

UNIVERSIDADE FEDERAL DO PARÁ  
INSTITUTO DE CIÊNCIAS EXATAS E NATURAIS  
FACULDADE DE COMPUTAÇÃO  
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO

Eliwelton Gomes Paes  
David Barros de Oliveira Junior

**DESENVOLVIMENTO DE SOFTWARE – ESTUDO DE CASO DO SISTEMA APAC  
BARROS BARRETO**

Belém  
2018

Eliwelton Gomes Paes  
David Barros de Oliveira Junior

**DESENVOLVIMENTO DE SOFTWARE – ESTUDO DE CASO DO SISTEMA APAC  
BARROS BARRETO**

Trabalho de Conclusão de Curso apresentado para obtenção do título de Bacharel em Sistemas de Informação pela Faculdade de Computação do Instituto de Ciências Exatas e Naturais da Universidade Federal do Pará.

Orientador: Prof. Dr. Alfredo Braga Furtado.

Belém  
2018

Paes, Eliwelton Gomes; Junior, David Barros de Oliveira

Desenvolvimento de Software – Estudo de Caso do Sistema APAC  
Barros Barreto / (Eliwelton Gomes Paes e David Barros de Oliveira Junior);  
orientador, Alfredo Braga Furtado – 2018.

139f. il. 28cm.

Trabalho de Conclusão de Curso (Graduação) – Universidade Federal  
do Pará. Instituto de Ciências Exatas e Naturais. Faculdade de  
Computação. Belém, 2018.

1. WebApp. 2. APAC. 3. HUJBB.

CDD \_\_\_\_ . ed. \_\_\_\_\_

Eliwelton Gomes Paes  
David Barros de Oliveira Junior

**DESENVOLVIMENTO DE SOFTWARE – ESTUDO DE CASO DO SISTEMA APAC  
BARROS BARRETO**

Trabalho de Conclusão de Curso apresentado para obtenção do título de Bacharel em Sistemas de Informação pela Faculdade de Computação do Instituto de Ciências Exatas e Naturais da Universidade Federal do Pará.

Data da aprovação: Belém-PA. \_\_\_\_/\_\_\_\_/\_\_\_\_

Conceito:

Banca Examinadora

---

Prof. Dr. Alfredo Braga Furtado  
Faculdade de Computação/ICEN/UFPA – Orientador

---

Prof. Dr. Raimundo Viégas Júnior  
Faculdade de Computação/ICEN/UFPA – Membro

---

Prof. M. Sc. José Maria Nascimento Bitar  
Faculdade de Computação/ICEN/UFPA – Membro

## **AGRADECIMENTOS**

Primeiramente a Deus por ter me dado saúde e força para superar todas as dificuldades.

À minha família, que sempre me deu apoio, carinho e amor em tudo o que eu faço especialmente minha mãe, Marta de Jesus dos Santos Gomes, minha heroína, responsável por tudo o que sou hoje.

A minha namorada e companheira, Mariza da Silva Borges, o melhor presente que a vida poderia ter me dado. Obrigado por tudo o que você representa na minha vida, por todas as alegrias, por todo o carinho, por todos os momentos e principalmente por todo o amor. Sem você, esta conquista não seria a mesma.

Ao meu orientador, Professor Alfredo Braga Furtado, pela paciência, compreensão e pelo empenho dedicado em nos orientar durante a elaboração deste trabalho.

Ao meu grande amigo, David Barros de Oliveira Junior, que participou comigo no desenvolvimento desse trabalho.

Aos meus grandes amigos Robson, Hevelyn, Arthur, Clelson e Luis, verdadeiros irmãos na amizade. Vocês fazem um grande diferencial na minha vida.

Aos demais amigos que estiveram ao meu lado nessa caminhada. O caminho ainda está no início, mas sem amizades verdadeiras, eu jamais teria chego até aqui.

Eliwelton Gomes Paes

## **AGRADECIMENTOS**

A Deus, pela dádiva da vida e por todas as portas que ele abriu e abrirá na minha vida e por nunca me abandonar nos bons e maus momentos.

À minha mãe, Maria Idalina Freire Carvalho, pelo amor, apoio, compreensão e carinho, essenciais nos momentos de dificuldades e a meu pai, David Barros de Oliveira, pelo apoio, conselhos, broncas e principalmente amor .

Ao meu orientador, Professor Alfredo Braga Furtado, pela disponibilidade e qualidade da orientação prestada.

A minha namorada Marla Fernandes, por estar ao meu lado nesse momento importante da minha vida, por toda paciência, incentivo, compreensão, carinho e companheirismo.

Ao meu grande amigo Eliwelton Gomes Paes, que está comigo no desenvolvimento desse trabalho com muito profissionalismo e dedicação.

A toda equipe do setor de gestão de processos e tecnologia da informação do hospital universitário João de Barros Barreto, em especial Bruce Leite, Chefe do setor, Tonny Miralha, gerente de redes, Lídia Beirão, analista de sistemas e Edney Viana, analista de sistemas, pelo apoio profissional, pela oportunidade de desenvolver este trabalho, pelos conselhos e também pela amizade.

Agradeço também a todos, que direta ou indiretamente, contribuíram até aqui em minha jornada acadêmica. No contexto acadêmico, sempre há pessoas que, com pequenas ou grandes ações, são responsáveis pela materialização de um projeto ou simplesmente de um sonho.

David Barros de Oliveira Junior

“A persistência é o caminho do  
êxito.”  
(Charles Chaplin)

## RESUMO

O presente trabalho apresenta o desenvolvimento (especificação, implementação e avaliação) de um ferramenta *WebApp*, cujo objetivo é executar, de maneira informatizada, o processo de solicitação de Autorização de Procedimento Ambulatorial de Alta Complexidade (APAC) do Hospital Universitário João de Barros Barreto (HUIBB). O processo de desenvolvimento do sistema envolveu a elaboração de três documentos de especificação, antes da implementação do sistema: especificação de requisitos, especificação de análise e especificação de projeto. A ferramenta (software) resultante do processo de desenvolvimento busca auxiliar e otimizar o processo de solicitação desses procedimentos, por meio de uma área exclusiva para criação e impressão de formulários no padrão aceito pelo Sistema Único de Saúde (SUS).

PALAVRAS-CHAVES: WebApp, APAC, HUIBB.

## **ABSTRACT**

The present work presents the development (specification, implementation and evaluation) of a WebApp tool, whose objective is to perform, in a computerized way, the process of requesting the Authorization of High Complexity Ambulatory Procedure (APAC) from the University Hospital João de Barros Barreto (HUJBB). The system development process involved the preparation of three specification documents, prior to system implementation: requirements specification, analysis specification and design specification. The tool resulting from the development process seeks to aid and optimize the process of requesting these procedures, through a unique area for creating and printing forms in the standard accepted by the Unified Health System (SUS).

**KEYWORDS:** WebApp, APAC, HUJBB.

## LISTA DE SIGLAS

CID 10	Código Internacional de Doenças
APAC	Autorização de Procedimento Ambulatorial de Alta Complexidade
HUJBB	Hospital Universitário João de Barros Barreto
AGHU	Aplicativo de Gestão para Hospitais Universitários
SGPTI	Setor de Gestão de Processos e Tecnologia da Informação
SRAS	Setor de Regulação e Avaliação em Saúde
UML	Linguagem de Modelagem Unificada
IDE	Ambiente de Desenvolvimento Integrado
IHC	Interação Humano-Computador
SGBD	Sistema de Gerenciamento de Banco de Dados
SQL	Linguagem de Consulta Estruturada

# SUMÁRIO

<b>1. INTRODUÇÃO</b> .....	12
1.1 Objetivos .....	13
1.1.1 Objetivo geral .....	13
1.1.2 Objetivos específicos .....	13
1.2 Justificativa .....	14
1.3 Contribuições .....	15
1.4 Organização do trabalho .....	15
<b>2. REFERENCIAL TEÓRICO</b> .....	16
2.1 Engenharia de Software .....	16
2.1.1 Processo de Software .....	16
2.1.2 Modelo de Processo de Software .....	17
2.1.2.1 Movimento Ágil .....	17
2.1.2.1.1 Scrum .....	18
2.2 Engenharia de Requisitos .....	20
2.3 Documentação de Requisitos Utilizando Modelos .....	21
2.3.1 UML .....	21
2.3.1.1 Diagrama de Casos de Uso .....	23
2.3.1.2 Diagrama de Pacotes .....	24
2.3.1.3 Diagrama de Classes .....	25
2.3.1.4 Diagrama de Sequência .....	26
2.3.1.5 Diagrama de Máquina de Estados .....	27
<b>3. FERRAMENTAS E TECNOLOGIAS UTILIZADAS NO DESENVOLVIMENTO</b> .....	29
3.1 Java EE .....	29
3.2 PrimeFaces .....	30
3.3 MySQL .....	31
3.4 PHPMyAdmin .....	31
<b>4. ESPECIFICAÇÃO DE REQUISITOS, ANÁLISE, PROJETO E IMPLEMENTAÇÃO DO APAC BARROS BARRETO</b> .....	32
4.1 Especificação de Requisitos do APAC Barros Barreto .....	32
4.2 Especificação de Análise do APAC Barros Barreto .....	34
4.3 Especificação de Requisitos do APAC Barros Barreto .....	35

4.3.1	Projeto Interação Humano-Computador .....	35
4.3.2	Projeto de Banco de Dados.....	36
4.3.3	Projeto de Componentes.....	40
4.4	Implementação da Ferramenta.....	42
4.4.1	Planejamento .....	42
4.4.2	Banco de Dados .....	43
4.4.3	Ambiente de Desenvolvimento.....	47
4.4.4	Finalização do Desenvolvimento.....	48
<b>5.</b>	<b>RESULTADOS .....</b>	<b>50</b>
5.1	Usabilidade .....	50
5.2	Funcionalidade .....	52
5.3	Confiabilidade .....	55
5.4	Eficiência.....	57
<b>6.</b>	<b>CONCLUSÃO .....</b>	<b>59</b>
	<b>REFERÊNCIAS.....</b>	<b>60</b>
	<b>APÊNDICE A – MODELO PADRÃO DO LAUDO MÉDICO PARA SOLICITAÇÃO DE PROCEDIMENTO AMBULATORIAL DE ALTA COMPLEXIDADE .....</b>	<b>62</b>
	<b>APÊNDICE B – DOCUMENTO DE ESPECIFICAÇÃO DE REQUISITOS DO APAC BARROS BARRETO.....</b>	<b>64</b>
	<b>APÊNDICE C – DOCUMENTO DE ESPECIFICAÇÃO DE ANÁLISE DO APAC BARROS BARRETO.....</b>	<b>79</b>
	<b>APÊNDICE D – DOCUMENTO DE ESPECIFICAÇÃO DE PROJETO DO APAC BARROS BARRETO.....</b>	<b>106</b>
	<b>APÊNDICE E – QUESTIONÁRIO DE AVALIAÇÃO ONLINE .....</b>	<b>136</b>

## 1. INTRODUÇÃO

O Hospital Universitário João de Barros Barreto é referência regional em Pneumologia, Infectologia e Endocrinologia e Diabetes, e Referência Nacional em AIDS. O hospital oferece diversos serviços, como consultas médicas, procedimentos ambulatoriais de alta complexidade entre outras assistências à saúde da população, por meio do Sistema Único de Saúde (SUS), como também atua na área de Ensino e Pesquisa e na geração e sistematização de conhecimentos.

Procedimentos ambulatoriais são aqueles que podem ser realizados em consultórios e/ou clínicas, para alguns dos quais pode ser necessária aplicação de anestésicos locais associadas ou não à sedação. Estes procedimentos requerem autorização por parte do SUS para serem realizados nos hospitais públicos e esta autorização é solicitada através do laudo médico via formulário padrão do próprio Sistema Único de Saúde.

O laudo médico para solicitação de Autorização de Procedimento Ambulatorial de Alta Complexidade (APAC) é o documento de justificativa para que um (ou mais de um) procedimento ambulatorial possa ser efetuado.

O atual processo de solicitação de Autorização de Procedimento Ambulatorial de Alta complexidade se dá por meio várias etapas.

Primeiramente, o profissional da área da saúde, por meio de um formulário padrão do Sistema Único de Saúde, preenche os dados para solicitar a autorização. Nele estarão descritos os dados do paciente, da instituição, o(s) procedimento(s) solicitado(s), as justificativas para o procedimento e os dados do profissional. O modelo oficial do formulário de autorização pode ser verificado no APÊNDICE A.

Após esta etapa, o formulário é enviado à entidade responsável por validar, ou não, a autorização. Em seguida o documento é reenviado ao hospital, para o setor administrativo contendo os dados referentes à autorização.

Os dados (quantitativo de solicitações, solicitante, CID 10 relacionado, solicitações autorizadas e não autorizadas) relacionados às solicitações de Autorização de Procedimento Ambulatorial de Alta Complexidade serão utilizados nas prestações de contas do hospital solicitante.

Atualmente não foram encontrados *softwares* que satisfazem esta necessidade. Neste contexto, o hospital determinou ter sua ferramenta específica para este fim.

## 1.1 Objetivos

### 1.1.1 Objetivo geral

Desenvolvimento do sistema APAC Barros Barreto, para solicitação de Autorização de Procedimentos Ambulatoriais de Alta Complexidade (APAC), dentro das normas do Sistema Único de Saúde (SUS), de forma semiautomática.

### 1.1.2 Objetivos específicos

Fazer pesquisa bibliográfica sobre processos de desenvolvimento de software e ferramentas e métodos de engenharia de software, aplicados no desenvolvimento de sistemas;

Coletar requisitos do sistema a ser desenvolvido, por meio de entrevistas dos responsáveis pelos setores de Gestão de Processos e Tecnologia da Informação (SGPTI) e de Regulação e Avaliação em Saúde (SRAS);

Fazer controle do cadastro de utilizadores (inclusão, edição, exclusão e consulta);

Efetuar preenchimento de forma semiautomática e emissão de formulários conforme as normas de Autorização para procedimentos de Alta Complexidade (APAC) do SUS (Sistema Único de Saúde);

Emitir quantitativo relacionado aos procedimentos utilizados, doenças diagnosticadas, setores e usuários do sistema;

Registrar as principais atividades desempenhadas pelos usuários do sistema;

Produzir a especificação necessária para o desenvolvimento do software, como documentos de especificação de requisitos, de análise e de projeto que possibilitem a implementação do software;

Avaliação de usabilidade da ferramenta por usuários que utilizarão em seu dia a dia; proceder aos ajustes necessários com base nas inadequações porventura verificadas.

