado	25/11/2016	avaliar	excluir
31	ver	 PLANOSAUDE3.pdf	06-12-2016 14:33:33	06-12-2016 15:09:58	Aprovado	10/11/2016	avaliar	excluir

Figura 38 Tela para gerenciamento/avaliação dos comprovantes enviados - parte 2.

Comprovante de Pagamento

[Ver](#) [Editar](#) [Excluir](#)

Submitted by **MARIA.FERNANDES** on ter, 06/12/2016 - 16:15
Submissão da documentação mensal referente ao auxílio saúde.

Comprovante de Pagamento *

 **HAPVIDA-CIDA-NOV.pdf** (327.31 KB) [REMOVED](#)

Incluir arquivo na visualização

Valor do Pagamento * R\$

Valor gasto com o plano de saúde.

Valor Devido R\$

Valor que deve ser ressarcido ao Beneficiário.

Diferença R\$

Diferença entre o valor gasto pelo beneficiário com o plano de saúde e o valor do auxílio.

Figura 39 Tela para gerenciamento/avaliação dos comprovantes enviados - parte 3.

Comprovante de Pagamento

[Ver](#) [Editar](#) [Excluir](#)

Submitted by **MARIA.FERNANDES** on ter, 06/12/2016 - 16:15
Submissão da documentação mensal referente ao auxílio saúde.

Comprovante de Pagamento *

 **HAPVIDA-CIDA-NOV.pdf** (327.31 KB) [REMOVED](#)

Incluir arquivo na visualização

Valor do Pagamento * R\$

Valor gasto com o plano de saúde.

Valor Devido R\$

Valor que deve ser ressarcido ao Beneficiário.

Diferença R\$

Diferença entre o valor gasto pelo beneficiário com o plano de saúde e o valor do auxílio.

Figura 40 Tela para gerenciamento/avaliação dos comprovantes enviados - parte 4.

Status *

Não avaliado

Aprovado

Reprovado

Justificativa

Data do boleto não corresponde à definida na submissão.

Data do boleto não corresponde à do comprovante.

Comprovante/Boleto ilegíveis.

Observação

Data de Vencimento *

Date

10/11/2016

E.g., 20/12/2016

Vigência

Mês Vigente

Figura 41 Tela para gerenciamento dos prazos de envio de comprovante.

Prazo de Envio 

Define a data limite para o envio da documentação mensal referente ao auxílio saúde.

Mês *

Junho 

Prazo *

Date

09/02/2017

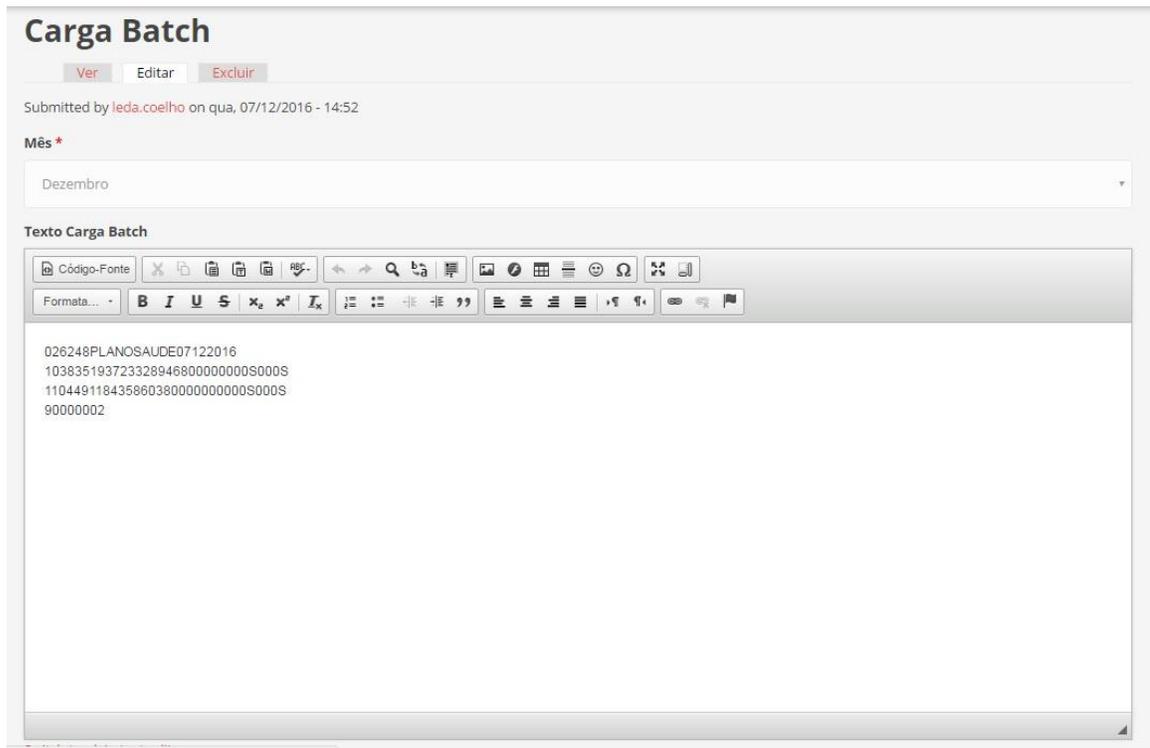
E.g., 09/02/2017

SALVAR

Gerenciar Prazos

Mês	Prazo	Visualização	Remoção	Modificação
Novembro	09/11/2016	ver	excluir	modificar
Dezembro	06/12/2016	ver	excluir	modificar

Figura 42 Tela com a carga batch gerada num editor WYSIWYG.



The screenshot shows a web-based WYSIWYG editor interface. At the top, the title "Carga Batch" is displayed. Below the title are three buttons: "Ver", "Editar", and "Excluir". A status message indicates the document was submitted by "leda.coelho" on "qua, 07/12/2016 - 14:52". A dropdown menu labeled "Mês" is currently set to "Dezembro". The main editing area is titled "Texto Carga Batch" and contains a rich text editor toolbar with various icons for text formatting, alignment, and insertion. The text area below the toolbar contains the following content:

```
026248PLANOSAUDE07122016  
103835193723328946800000000S000S  
11044911843586038000000000S000S  
90000002
```