## 1.2 Justificativa

O procedimento para solicitação e autorização de exames e de procedimentos ambulatoriais, no Hospital Universitário João de Barros Barreto, é todo feito de forma manual adicionando tempo ao atendimento, tendo em vista que o profissional responsável precisa preencher todas as informações do paciente, informações do procedimento, seus próprios dados profissionais e dados da instituição. Para alguns destes dados se faz necessário uma busca em lista de apoio. Este tempo poderia ser mais bem aproveitado, colaborando para o atendimento de mais pacientes.

Caso alguma solicitação prescrita não seja compreendida em virtude da caligrafia ou outro motivo que torne o documento ilegível por seu receptor, este é reenviado ao Hospital para correção. Outro grave problema está ligado à emissão dos relatórios relacionados com as solicitações de autorização de procedimentos, não há um sistema que gerencie as informações (paciente, procedimento e dados médicos), sendo assim quando necessária a contagem e feita de forma manual dificultando a elaboração de um relatório preciso para fins de prestação de contas.

O Sistema foi proposto para otimizar o trabalho, tornando-o mais simples e eficaz. A ferramenta disponibiliza uma área para criação de formulários de solicitação Autorização de Procedimentos Ambulatoriais de Alta Complexidade. O Sistema, por intermédio dos comandos do profissional, efetua uma busca (informações do paciente, do procedimento, do profissional) no sistema principal do Hospital, Aplicativo de Gestão para Hospitais Universitários (AGHU), fazendo com que todas as informações necessárias para o formulário sejam preenchidas quase que automaticamente, gerando um formulário com uma legibilidade maior.

A aplicação também conta com uma área específica para emissão de relatórios; estes, por sua vez, fornecem o quantitativo das solicitações efetuadas, procedimentos solicitados, CIDs 10 relacionados, profissionais solicitantes, solicitações autorizadas e não autorizadas.

### 1.3 Contribuições

Otimizar o tempo de consulta médica relacionada ao preenchimento de solicitação APAC; o tempo poderá ser melhor utilizado no atendimento a outros pacientes;

Otimização e informatização do processo de solicitação de Autorização de Procedimento Ambulatorial de Alta Complexidade;

Maior legibilidade nos formulários enviados ao SUS para autorização;

Registro e controle das atividades relacionadas ao processo de solicitação da autorização;

Emissão do quantitativo relacionado ao laudo APAC (procedimentos, CID 10, solicitantes, setores) de forma precisa;

### 1.4 Organização do trabalho

Este trabalho está estruturado da seguinte maneira: o Capítulo 2 apresenta o Referencial Teórico utilizado nas etapas do desenvolvimento do projeto. O Capítulo 3 apresenta Ferramentas e Tecnologias Utilizadas. O Capítulo 4 apresenta as fases do desenvolvimento: Especificação de Requisitos, Análise, Projeto e Implementação da Ferramenta. O Capítulo 5 apresenta Avaliação do Sistema. O trabalho é finalizado com a Conclusão e os Apêndices.

## 2. REFERENCIAL TEÓRICO

Este Capítulo tem por objetivo descrever os principais conceitos, métodos e ferramentas empregadas neste trabalho. O capítulo está estruturado da seguinte forma: Primeiramente serão abordados alguns conceitos de diferentes autores referentes à metodologia utilizada no trabalho. Em seguida, são apresentadas as ferramentas empregadas no desenvolvimento do sistema.

### 2.1 Engenharia de Software

A engenharia de software tem por objetivo apoiar o desenvolvimento profissional de software, mais do que a programação individual. Ela inclui técnicas que apoiam especificação, projeto e evolução de programas, que normalmente não são relevantes para o desenvolvimento de software pessoal. (SOMMERVILLE, 2011)

SWEBOK *apud* Furtado e Costa Jr. (2010, p.51) define Engenharia de *Software* como “a aplicação de uma abordagem sistemática, disciplinada, quantificável ao desenvolvimento à operação e à manutenção de *software*, isto é, a aplicação de engenharia ao *software*”. Acrescenta ainda que a Engenharia de Software inclui o estudo de abordagens que levem a disciplina de engenharia à área de *software*. Pressman (2011) aponta que, além de disciplina, devem-se acrescentar adaptabilidade e agilidade, qualidades exigidas pela atual demanda de computação.

#### 2.1.1 Processo de Software

Um processo de *software* é um conjunto de atividades relacionadas que levam à produção de um produto de software. Essas atividades podem envolver o desenvolvimento de *software* a partir do zero em uma linguagem padrão de programação como Java ou C. (SOMMERVILLE, 2011).

Pressman (2011, p. 40) tem uma visão parecida e define um processo de *software* como “um conjunto de atividades, ações e tarefas realizadas na criação de algum produto de trabalho.”.

Segundo Sommerville (2011), existem muitos processos de *software* diferentes, mas todos devem incluir quatro atividades fundamentais para a engenharia de software:

1. Especificação de software. A funcionalidade do software e as restrições a seu funcionamento devem ser definidas.
2. Projeto e implementação de software. O software deve ser produzido para atender às especificações.
3. Validação de software. O software deve ser validado para garantir que atenda às demandas do cliente.
4. Evolução de software. O software deve evoluir para atender às necessidades de mudança dos clientes.

Os processos de *software* são complexos e, como todos os processos intelectuais e criativos, dependem de pessoas para tomar decisões e fazer julgamentos. (SOMMERVILLE, 2011)

Nesse contexto, cada instituição pode adotar o modelo de processo de *software* que considerar necessário.

## 2.1.2 Modelo de Processo de Software

Sommerville (2011) diz que o modelo de processo de *software* é uma descrição simplificada de um processo de *software*. Cada modelo representa uma perspectiva particular de um processo e, portanto, fornece informações parciais sobre ele.

### 2.1.2.1 Movimento Ágil

Sommerville (2011) afirma que a metodologia prescritiva se fazia dominante no campo de engenharia de *software* no final da década de 1980, bem como a forma para desenvolver o melhor software era por meio de um planejamento metódico do projeto, qualidade da segurança formalizada, entre outras circunstâncias que levavam a um desenvolvimento de *software* severo e controlado.

Ainda que as ideias subjacentes que norteiam o desenvolvimento ágil tenham estado conosco há muitos anos, somente durante a década de 1990 é que essas ideias foram consolidadas em um “movimento”.

Em essência, os métodos ágeis foram desenvolvidos em um esforço para vencer as fraquezas percebidas e reais da engenharia de *software* convencional. O desenvolvimento ágil pode fornecer importantes benefícios, mas ele não é aplicável a todos os tipos de projetos, de produtos. Ele também não é contrário à sólida prática de engenharia de software e pode ser aplicado como uma filosofia prevalecte a todo o trabalho de *software* (PRESSMAN, 2011).

Pressman (2011) afirma que existem muitas abordagens ágeis, porém as mais comuns, e que se encontram em uso no setor, são:

- Desenvolvimento de software adaptativo;
- Scrum;
- Método de desenvolvimento de sistemas dinâmicos;
- Crystal;
- Desenvolvimento dirigido a funcionalidades;
- Desenvolvimento de software enxuto;
- Modelagem ágil;
- Processo unificado ágil.

Pressman (2011) salienta que todos os métodos descritos acima estão em conformidade com o movimento ágil.

#### 2.1.2.1.1 Scrum

*Scrum* é um *framework* utilizado para aprimorar a produção de projetos e usar suas características de desenvolvimento ágil para o desenvolvimento de *software*.

O *Scrum* é de suma importância, pois discorre de metodologias ágeis para a gestão de projetos de desenvolvimento de *software*, a qual expressa dinamismo no ambiente de negócios já que a Tecnologia da Informação transpassa pela constante mudança e conseqüentemente por pressões por inovações e resultados melhores do produto final (MACHADO & MEDINA,

2017).

*Scrum* é utilizado para otimizar a produção de projetos e usar suas características de desenvolvimento ágil para o desenvolvimento de software. Suas características são:

- Agilidade para gerenciar e controlar o desenvolvimento;
- Otimização de desenvolvimento;
- Desempenho equivalente entre as partes;
- Produtividade aumentada durante o desenvolvimento.

Por essa razão, o *Scrum* acaba sendo o mais utilizado no desenvolvimento devido aos benefícios oferecido pelo *framework*.

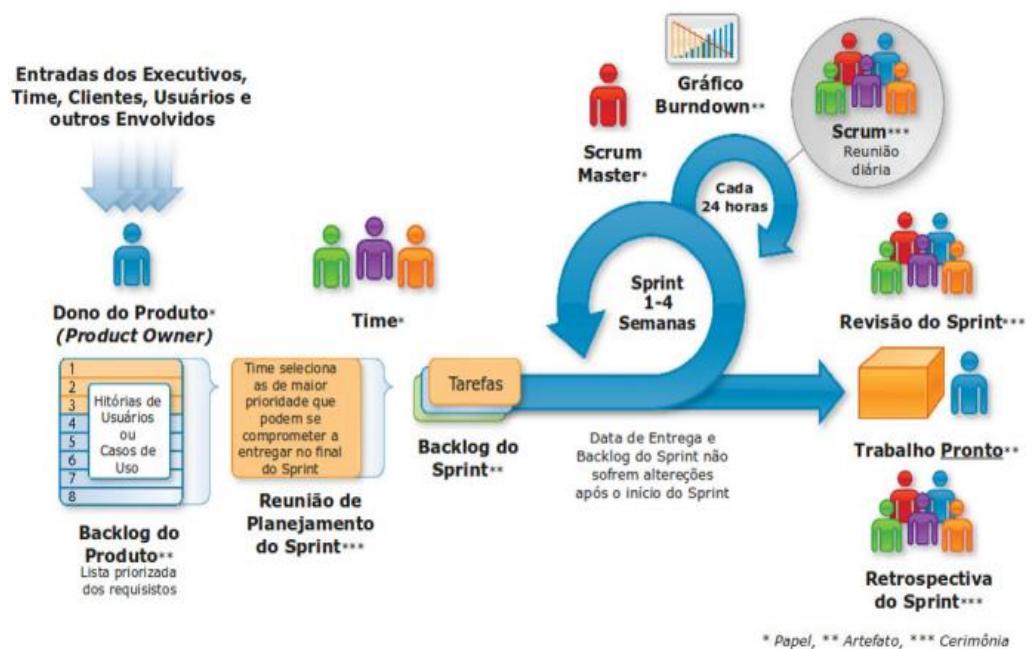


Figura 1. Ciclo de vida do Scrum.

Fonte: <http://ned.unifenas.br/cursosgratuitos/201302/scrum/funcionamento.html>

O *Product Owner* pode ser representado por um cliente, gerente, ou até mesmo por um membro da equipe de desenvolvimento, o mesmo é responsável por definir todas as funcionalidades e prioridades para adicionar ao *Product Backlog*. Sommerville (2011) define o *backlog* como o ponto de partida do produto. O *backlog* é revisto durante a fase de avaliação da *Sprint*, e as prioridades e os riscos são identificados. Assim o cliente que está envolvido

no projeto pode introduzir novos requisitos ou tarefas no *backlog*.

O *Scrum Team* olha para o *Product Backlog* priorizado, seleciona os itens mais prioritários e se compromete a entregá-los ao final de um Sprint (iteração). Estes itens transformam-se no *Sprint Backlog*.

Compete ao *Scrum Master* a responsabilidade de tornar o ambiente funcional, garantir que o time seja produtivo e que atenda ao processo estabelecido, bem como certificar que as práticas do *Scrum* estejam sendo adotadas e, o projeto e a cultura organizacional estejam em harmonia com as metas e o retorno sobre o investimento seja alcançado a cada Sprint (Fonseca, 2009). *Scrum Master* remove obstáculos e evita interferências, garante colaboração, guardião das práticas do Scrum, ou seja, mantém o foco na meta da Sprint.

No *Scrum* os projetos são divididos em ciclos. Esses ciclos são chamados de *Sprint*, onde a equipe planeja e define os prazos a ser entregue em reuniões mensais ou semanais, podendo a mesma ser revisada diariamente por meio do *Daily Scrum* para a priorização da responsabilidade do desenvolvimento da equipe.

A cada dia da *Sprint* a equipe faz reuniões diárias que chama *Daily Scrum*, normalmente no mesmo lugar e no mesmo horário geralmente feito na parte da manhã para estabelecer prioridades do dia. As reuniões possuem efeitos admiráveis, pois o *Team Scrum* compreende a importância dos seus compromissos.

## 2.2 Engenharia de Requisitos

O processo de descobrir, analisar, documentar e verificar esses serviços e restrições é chamado engenharia de requisitos. (SOMMERVILLE, 2011)

Nesta etapa são levantados requisitos ou informações fundamentais para modelagem do sistema. Para poder analisar e entender o problema a ser solucionado e estabelecer, de maneira precisa, as funções que o sistema deve executar.

Com a análise de requisitos serão estabelecidos requisitos com a

intenção de transformar a explicação, de alto nível desses requisitos, em uma lista precisa que possa ser utilizada como informação para o restante da fase de análise.

A classificação dos requisitos, em grupos com níveis distintos de definições, permite que não ocorram problemas durante o processo de engenharia de requisitos.

### 2.2.1 Requisitos funcionais

Os requisitos funcionais são declarações de funções que o sistema deve fornecer como o sistema deve reagir a entradas específicas e como deve se comportar em determinadas situações. Em alguns casos os requisitos funcionais podem também explicitamente declarar o que o sistema não deve fazer. (SOMMERVILLE, 2011).

### 2.2.2 Requisitos não funcionais

Requisitos não funcionais de *software* são aqueles que descrevem não o que o sistema fará, mas como ele fará. Assim, por exemplo, existem requisitos de desempenho, requisitos da interface externa do sistema, restrições de projeto e atributos da qualidade.

## 2.3 Documentação de Requisitos Utilizando Modelos

A documentação de requisitos será utilizada por múltiplos profissionais durante o desenvolvimento do software. Neste contexto, se faz necessário o uso de uma linguagem parâmetro e conhecida por todos.

### 2.3.1 UML

Furtado e Costa Jr (2010) definem a UML como uma linguagem para especificação de artefatos de software, assim como para modelagem de negócios.

A UML é constituída de itens, relacionamentos e diagramas que facilitam

um melhor entendimento da modelagem.

Os itens estão classificados como:

- Estruturais (classe, interface, casos de uso e componentes); Comportamentais (interações, máquinas de estado);
- Grupos de elementos (pacotes, *frameworks* e subsistemas);
- Anotações (notas).

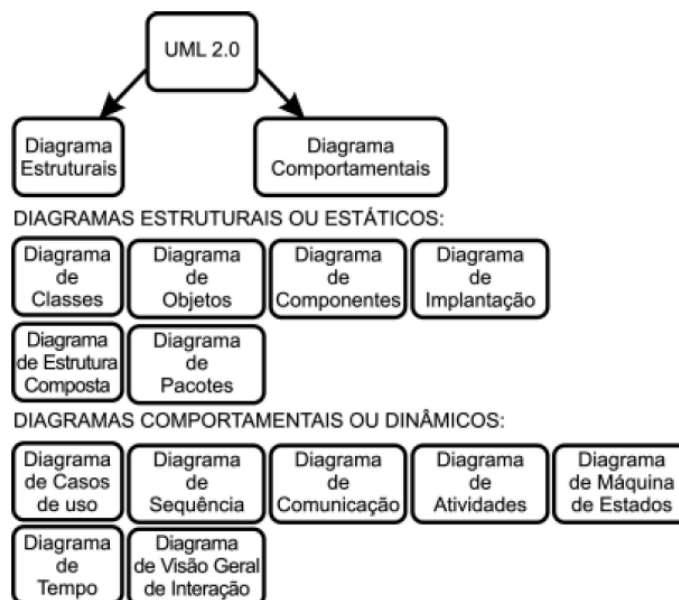
A UML possui quatro tipos de relacionamentos:

1. Dependência;
2. Associação;
3. Generalização
4. Realização.

Booch *et al* (2006, p. 26) ) enunciam que o diagrama estabelece a associação entre os itens e os relacionamentos, com o intuito de formar um conjunto de representação gráfica, objetivando um maior entendimento sobre o sistema. A UML, em sua versão 2.0, inclui 13 diagramas:

1. Diagrama de classes
2. Diagrama de objetos
3. Diagrama de componentes
4. Diagrama de estrutura compostas
5. Diagrama de casos de uso
6. Diagrama de sequências
7. Diagrama de comunicações
8. Diagrama de gráfico de estados
9. Diagrama de atividades
10. Diagrama de implantação
11. Diagrama de pacote
12. Diagrama de temporização
13. Diagrama de visão geral da interação

A Figura 2 divide os diagramas, de forma separada, em dois subgrupos: Estruturais e comportamentais.



**Figura 2. Diagramas da UML 2.0.**

**Fonte: (Furtado & Costa Jr, 2010, p. 106)**

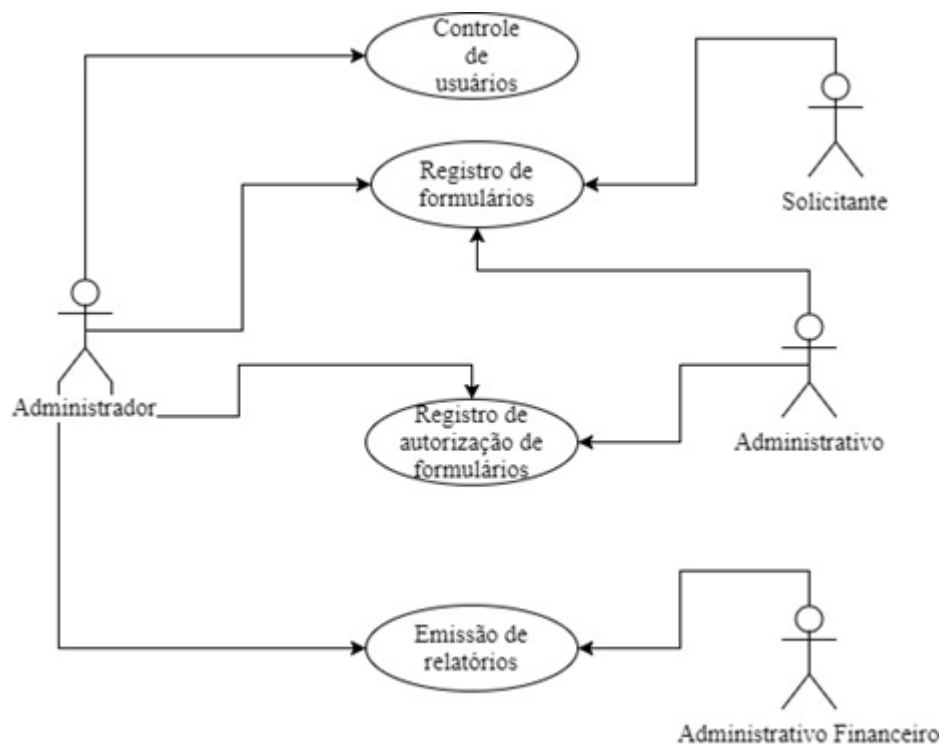
Os diagramas usados neste trabalho são: diagrama de casos de uso, diagrama de pacotes, diagrama de classes, diagrama de sequência e diagrama de máquina de estados.

### 2.3.1.1 Diagrama de Casos de Uso

Furtado e Costa Jr (2010), citam que o diagrama de casos de uso é utilizado para especificar os requisitos funcionais de um sistema. Por isso, é o primeiro a ser construído pelo analista de sistemas no seu trabalho de especificação.

O diagrama de casos de uso de um sistema apresenta todas as interações possíveis que deverão estar descritas nos requisitos do sistema. É fácil entender a simbologia do diagrama de casos de uso. Temos os atores representados por “bonecos-palito” que podem ser pessoas ou outros sistemas. As classes de interação são representadas por uma elipse. A ligação entre atores e interação é feita por meio de linhas. Pontas de flechas podem ser adicionadas opcionalmente para mostrar a direção da interação (SOMMERVILLE, 2011). Na Figura 3 é ilustrado um caso de uso do sistema APAC Barros Barreto. Este caso de uso faz a representação do contexto do

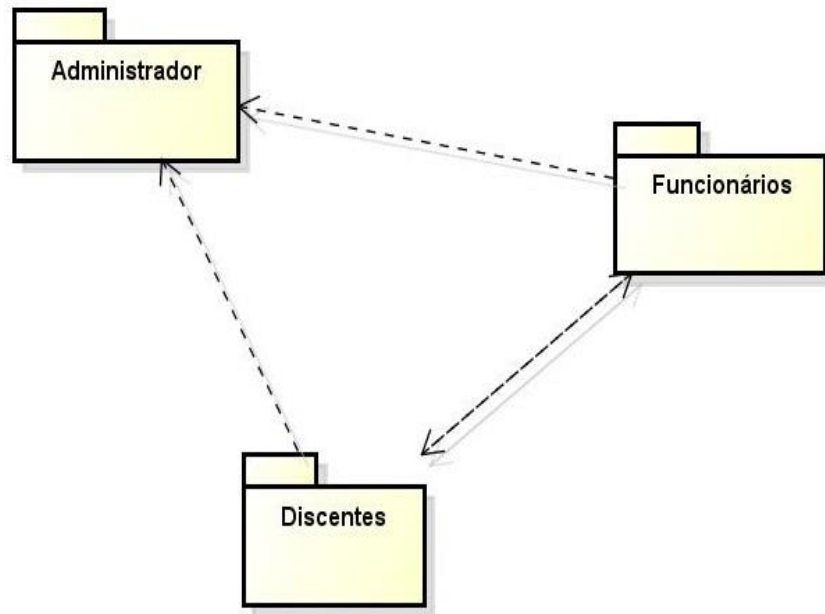
sistema, temos os atores: Administrador faz uso do Controle de Usuários, assim como todos os outros subsistemas. Temos o ator Solicitante interagindo com o Registro de Formulários. O ator Administrativo interage com o Registro de Formulários e com o Registro de autorização de formulários. Por fim, o ator Administrativo Financeiro que interage com o subsistema Emissão de Relatórios.



**Figura 3. Diagrama de casos de uso do APAC Barros Barreto.**

### 2.3.1.2 Diagrama de Pacotes

Para Furtado e Costa Jr (2010), o diagrama de pacotes mostra como os elementos do modelo são organizados em pacotes, como também as dependências entre pacotes. Os pacotes representam os subsistemas ou módulos do sistema. O diagrama de pacotes pode ser utilizado independentemente ou associado a outros diagramas.



**Figura 4. Exemplo de diagrama de pacotes.**

**Fonte: <http://danielcavalcanti.com.br>**

### 2.3.1.3 Diagrama de Classes

Os diagramas de classe são usados no desenvolvimento de um modelo de sistema orientado a objetos para mostrar as classes de um sistema e as associações entre essas classes. Em poucas palavras, uma classe de objeto pode ser pensada como uma definição geral de um tipo de objeto do sistema. Uma associação é um link entre classes que indica algum relacionamento entre essas classes. Consequentemente, cada classe pode precisar de algum conhecimento sobre sua classe associada. (SOMMERVILLE, 2011).

Segundo Fowler & Scott (1997) *apud* Furtado e Costa Jr (2010, p. 196) “um diagrama de classes descreve os tipos de objetos encontrados no sistema (ou seja, as classes dos objetos) e os diversos tipos de relações estáticas (associações herança) que existem entre esses diversos tipos”.

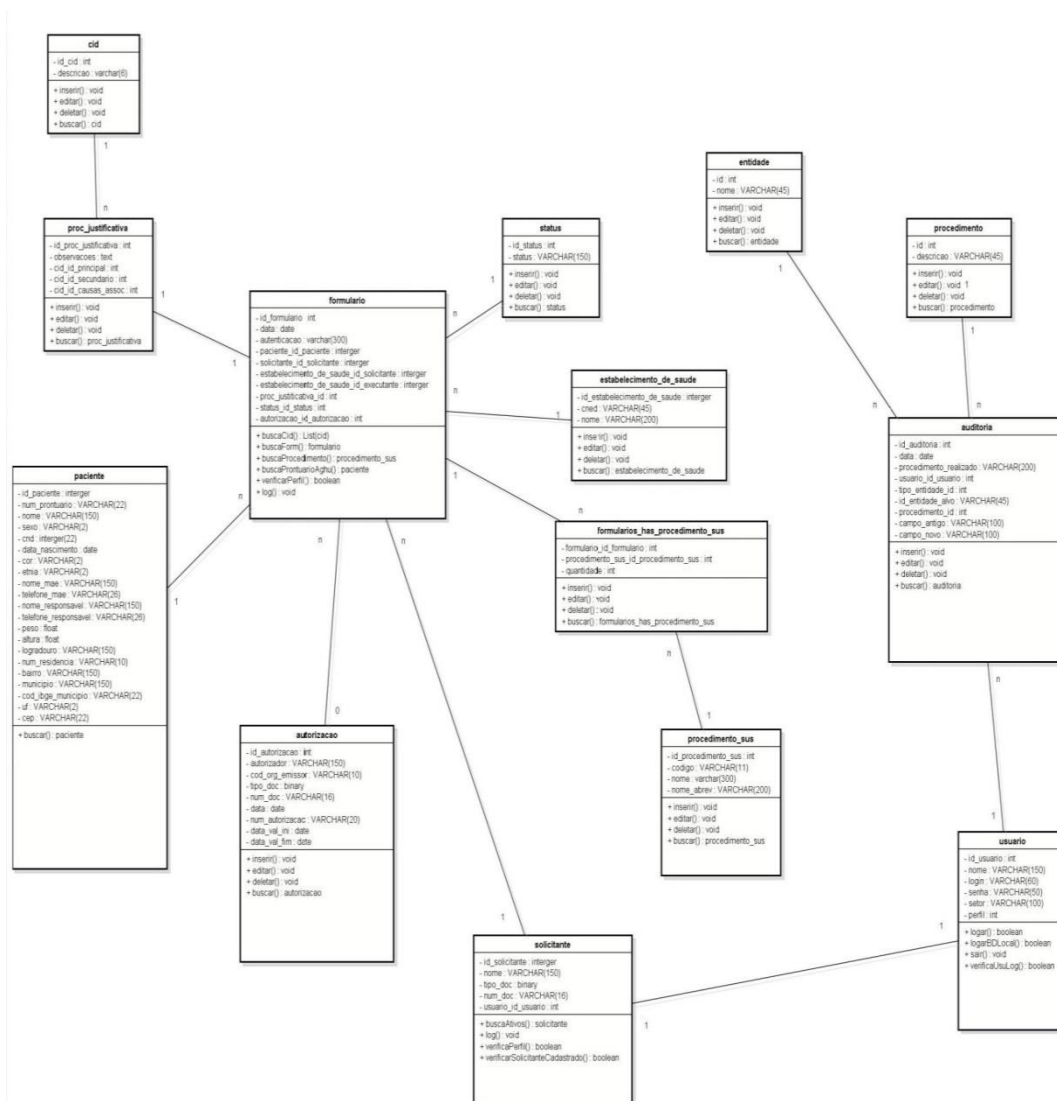


Figura 5. Diagrama de classes do APAC Barros Barreto.

### 2.3.1.4 Diagrama de Sequência

Os diagramas de sequência em UML são usados, principalmente, para modelar as interações entre os atores e os objetos em um sistema e as interações entre os próprios objetos. A UML tem uma sintaxe rica para diagramas de sequência, que permite a modelagem de vários tipos de interação (SOMMERVILLE, 2011).

Sommerville (2011) expõe como deve ser feita a leitura do diagrama de sequência: na parte superior temos os atores e objetos; Interações entre objetos são feitas por linhas anotadas; A linha de vida do objeto é indicada por um retângulo seguido de uma linha tracejada que a leva ao objeto; As

seqüências de interações devem ser lidas de cima para baixo.

A Figura 6 é um exemplo de diagrama de seqüência usado APAC Barros Barreto.

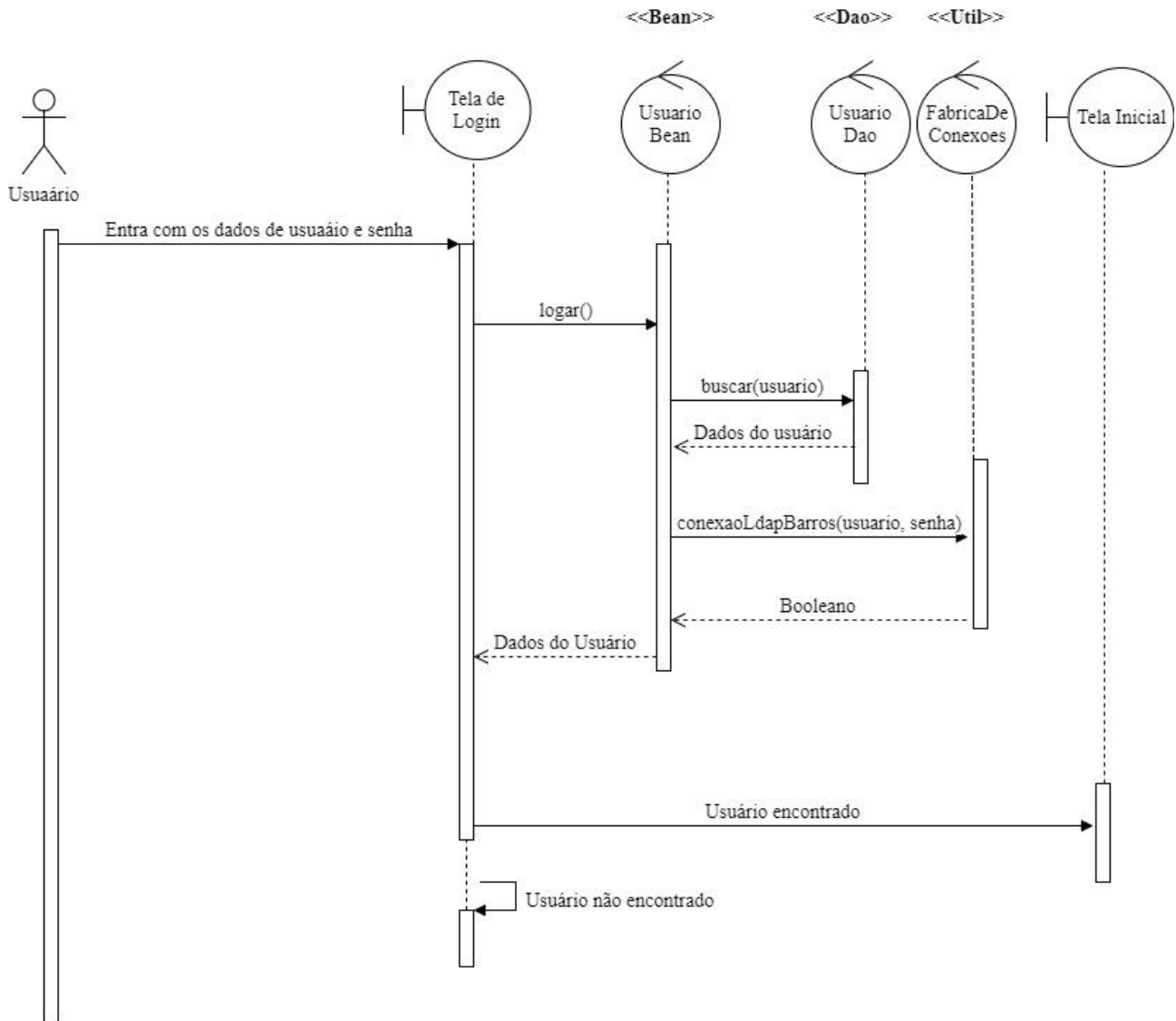
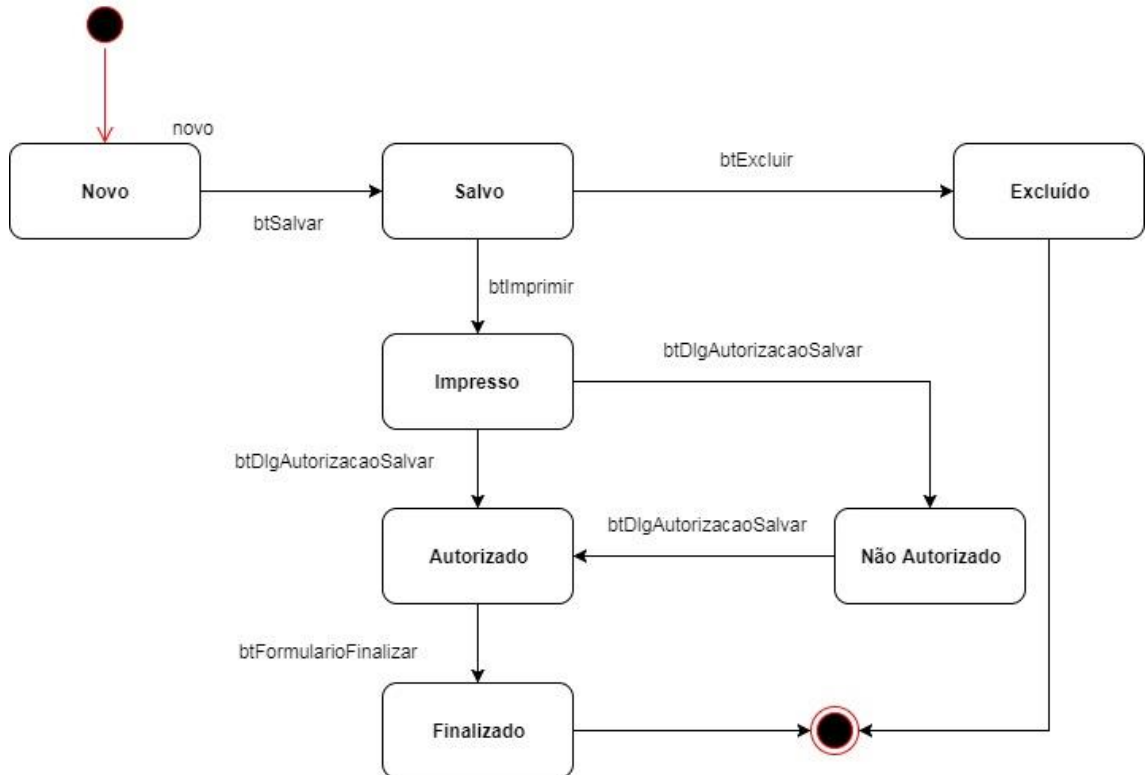


Figura 6. Exemplo de um diagrama de seqüência do APAC Barros Barreto.

### 2.3.1.5 Diagrama de Máquina de Estados

Para Furtado e Costa Jr (2010) o diagrama de máquina de estados é tipicamente um complemento para a descrição das classes. Este diagrama mostra todos os estados possíveis em que o objeto de certa classe podem se encontrar, além dos eventos do sistema que provocam mudanças de um estado para o outro.

No exemplo da figura 8, podemos ver um diagrama de máquina de estados utilizado do APAC Barros Barreto.



**Figura 8. Exemplo de um diagrama de máquina de estados do APAC Barros Barreto**

### 3. FERRAMENTAS E TECNOLOGIAS UTILIZADAS NO DESENVOLVIMENTO

Este Capítulo tem como intenção expor, de forma resumida, uma descrição das principais ferramentas e tecnologias utilizadas no desenvolvimento e implementação do sistema APAC Barros Barreto. A aplicação foi desenvolvida com o uso da IDE *NetBeans*, utilizando *Java EE* (*Java Enterprise Edition*), com o auxílio do *Framework PrimeFaces* e a base de dados foi criada em *MySQL*, utilizando o sistema de gestão de base de dados (SGBD), *phpmyadmin*. Estas tecnologias e ferramentas são descritas em seguida.

#### 3.1 Java EE

Java é uma linguagem de programação orientada a objetos desenvolvida na década de 1990.

O *Site* da linguagem, [java.com](http://java.com), mostra a seguinte definição: O Java é uma tecnologia usada para desenvolver aplicações que tornam a Web mais divertida e útil.

O *Java EE* consiste em uma série de especificações, de como deve ser implementado um software que faz cada um desses serviços (CAELUM, 2015).

A partir do *Java EE 6*, existe o termo "*application server web profile*", onde se referênciamos os servidores que não oferecem tudo, mas um grupo menor de especificações, consideradas essenciais para o desenvolvimento web (CAELUM, 2015).

Para o desenvolvimento da aplicação deste trabalho foi utilizado o *Java EE 7 Web*.

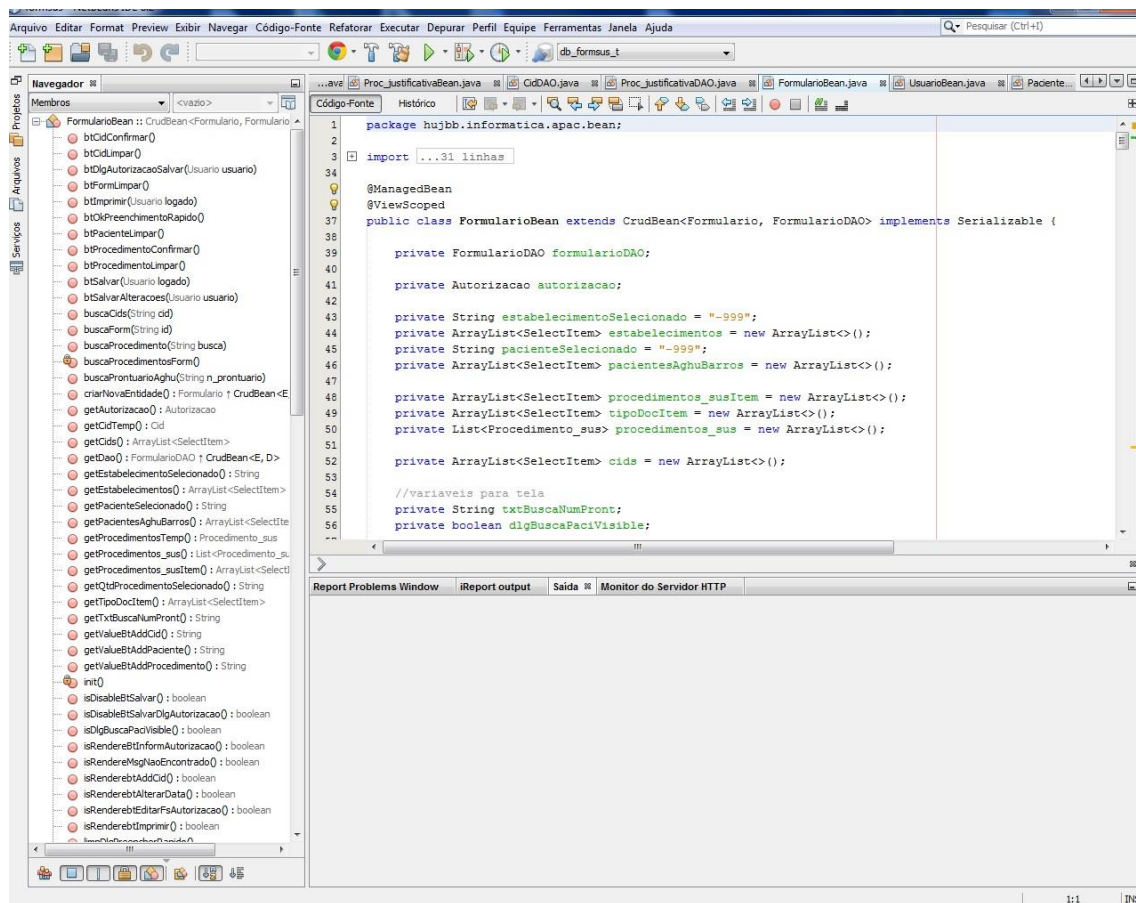


Figura 9. Exemplo de um código Java utilizado no APAC Barros Barreto

### 3.2 PrimeFaces

A biblioteca do *PrimeFaces* é composta por cerca de 100 componentes de interface todos estes personalizados e de código aberto, e implementa tecnologia AJAX, gráficos animados *JavaScript* entre outros.

A utilização do *PrimeFaces* em projetos proporciona uma infinita gama de possibilidades para criação de layouts já que em seu *ShowCase* acerca de 30 temas diferentes que podem ser facilmente adicionados ao seu projeto (SANTOS, 2010).

Segundo Gonçalves (2007):

AJAX é a sigla de *Asynchronous JavaScript and XML*, não é uma tecnologia e sim o uso de tecnologias incorporadas que tem as principais o *Javascript* e o *XML*, onde juntos são capazes de torna o navegador mais interativo,

utilizando-se de solicitações assíncronas de informações. É sim um conjunto de tecnologia; cada uma com sua forma de exibir e interagir com o usuário.

O *PrimeFaces* será usado na construção das interfaces, ela é uma biblioteca que define componentes JSF.

### 3.3 MySQL

Souza (2017) afirma que o MySQL é um servidor e gerenciador de banco de dados (SGBD) relacional, de licença dupla (sendo uma delas de software livre), projetado inicialmente para trabalhar com aplicações de pequeno e médio portes, mas hoje atendendo a aplicações de grande porte e com mais vantagens do que seus concorrentes.

Neste trabalho, foi utilizada a versão 5.2 do MySQL.

### 3.4 PHPMyAdmin

Bento (2014) afirma que esta é uma ferramenta escrita em PHP usada para prover melhor o gerenciamento do o MySQL por meio de diversas opções, como criação bancos, usuários, tabelas, inserir, pesquisar, remover registros etc.

Até aqui foi apresentado o referencial teórico utilizado neste trabalho. Estes conceitos são fundamentais para o entendimento da criação de um produto de software. Com esses conhecimentos adquiridos, é possível especificar e implementar o APAC Barros Barreto. O próximo Capítulo aborda estes tópicos.

## **4. ESPECIFICAÇÃO DE REQUISITOS, ANÁLISE, PROJETO E IMPLEMENTAÇÃO DO APAC BARROS BARRETO**

Este capítulo tem por objetivo descrever as fases do desenvolvimento do sistema APAC Barros.

O processo de desenvolvimento de software iniciou-se com a coleta de requisitos. A princípio é construído o documento de especificação de requisitos. Nele serão definidos o escopo, os objetivos do software, finalizando com a elaboração dos diagramas de casos de uso.

A próxima etapa é a construção dos documentos de especificação de análise. Neste documento estarão definidos os diagramas de pacotes, diagramas de classes, diagramas de sequências, diagrama de máquina de estados e dicionário de dados (breve descrição dos dados manipulados pelo sistema).

A terceira fase se deu com a produção do documento de especificação de projeto. Nele serão definidas a plataforma computacional, a arquitetura do sistema, o projeto de interação humano-computador (IHC) e o projeto de componentes. Com as informações coletadas e os documentos elaborados, o desenvolvedor tem o suporte necessário para implementar a sistema.

### **4.1 Especificação de Requisitos do APAC Barros Barreto**

Durante o desenvolvimento de software é necessário ter uma ideia inicial do que se deseja desenvolver. Esse processo, inicialmente, pode ser confuso, afinal, os envolvidos não sabem ao certo o que exatamente querem para o projeto. Então, um escopo inicial se faz necessário. A especificação de requisitos é uma das etapas principais do processo de desenvolvimento de um software.

A aplicação *Web* proposta neste trabalho tem como cliente-alvo o Hospital Universitário João de Barros Barreto.

O atual processo de solicitação de Autorização de Procedimento Ambulatorial de Alta Complexidade é todo feito de forma manual e utilizando papel impresso, requisitando uma informatização de suas etapas e um

gerenciamento eficiente das informações. A partir deste contexto, a análise de requisitos se deu por meio de reuniões periódicas (normalmente a cada 2 semanas) com o setores SGPTI e SRAS da instituição.

Ao final das coletas de requisitos, os objetivos do sistema estavam visíveis. A aplicação conta com as seguintes áreas:

- Área para a criação dos formulários: área onde o usuário preenche o formulário com todas as informações pertinentes daquele paciente, o usuário identificará o procedimento solicitado, por meio de uma aba de busca, com o código do procedimento ou o nome do procedimento. Igualmente para o campo do CID relacionado. Após a finalização do preenchimento, o formulário é exibido na tela, em PDF, no formato padrão do Sistema Único de Saúde, pronto para impressão.
- Área para o registro de autorização dos formulários: área responsável pelo preenchimento das informações relacionadas à autorização ou não de determinado formulário.
- Área para a emissão de relatórios: responsável pela emissão do quantitativo relacionado ao número de procedimentos solicitados, profissionais solicitantes, CIDs relacionados e número de solicitações autorizadas e não autorizadas.
- Área para criação de novos usuários: criação de novos utilizadores da aplicação.

Para tal, identificou-se a necessidade do APAC Barros Barreto contar com quatro diferentes níveis de acesso e suas respectivas responsabilidades. São eles: 'Solicitante', 'Administrativo', 'Administrativo Financeiro' e 'Administrador'.

Para cada perfil, identificou-se:

- Solicitante: responsável pelo preenchimento do formulário de solicitação de Autorização de Procedimentos Ambulatoriais de Alta Complexidade;
- Administrativo: responsável pelo preenchimento das informações relacionadas a autorização ou não autorização de um procedimento, isto é, ele quem indica se o procedimento ambulatorial foi autorizado ou não;
- Administrativo Financeiro: responsável pela emissão do quantitativo de profissionais solicitantes, procedimentos solicitados, CIDs relacionados e quantidade de autorizações aceitas ou não;
- Administrador: responsável pela criação de novos usuários do sistema e pode assumir o papel de qualquer um dos outros perfis.

A partir destas informações, o documento de especificação de requisitos foi elaborado, com seus respectivos casos de uso.

Após a finalização do Documento de Especificação de Requisitos é possível obter uma visão mais adequada da ferramenta. O próximo passo foi a elaboração do Documento de Especificação de Análise. Assunto que será tratado a seguir.

#### 4.2 Especificação de Análise do APAC Barros Barreto

O primeiro passo desta etapa se deu com a definição dos pacotes que comportariam as classes do sistema, então o diagrama de pacotes da aplicação foi desenvolvido. Este diagrama tem como objetivo garantir uma visualização *top-down* do sistema. Os pacotes foram divididos em:

- Pacote Bean: onde se encontram as classes de conexão entre telas e a própria aplicação.
- Pacote Dao (*Data Access Object*): onde se encontram as classes

que fazem a conexão com o banco de dados da aplicação.

Definidos os pacotes, a próxima etapa se deu com a definição das classes e criação do diagrama de classes.

O diagrama de classes é um dos mais importantes da UML, pois representa os vários objetos do sistema, seus métodos, comportamentos e relacionamentos. Visando um melhor entendimento dos métodos e dados concebidos para o diagrama de classes, formulou-se um dicionário de dados para compreensão do que estava representado em cada classe.

Objetivando representar de maneira simples e lógica a sequência de mensagens que transitam entre os objetos no sistema, também foi elaborado o diagrama de Sequência, este que por sua vez descreve o comportamento das mensagens no contexto dos casos de uso.

Durante a criação do documento de análise, observou-se que um formulário criado, poderia assumir diferentes estados durante seu ciclo de vida na aplicação. Nesse âmbito, para o componente de criação de formulário, foi elaborado um diagrama de Máquina de Estados responsável por demonstrar todos os possíveis estados do objeto durante a execução dos processos.

A estrutura completa de todos os diagramas, desenvolvidos nesta etapa de análise do sistema, está descrita no Documento de Especificação de Análise (ver APÊNDICE C). Após a finalização do documento, iniciou-se a elaboração da Especificação de Projeto do sistema.

#### 4.3 Especificação de Requisitos do APAC Barros Barreto

Este documento contém a Especificação de Projeto para o Sistema APAC Barros Barreto. Uma parte importante desse documento é o Projeto Interação Humano Computador. Nele serão apresentados alguns protótipos de tela importantes da aplicação.

##### 4.3.1 Projeto Interação Humano-Computador

Esta seção apresenta o protótipo da tela principal da aplicação. Na Figura



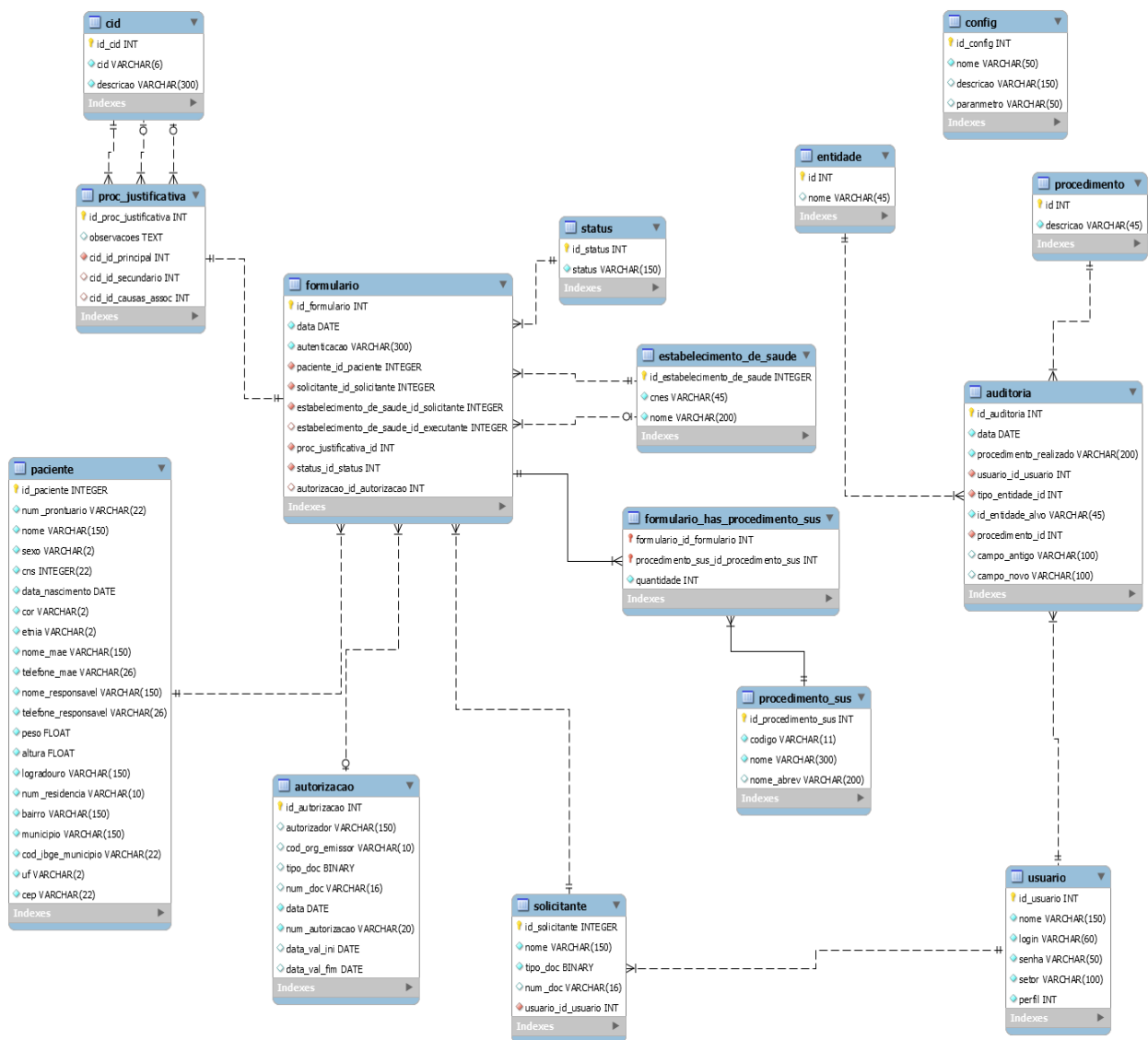


Figura 11. Diagrama de Entidade Relacional do APAC Barros Barreto.

- cid(id\_cid, cid, descricao);
  - id\_cid chave primária;
- proc\_justificativa(id\_proc\_justificativa, observacoes, cid\_id\_principal, cid\_id\_secundario, cid\_id\_causas\_assoc);
  - id\_proc\_justificativa chave primária;
  - cid\_id\_principal chave estrangeira referenciando a chave primária id\_cid da tabela cid;
  - cid\_id\_secundario chave estrangeira referenciando a chave primária id\_cid da tabela cid;

- cid\_id\_causas\_assoc chave estrangeira referenciando a chave primária **id\_cid** da tabela **cid**;
- formulario(**id\_formulario**, data, autenticacao, paciente\_id\_paciente, solicitante\_id\_solicitante, estabelecimento\_de\_saude\_id\_solicitante, estabelecimento\_de\_saude\_id\_executante, proc\_justificativa\_id, status\_id\_status, autorizacao\_id\_autorizacao);
  - **id\_formulario** chave primária;
  - paciente\_id\_paciente chave estrangeira referenciando a chave primária **id\_paciente** da tabela **paciente**;
  - solicitante\_id\_solicitante chave estrangeira referenciando a chave primária **id\_solicitante** da tabela **solicitante**;
  - estabelecimento\_de\_saude\_id\_solicitante chave estrangeira referenciando a chave primária **id\_estabelecimento\_de\_saude** da tabela **estabelecimento\_de\_saude**;
  - estabelecimento\_de\_saude\_id\_executante chave estrangeira referenciando a chave primária **id\_estabelecimento\_de\_saude** da tabela **estabelecimento\_de\_saude**;
  - proc\_justificativa chave estrangeira referenciando a chave primária **id\_proc\_justificativa** da tabela **proc\_justificativa**;
  - status\_id\_status chave estrangeira referenciando a chave primária **id\_status** da tabela **status**;
  - autorizacao\_id\_autorizacao chave estrangeira referenciando a chave primária **id\_autorizacao** da tabela **autorizacao**;
- paciente(**id\_paciente**, num\_prontuario, nome, sexo, cns, data\_nascimento, cor, etnia, nome\_mae, telefone\_mae, nome\_responsavel, peso, altura, logradouro, num\_residencia, bairro, municipio, cod\_ibge\_municipio, uf, cep);
  - **id\_paciente** chave primária;
- autorizacao(**id\_autorizacao**, autorizador, cod\_org\_emissor, tipo\_doc, num\_doc, data, num\_autorizacao, data\_val\_ini, data\_val\_fim);
  - **id\_autorizacao** chave primária;
- solicitante(**id\_solicitante**, nome, tipo\_doc, num\_doc, usuario\_id\_usuario);
  - **id\_solicitante** chave primária;
  - usuario\_id\_usuario chave estrangeira referenciando a chave primária

**id\_usuario** da tabela **usuario**;

- usuario(**id\_usuario**, nome, login, senha, setor, perfil);
  - **id\_usuario** chave primária;
- procedimento\_sus(**id\_procedimento\_sus**, codigo, nome, nome\_abrev);
  - **id\_procedimento\_sus** chave primária;
- formulario\_has\_procedimento\_sus(**formulario\_id\_formulario**, **procedimento\_sus\_id\_procedimento\_sus**, quantidade);
  - **formulario\_id\_formulario**, **procedimento\_sus\_id\_procedimento\_sus** chave primária composta;
  - formulario\_id\_formulario chave estrangeira referenciando a chave primária **id\_formulario** da tabela **formulario**;
  - procedimento\_sus\_id\_procedimento\_sus chave estrangeira referenciando a chave primária **id\_procedimento\_sus** da tabela **procedimento\_sus**;
- status(**id\_status**, status);
  - **id\_status** chave primária;
- estabelecimento\_de\_saude(**id\_estabelecimento\_de\_saude**, cnes, nome);
  - **id\_estabelecimento\_de\_saude** chave primária;
- entidade(**id**, nome);
  - **id** chave primária;
- auditoria(**id\_auditoria**, data, procedimento\_realizado, usuario\_id\_usuario, tipo\_entidade\_id, id\_entidade\_alvo, procedimento\_id, campo\_antigo, campo\_novo);
  - **id\_auditoria** chave primária;
  - usuario\_id\_usuario chave estrangeira referenciando a chave primária **id\_usuario** da tabela **usuario**;
  - tipo\_entidade\_id chave estrangeira referenciando a chave primária **id\_entidade** da tabela **entidade**;
  - procedimento\_id chave estrangeira referenciando a chave primária **id** da tabela **procedimento**;
- procedimento(**id**, descricao);
  - **id** chave primária;
- config(**id\_config**, nome, descricao, parametro);

- **id\_config** chave primária.

Com o diagrama de Entidade e Relacionamento, fechamos o Documento de Especificação de Projeto com a descrição projeto de componentes.

#### 4.3.3 Projeto de Componentes

Furtado e Costa Jr. (2010) afirmam que esta fase tem como objetivo especificar, detalhadamente, cada componente/módulo do sistema, de modo que seja possível ao programador implementá-lo de forma adequada. Aqui existe uma preocupação com uma boa especificação de componente, ou seja, o programador deve ser capaz de programar o componente, sem a necessidade de procurar o projetista para tirar dúvidas sobre a especificação.

A seguir é demonstrada a especificação do componente NOVO FORMULÁRIO; este é o componente mais complexo da aplicação, pois é responsável pelo algo algoritmo que trata a criação de um formulário e seus diversos módulos, por isso é importante ser detalhado neste projeto. Os demais seguem o padrão de manter itens de dados, já conhecido pelos profissionais da área de desenvolvimento.

1) **FUNÇÃO:** Criar um novo formulário de solicitação de Autorização de Procedimentos Ambulatoriais de Alta Complexidade (APAC).

2) **PSEUDOCÓDIGO DO PROGRAMA NOVO FORMULÁRIO**

1. Início.

Fazer inicializações

Receber número de prontuário ou nome do paciente

localizar prontuário ou nome do paciente em BANCO DE DADOS DO AGHU

SE localizou o paciente ENTÃO

Preenche os dados do paciente no formulário

SENÃO

Exibir a mensagem "Prontuário não cadastrado no BANCO DE DADOS DO AGHU"

FIMSE

2. Receber código do procedimento ou nome do procedimento ambulatorial

localizar procedimento ou nome do procedimento em  
PROCEDIMENTO\_SUS

SE localizou o procedimento ENTÃO

Preenche os dados do procedimento no formulário

SENÃO

Exibir a mensagem "Procedimento não cadastrado no  
banco de dados"

FIMSE

3. Receber código do cid ou nome do CID

localizar código ou nome do cid em CID

SE localizou o cid ENTÃO

Preenche os dados do cid no formulário

SENÃO

Exibir a mensagem "CID não cadastrado no banco de  
dados"

FIMSE

4. Executar verificarPreenchimento

SE todos os campos foram preenchidos de forma correta ENTÃO

Executar Salvar

SE o formulário foi salvo com sucesso em FORMULÁRIO

ENTÃO

Exibe a mensagem "Formulário N° (número do  
formulário) salvo com sucesso"

SENÃO

Exibe a mensagem "Erro ao salvar formulário no  
banco de dados"

FIMSE

SENÃO

Exibir mensagem "Por favor, preencha todos os campos  
corretamente"

FIMSE

5. Finalizações

6. Fim.

verificarPreenchimento

Início.

SE Id\_estabelecimento\_saude\_solicitante = -1 ENTÃO

Exibe a mensagem "Erro - Informe o estabelecimento de saúde  
solicitante!"

Retorna valor Falso

FIMSE

Este documento finaliza esta etapa do processo de desenvolvimento do software. O Documento de Especificação de Projeto se encontra completo no APÊNDICE D. Com todos os documentos produzidos: Documento de Especificação de Requisitos, Documento de Especificação de Análise e Documento de Especificação de Projetos, a aplicação foi implementada. O processo de implementação da ferramenta será apresentada com mais detalhes a seguir.

#### 4.4 Implementação da Ferramenta

O objetivo desta seção é apresentar o processo da implementação do APAC Barros Barreto: planejamento, criação do banco de dados, configuração do ambiente de desenvolvimento e finalização.

##### 4.4.1 Planejamento

Nesta etapa, utilizou-se a metodologia ágil *Scrum*. Este modelo de processo Scrum é recomendado para trabalho em equipe, utilizando: *Product Owner*, *Scrum Master* e o Time de Desenvolvimento. Apesar disto, o projeto foi desenvolvido por duas pessoas somente, que representaram todos os papéis do Time *Scrum*. Os artefatos e os eventos (*Backlog*, *Backlog da Sprint*, *Sprint*, *Revisão da Sprint*) foram elaborados durante este processo de desenvolvimento da aplicação.

Por meio deste planejamento foram definidas as tarefas do produto no *Backlog* do Produto. O Documento de Especificação de Requisitos foi utilizado como fonte para elaboração dessas atividades.

As tarefas foram separadas pelos módulos que compõem o sistema: Controle de Usuários, Registro de Formulários, Registro de Autorização de Formulários e Emissão de Relatórios.

Para o Módulo Controle de usuários foram especificadas as seguintes atividades: cadastramento, edição, desativação ou busca de usuários.

Para o Módulo de Registro de Formulários foram levantadas as seguintes tarefas: definição do algoritmo para criação do formulário APAC, levando em consideração a divisão entre os perfis que podem interagir (de maneiras

diferentes) com o formulário; definição dos status (novo, autorizado, não autorizado, impresso e salvo) que um formulário poderá assumir; definição das regras (conforme descrito no caso de uso) de edição do formulário; criação da área de Registro de Formulários.

Para o Módulo de Registro de Autorização de Formulários foram elencadas as seguintes funções: registro das autorizações (informar os dados da autorização ou não do procedimento requerido); criação da área para o registro da autorização dos formulários APAC.

Para o Módulo de Emissão de formulários foram classificadas as seguintes atividades: definição do algoritmo que fará o quantitativo de CID 10, Setores, Solicitantes e Procedimentos utilizados pela aplicação; criação da área de emissão de relatórios.

As atividades do *Backlog* do Produto foram inseridas em uma *Sprint* mensal, e foram aprimoradas a cada Retrospectiva do *Backlog*. Assim sendo, foi possível ter melhorias mensais do produto. Ao final da última *Sprint*, a aplicação estava pronta para ser disponibilizado para os seus usuários.

O próximo passo do desenvolvimento foi a criação do Banco de Dados do sistema, assunto este que será tratado a seguir.

#### 4.4.2 Banco de Dados

O processo de implementação da aplicação começa com a construção das tabelas do banco de dados, usando como base o Diagrama de Entidade e Relacionamento presente no Documento de Especificação de Projetos. Conforme mostra a Figura 12, o banco de dados foi construído por meio de um *script* em SQL. Neste *script* é apresentado a SQL que gerou a tabela formulário, que representa a entidade principal do sistema.

```

-- Table 'db_formaus_t`.`formulario'
CREATE TABLE IF NOT EXISTS 'db_formaus_t`.`formulario' (
  'id_formulario' INT NOT NULL AUTO_INCREMENT COMMENT '',
  'data' DATE NOT NULL COMMENT '',
  'autenticacao' VARCHAR(300) NOT NULL COMMENT '',
  'paciente_id_paciente' INT(11) NOT NULL COMMENT '',
  'solicitante_id_solicitante' INT(11) NOT NULL COMMENT '',
  'estabelecimento_de_saude_id_solicitante' INT(11) NOT NULL COMMENT '',
  'estabelecimento_de_saude_id_executante' INT(11) NULL COMMENT '',
  'proc_justificativa_id' INT NOT NULL COMMENT '',
  'status_id_status' INT NOT NULL COMMENT '',
  'autorizacao_id_autorizacao' INT NULL COMMENT '',
  PRIMARY KEY ('id_formulario') COMMENT '',
  UNIQUE INDEX 'id_UNIQUE' ('id_formulario' ASC) COMMENT '',
  INDEX 'fk_formulario_pacientel_idx' ('paciente_id_paciente' ASC) COMMENT '',
  INDEX 'fk_formulario_solicitantel_idx' ('solicitante_id_solicitante' ASC) COMMENT '',
  INDEX 'fk_formulario_estabelecimento_de_saude1_idx' ('estabelecimento_de_saude_id_solicitante' ASC) COMMENT '',
  INDEX 'fk_formulario_estabelecimento_de_saude2_idx' ('estabelecimento_de_saude_id_executante' ASC) COMMENT '',
  INDEX 'fk_formulario_proc_justificativa1_idx' ('proc_justificativa_id' ASC) COMMENT '',
  INDEX 'fk_formulario_status1_idx' ('status_id_status' ASC) COMMENT '',
  INDEX 'fk_formulario_autorizacao1_idx' ('autorizacao_id_autorizacao' ASC) COMMENT '',
  CONSTRAINT 'fk_formulario_pacientel'
    FOREIGN KEY ('paciente_id_paciente')
    REFERENCES 'db_formaus_t`.`paciente' ('id_paciente')
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT 'fk_formulario_solicitantel'
    FOREIGN KEY ('solicitante_id_solicitante')
    REFERENCES 'db_formaus_t`.`solicitante' ('id_solicitante')
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT 'fk_formulario_estabelecimento_de_saude1'
    FOREIGN KEY ('estabelecimento_de_saude_id_solicitante')
    REFERENCES 'db_formaus_t`.`estabelecimento_de_saude' ('id_estabelecimento_de_saude')
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT 'fk_formulario_estabelecimento_de_saude2'
    FOREIGN KEY ('estabelecimento_de_saude_id_executante')
    REFERENCES 'db_formaus_t`.`estabelecimento_de_saude' ('id_estabelecimento_de_saude')
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT 'fk_formulario_proc_justificativa1'
    FOREIGN KEY ('proc_justificativa_id')
    REFERENCES 'db_formaus_t`.`proc_justificativa' ('id_proc_justificativa')
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT 'fk_formulario_status1'
    FOREIGN KEY ('status_id_status')
    REFERENCES 'db_formaus_t`.`status' ('id_status')
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT 'fk_formulario_autorizacao1'

```

Figura 12. Script SQL da tabela formulário

A execução do *script* completo gerou as tabelas da Figura 12. Estas tabelas estão apresentadas por meio do *PhpMyAdmin*.



**Figura 13. Tabelas do banco de dados do APAC Barros Barreto**

Na Figura 13, estão todas as tabelas contidas no Diagrama de Entidade e Relacionamento. Visando melhor entendimento, será descrito, de forma breve, o objetivo das principais tabelas da aplicação:

- **auditoria:** tabela responsável por manter as informações do registro de atividades do sistema.
- **autorizacao:** tabela responsável por manter as informações da autorização de um formulário APAC: nome do profissional autorizador, código do órgão emissor, data de autorização, número da autorização.
- **cid:** tabela responsável por manter as informações do CID. Como a descrição do CID e o código.

- **config:** tabela ainda não utilizada, proposta para uma próxima versão da aplicação. Será responsável por manter as informações das configurações do sistema.
- **entidade:** tabela responsável por manter as informações de todas as tabelas do banco de dados.
- **estabelecimento\_de\_saude:** tabela responsável por manter as informações do estabelecimento de saúde solicitante e o estabelecimento executante dos procedimentos solicitados. Como nome do estabelecimento e seu número de Cadastro Nacional de Estabelecimento de Saúde (CNES).
- **formulario:** tabela responsável por manter as informações do formulário APAC.
- **formulario\_has\_procedimento\_sus:** tabela responsável por associar um formulário ao procedimento solicitado.
- **log\_procedimento:** tabela responsável por manter as informações das possíveis ações (inserir, editar, deletar, imprimir e desativar) que podem ser desempenhadas pelo usuário no sistema.
- **paciente:** tabela responsável por manter as informações padrões do paciente. Como nome, telefone, nome do responsável, telefone do responsável, endereço e etc.
- **procedimento\_sus:** tabela responsável por manter as informações dos procedimentos que podem ser solicitados no formulário APAC. Código do procedimento, nome do procedimento e nome abreviado do procedimento.
- **proc\_justificativa:** tabela responsável por manter as informações da justificativa para solicitação de solicitação APAC: As observações da justificativa.
- **setor:** tabela responsável por manter as informações do setor o qual o usuário do sistema está lotado: Nome, sigla do setor.
- **solicitante:** tabela responsável por manter as informações do solicitante de APAC: nome, CNS e CPF.

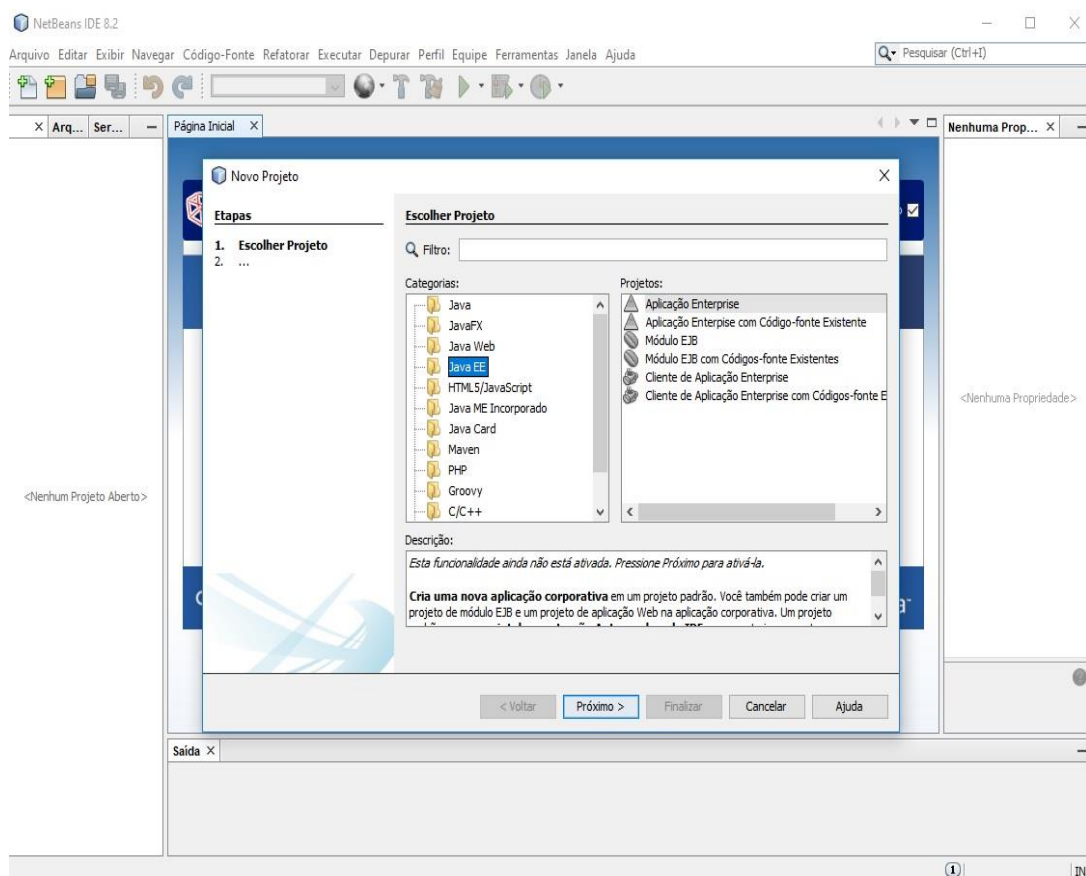
- **status:** tabela responsável por manter as informações dos possíveis status que um formulário pode assumir, isto é, se ele é um novo formulário, se está autorizado ou não autorizado, se foi impresso ou se está salvo.
- **usuario:** tabela responsável por manter as informações do usuário: nome, login, senha, perfil, setor em que o usuário está lotado e se está ativo no sistema ou não.

Após a criação do banco de dados, a próxima etapa é a definição do ambiente de desenvolvimento.

#### 4.4.3 Ambiente de Desenvolvimento

Um ambiente de desenvolvimento pode influenciar diretamente no tempo de desenvolvimento e na produtividade da equipe, então é necessário fazer uma escolha sensata. Na atualidade, o padrão IDE (*Integrated Development Environment* ou Ambiente de Desenvolvimento Integrado) já oferece as opções de identificação da sintaxe por meio de cores, endentação de códigos, *autocomplete*, etc., facilitando o trabalho da equipe.

Para o desenvolvimento do APAC Barros Barreto, foi utilizada a ferramenta *NetBeans* IDE. Por fornecer suporte à implementação de aplicações *Java* para *Web* (Figura 14), que é a linguagem de programação utilizada neste trabalho.



**Figura 14. Ambiente de desenvolvimento do NetBeans.**

O *NetBeans* IDE, proporciona a opção de versionamento de código por meio do plugin Git. Com o auxílio deste plugin é possível retroceder as versões anteriores do código fonte e unir novas e antigas modificações, entre outras funções que facilitam o processo de desenvolvimento.

#### 4.4.4 Finalização do Desenvolvimento

A etapa de finalização consiste na realização de testes e a disponibilização do APAC Barros Barreto para seus usuários. O projeto foi realizado por dois desenvolvedores, então por não contar com uma equipe específica para realização dos testes, foi realizado um teste de validação de seus componentes. Segundo Pressman (2010) *apud* Furtado e Costa Jr. (2010) a validação dos requisitos, ou seja, a aplicação deve cumprir o que prometeu. Nesta etapa, todas as funcionalidades devem estar em conformidade com o Documento de Especificação de Requisitos.

Após a realização do teste e correções de falhas apresentadas, o sistema foi publicado na intranet do Hospital Universitário João de Barros Barreto. Para assegurar que a migração para o ambiente de produção tenha obtido sucesso, foi realizado novamente o teste de validação.

A aplicação encontra-se em funcionamento até a defesa deste trabalho. Para confirmar se o *software* atende as necessidades dos usuários, foi aplicado um formulário *online* de avaliação, assunto que tratado no próximo capítulo.

## 5. RESULTADOS

O objetivo deste Capítulo é apresentar os resultados da avaliação do APAC Barros Barreto. A aplicação *WEB* foi avaliada 16 usuários do próprio hospital, estes foram selecionados 5 do setor de tecnologia e os outros 11 foram profissionais da área de saúde. Os testes foram feitos através de formulário *online* enviado via *e-mail*. O formulário se encontra no APÊNDICE E.

Para garantir que o maior número de falhas seja identificado (e, devidamente, corrigido) antes da implantação definitiva da ferramenta, o número de usuários que utilizaram a aplicação não foi o ideal, devido a necessidade de tempo para o fechamento do trabalho. Nesse contexto foi selecionado apenas um número reduzido de usuários com o perfil necessário para avaliar o sistema.

Durante a avaliação, foram levados em consideração os requisitos de qualidade: usabilidade, funcionalidade, confiabilidade e eficiência. Para os itens de usabilidade, funcionalidade e confiabilidade foram selecionadas duas perguntas. Para o item de eficiência, apenas uma pergunta. Mais à frente os requisitos de qualidades são detalhados, assim como os resultados da avaliação da ferramenta.

### 5.1 Usabilidade

A usabilidade é capacidade do sistema em fazer com que o usuário tenha sucesso na execução de suas tarefas. O usuário precisa entender como a ferramenta funciona e ter facilidade em gravar seus comandos, isto é, na próxima utilização do sistema, ele saberá exatamente o que fazer, pois entendeu facilmente como a aplicação funciona.

As duas perguntas relacionadas à usabilidade foram:

1. Ao acessar o sistema APAC Barros Barreto, é possível identificar o objetivo da ferramenta?
  - a. Sim;
  - b. Sim, porém algumas coisas não ficaram claras;

- c. Não.
2. O APAC Barros Barreto possui uma aparência agradável e intuitiva?
- a. Sim, sua aparência é agradável e consigo entender as suas funcionalidades pelo designer;
  - b. Sim, sua aparência é agradável, mas não consigo entender exatamente as suas funcionalidades;
  - c. Não é agradável, mas é intuitivo;
  - d. Não.

A primeira pergunta tem como objetivo constatar que o sistema é autoexplicativo, ou seja, mesmo sem o auxílio do manual da aplicação, o usuário consegue perceber qual é a sua proposta. A Figura 15 mostra que para 93,3% dos usuários avaliadores, a ferramenta consegue transmitir o seu objetivo. Já 6,7% tiveram dúvidas relacionadas à criação de um novo formulário, não encontravam as opções de busca de prontuário em um primeiro momento. Não houve, entre os usuários avaliadores, casos em que o objetivo da aplicação não foi compreendido.

Usabilidade: Ao acessar o sistema APAC Barros Barreto, é possível identificar o objetivo da ferramenta?

16 respostas

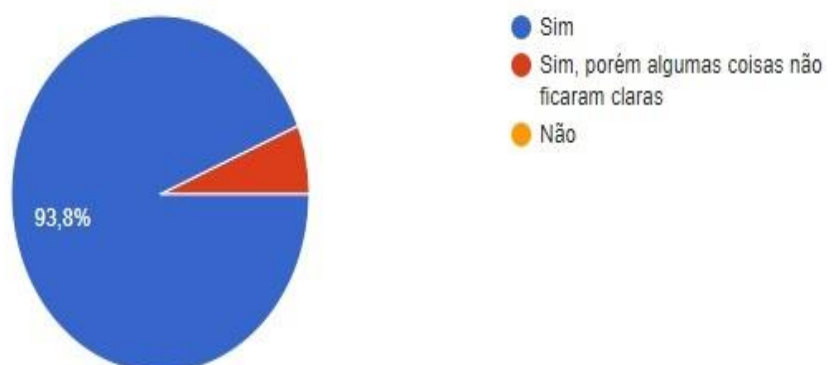


Figura 15. Resultado gráfico da primeira pergunta sobre usabilidade.

A Figura 16 mostra o resultado da segunda pergunta sobre o item de usabilidade. Esta pergunta tinha como objetivo saber do usuário sua opinião sobre a estética do APAC Barros Barreto, ou seja, se a ferramenta é agradável visualmente e, também se possui visual intuitivo em relação a sua proposta. Para 100% dos usuários, a aplicação possui um visual agradável e intuitivo, ou seja, além de ter uma aparência agradável, é possível entender o funcionamento da ferramenta por meio da sua interface.

Os resultados mostram que o APAC Barros Barreto possui uma aparência atrativa em sua utilização e, ao mesmo tempo, consegue ser fácil de usar. A interface da ferramenta é limpa e objetiva.

Usabilidade: O APAC Barros Barreto possui uma aparência agradável e intuitiva?

16 respostas



**Figura 16. Resultado gráfico da segunda pergunta sobre o item usabilidade.**

O APAC Barros Barreto, durante sua avaliação de usabilidade, obteve um bom aproveitamento nos resultados. A ferramenta mostrou-se fácil de usar e de entender. Além de possuir uma interface agradável, o *software* precisa ser funcional. Este item será tratado a seguir.

## 5.2 Funcionalidade

A aplicação deve atender as necessidades para o qual foi projetada. Uma das características esperada pelo APAC Barros Barreto é a capacidade de

criação de um formulário APAC e impressão do mesmo conforme as normas estabelecidas pelo SUS.

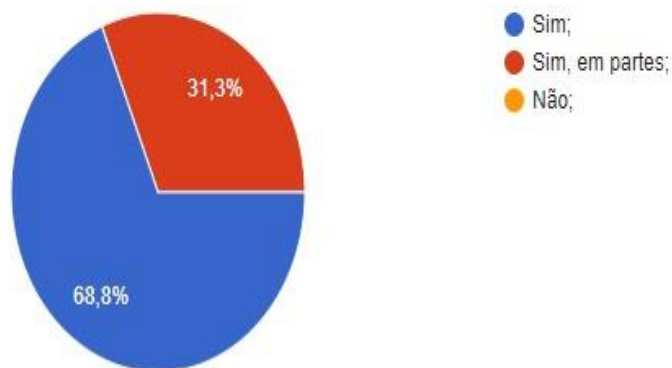
As duas perguntas elaboradas para o item de funcionalidade foram:

1. O APAC Barros Barreto cumpre o que promete fazer, ou seja, fornece a criação de um formulário APAC e imprime no padrão SUS?
  - a. Sim;
  - b. Sim, em partes;
  - c. Não.
  
2. Em algum momento, durante o uso do sistema, ocorreu alguma falha inesperada?
  - a. Nenhuma vez;
  - b. 1 vez;
  - c. 2 vezes;
  - d. Entre 5 e 10 vezes;
  - e. Mais de 10 vezes.

A aplicação foi desenvolvida para cumprir um objetivo, se esta não cumpre o que propõe, se torna uma aplicação inútil. Este é o intuito dessas perguntas. A Figura 17 mostra que para 68,3% dos usuários, o sistema cumpre o que se propõe, ou seja, fornece a criação de formulários APAC e impressão conforme as normas estabelecidas pelo SUS. Para 31,3% dos usuários, o sistema consegue cumprir seu objetivo de certa forma. Esta dúvida se deu pelos usuários em questão não conseguirem efetuar a impressão do formulário, em um primeiro momento não visualização o botão de impressão (visto que ele fica visível somente a partir do momento que o status do formulário muda para salvo). Não houve usuários que não conseguiram entender as funcionalidades do sistema.

Funcionalidade: O APAC Barros Barreto cumpre o que promete fazer, ou seja, fornece a criação de um formulário APAC e imprime no padrão SUS?

16 respostas



**Figura 17. Resultado gráfico da primeira pergunta sobre o item funcionalidade.**

A segunda pergunta busca extrair dos usuários o número de vezes o sistema falhou enquanto estava em uso. Para 81,3% dos usuários, o APAC Barros Barreto não obteve nenhum tipo de erro. Para 18,8% dos usuários, o erro ocorreu apenas uma vez.

O APAC Barros Barreto é uma ferramenta que trabalha com criação e emissão de formulários, para preenchimento destes formulários é necessário buscar os dados no sistema principal do hospital (AGHU). Algumas falhas que podem apresentar estão relacionadas à busca, pois algumas vezes o sistema principal sofre algumas quedas, ocorreram relatos por parte dos usuários que o algum prontuário não foi encontrado na primeira vez ou um procedimento, mas em seguida conseguiram. Este pode ser o motivo responsável pelas porcentagens dos erros que ocorreram.

Funcionalidade: Em algum momento, durante o uso do sistema, ocorreu alguma falha inesperada?

16 respostas

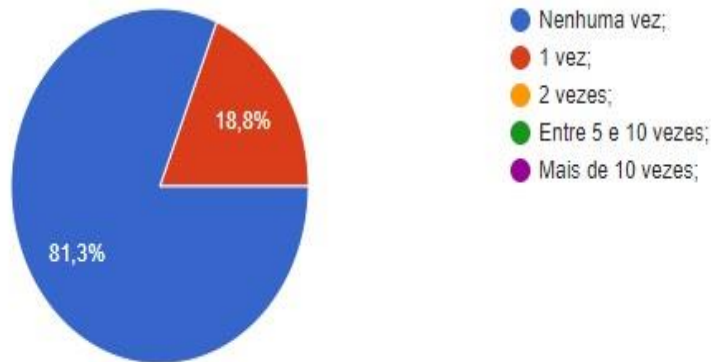


Figura 18. Resultado gráfico da segunda pergunta sobre o item de funcionalidade.

O sistema mostrou-se bastante funcional, obtendo aprovação da maioria dos avaliadores. Além de cumprir o objetivo proposto, a ferramenta apresentou poucos erros e estes, quase que em sua totalidade, se deram por motivos externos a aplicação. Algumas melhorias estão previstas para as próximas versões do sistema. O item que será tratado a seguir será em relação a confiabilidade da ferramenta.

### 5.3 Confiabilidade

A aplicação deve ser capaz de realizar e manter seu funcionamento em circunstâncias normais, bem como em circunstâncias inesperadas. Para extrair a opinião dos usuários em relação à confiabilidade, as seguintes perguntas foram feitas:

1. Todos os botões do APAC Barros Barreto direcionam para páginas corretas?
  - a. Sim;
  - b. Não.
2. O sistema consegue orientar durante sua utilização? Isto é, caso você

não consiga executar determinada tarefa, o sistema indica o que pode estar acontecendo?

- a. Sim;
- b. Algumas vezes, sim;
- c. Não.

A Figura 19 mostra que, para todos os usuários avaliados, os botões do sistema realmente funcionam e direcionam para páginas reais. Em relação a orientação do sistema, podemos ver na Figura 20 que 93,3%% dos usuários acreditam que a ferramenta consegue dar um retorno satisfatório durante o seu uso. Para 6,2% somente às vezes a ferramenta cumpre esse papel, ou seja, em determinado momento da utilização, o usuário não entendeu o motivo de não conseguir prosseguir durante a utilização. Não houve casos em que a ferramenta não conseguiu, de alguma forma, informar o usuário do que estava acontecendo ou não se recuperou de uma falha.

Confiabilidade: Todos os botões do APAC Barros Barreto direcionam para páginas corretas?

16 respostas

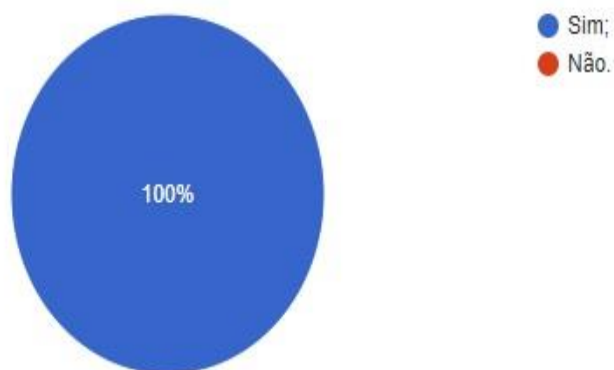
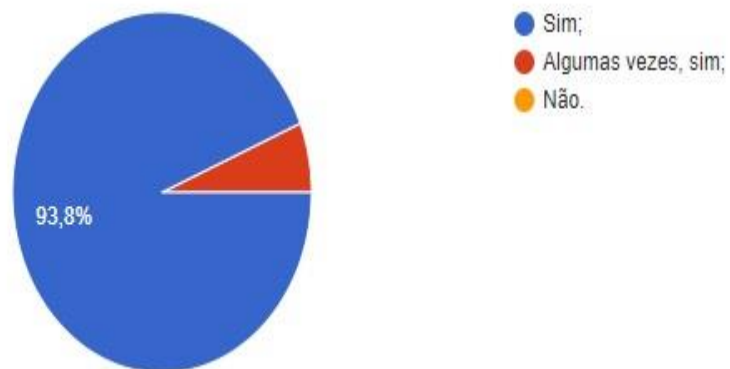


Figura 19. Resultado gráfico da primeira pergunta sobre o item de confiabilidade.

Confiabilidade: O sistema consegue orientar durante sua utilização? Isto é, caso você não consiga executar determinada tarefa, o sistema indica o que pode estar acontecendo?

16 respostas



**Figura 20. Resultado gráfico da segunda pergunta sobre o item de confiabilidade.**

O APAC Barros Barreto conquistou um resultado positivo no quesito confiabilidade. Além de ser funcional e confiável, a ferramenta precisa ser eficiente. Este item é tratado a seguir.

#### 5.4 Eficiência

Sommerville (2011, p. 5) diz, em relação à eficiência: “O software não deve desperdiçar os recursos do sistema, como memória e ciclos do processador. Portanto, eficiência inclui a capacidade de resposta, tempo de processamento, uso de memória, etc.”.

Em relação à eficiência foi elaborada uma pergunta:

1. O APAC Barros Barreto consegue processar as requisições com tempo de resposta aceitável?
  - a. Sim, estou satisfeito com o tempo de resposta;
  - b. Ele demora um pouco com algumas requisições;
  - c. Não, demora bastante;

O resultado foi totalmente satisfatório. Para 93,8% dos avaliadores, a

ferramenta possui um ótimo tempo de resposta com suas requisições. Os 6,2% que informaram que algumas requisições demoraram um pouco, podem estar relacionados com suas máquinas específicas, isto pode ter afetado o desempenho da aplicação.

### Eficiência: O APAC Barros Barreto consegue processar as requisições com tempo de resposta aceitável?

16 respostas



**Figura 21. Resultado gráfico da pergunta sobre o item de eficiência.**

De modo geral, pode-se concluir que, através da avaliação realizada, a ferramenta foi bem desenvolvida, atendendo devidamente aos objetivos propostos inicialmente. Esta avaliação foi importante, pois com ela pode-se registrar todos os *feedbacks* e refletir melhorias para o próximo incremento do sistema.

Este Capítulo fecha este trabalho. Através da avaliação foi possível fechar todas as etapas que compõem o processo de desenvolvimento. O APAC Barros Barreto foi concebido através das documentações de software, codificado, testado e, por fim, avaliado.

## 6. CONCLUSÃO

Os profissionais do Hospital Universitário João de Barros Barreto emitem solicitações de Autorização de Procedimento Ambulatorial de Alta complexidade por meio de um antiquado processo manual, tornando-o cansativo e passível a erros. Todo o desenvolvimento do projeto teve por objetivo elaborar um sistema informatizado para solucionar este problema e agregar como diferencial a funcionalidade emissão do formulário impresso já no modelo padronizado pelo Sistema Único de Saúde.

Utilizamos para isto metodologias e modelos de engenharia de software, aplicando conceitos que abrangem todos os semestres do curso. Por meio das técnicas de levantamento de requisitos, abstraiu-se uma boa percepção do problema e, a partir daí, desenvolvemos todos os artefatos que levaram a codificação do sistema.

O projeto como um todo adicionou grande experiência aos dois membros da equipe. A principal razão para isto é que, estávamos focados e envolvidos em todas as atividades desenvolvidas. Com o levantamento dos requisitos do sistema, colocamos em prática os conceitos aprendidos sobre análise de sistemas. Aplicaram-se as metodologias, o desenvolvimento dos diagramas e a codificação do sistema baseados nas habilidades adquiridas em sala de aula. A oportunidade de colocar todos estes conhecimentos em prática de uma única vez foi muito satisfatória.

### 6.1 Trabalhos Futuros

Em função do período de aceitabilidade do sistema ainda estar em andamento, recomenda-se para trabalhos futuros: a integralização com Hospital Universitário Bettina Ferro de Souza com o intuito de atender a necessidade de ambos os complexos e outros modelos de formulários, estes devem atender a área de clínica médica, área cirúrgica, clínica fisioterapeuta, laboratório de exames, assim como seus respectivos indicadores de produção hospitalar.

## REFERÊNCIAS

BOOCH, Grady et al. **UML: Guia do usuário**. 6. ed. Rio de Janeiro: Elsevier, 2006.

CAELUM. **Apostila-java**. Disponível em: <https://www.caelum.com.br/apostila-javaweb/>, acessado em 5 de Janeiro de 2018.

FONSECA, Isabella. **Engenharia de Software Conference**. São Paulo: DevMedia, 2009.

FURTADO, Alfredo Braga; COSTA JR., Júlio Valente da. **Prática de Análise e Projeto de Sistemas**. 1 ed. Belém: abfurtado.com.br, 2010.

GONÇALVES, E. **AJAX na prática: Todo o poder dos melhores frameworks AJAX aliados ao desenvolvimento WEB 2.0**. Rio de Janeiro: Ciência Moderna, 2007.

PRESSMAN, Roger S. **Uma Abordagem Profissional**. 7. ed. São Paulo: Pearson Books, 2011.

MACHADO, Marcos; MEDINA, Sérgio Gustavo. **SCRUM – Método Ágil: uma mudança cultural na Gestão de Projetos de Desenvolvimento de Software**. Revista Científica Intr@ciência, UNIESP, 2017.

SANTOS, R. **Java na Web**. Axcel Books do Brasil Editora Ltda., 2010,

SOMMERVILLE, Ian. **Engenharia de Software**. 9. ed. São Paulo: Pearson Prentice Hall, 2011.

SOUZA, Maykon Araújo de. **Concepção, Especificação, Implementação e Avaliação de Aplicação Web para Prestadores de Serviço**. Universidade Federal do Pará. 2017.

**Usando Suporte Git no NetBeans IDE.** Disponível em [https://netbeans.org/kb/docs/ide/git\\_pt\\_BR.html](https://netbeans.org/kb/docs/ide/git_pt_BR.html). Acesso em 10 de Janeiro de 2018.

**APÊNDICE A – MODELO PADRÃO DO LAUDO MÉDICO PARA  
SOLICITAÇÃO DE PROCEDIMENTO AMBULATORIAL DE ALTA  
COMPLEXIDADE**



Estado de Santa Catarina  
Ministério da Saúde

### LAUDO MÉDICO PARA PROCEDIMENTOS DE ALTA COMPLEXIDADE - APAC

#### IDENTIFICAÇÃO DO ESTABELECIMENTO DE SAÚDE (SOLICITANTE)

1 - NOME DO ESTABELECIMENTO DE SAÚDE SOLICITANTE	2 - CNES
--	----------

#### IDENTIFICAÇÃO DO PACIENTE

3 - NOME DO PACIENTE	4 - Nº DO PRONTUÁRIO		
5 - CARTÃO NACIONAL DE SAÚDE (CNS)	6 - DATA DE NASCIMENTO	7 - SEXO Masc. <input type="checkbox"/> Fem. <input type="checkbox"/>	8 - RAÇA/COR
9 - NOME DA MÃE	10 - TELEFONE DE CONTATO Nº DO TELEFONE DDD		
11 - NOME DO RESPONSÁVEL	12 - TELEFONE DE CONTATO Nº DO TELEFONE DDD		
13 - ENDEREÇO (RUA, Nº, BAIRRO)			
14 - MUNICÍPIO DE RESIDÊNCIA	15 - Cód. IBGE MUNICÍPIO	16 - UF	17 - CEP

#### PROCEDIMENTO SOLICITADO

CÓDIGO DO PROCEDIMENTO	NOME DO PROCEDIMENTO	QTDE
CÓDIGO DO PROCEDIMENTO	NOME DO PROCEDIMENTO	QTDE
CÓDIGO DO PROCEDIMENTO	NOME DO PROCEDIMENTO	QTDE

#### JUSTIFICATIVA DO(S) PROCEDIMENTO(S) SOLICITADO(S)

DESCRIÇÃO DO DIAGNÓSTICO	CID 10 PRINCIPAL	CID 10 SECUNDÁRIO	CID 10 CAUSAS ASSOCIADAS
RESUMO DA ANAMNESE E EXAME FÍSICO			
EXAMES COMPLEMENTARES REALIZADOS			
JUSTIFICATIVA DO PROCEDIMENTO			

#### SOLICITAÇÃO

NOME DO PROFISSIONAL SOLICITANTE	DATA DA SOLICITAÇÃO	ASSINATURA E CARIMBO
DOCUMENTO ( ) CNS ( ) CPF	NÚMERO DO DOCUMENTO (CNS/CPF)	

#### AUTORIZAÇÃO

NOME DO PROFISSIONAL AUTORIZADOR	COD ORGÃO EMISSOR	NÚMERO DA AUTORIZAÇÃO (APAC)
DOCUMENTO ( ) CNS ( ) CPF	NÚMERO DO DOCUMENTO (CNS/CPF)	
DATA DA AUTORIZAÇÃO	ASSINATURA E CARIMBO	PERÍODO DE VALIDADE DA APAC a

#### IDENTIFICAÇÃO DO ESTABELECIMENTO DE SAÚDE (EXECUTANTE)

NOME DO ESTABELECIMENTO DE SAÚDE EXECUTANTE	CNES
---	------

VeriDocument

**APÊNDICE B – DOCUMENTO DE ESPECIFICAÇÃO DE REQUISITOS DO  
APAC BARROS BARRETO**

# Documento de Especificação de Requisitos de Software do APAC Barros Barreto

## Sistema APAC Barros Barreto

### Documento de Especificação de Requisitos

#### 1. Introdução

Este documento contém a especificação de requisitos para o sistema de autorização de procedimentos do Hospital Universitário João de Barros Barreto, com a utilização de técnica de modelagem de Casos de Uso. O sistema possibilita o preenchimento e a impressão, de forma semiautomática, de formulários de autorização de procedimentos ambulatoriais no padrão SUS e a emissão de relatórios quantitativos.

#### 2. Ambiente do Sistema.

O sistema a ser desenvolvido será utilizado no Hospital Universitário João de Barros Barreto (HUJBB). Considera-se um usuário responsável pela realização de cadastro de usuários no sistema: criação, edição e desativação dos usuários que utilizarão a aplicação. Neste trabalho, esse usuário é identificado com o perfil de **Administrador**.

No momento da consulta do paciente, o usuário utilizador da aplicação, preenche as informações de um novo formulário de autorização de procedimentos, de forma semiautomática, fornecendo para isto os dados de identificação do paciente, do procedimento solicitado, procedimentos secundários (caso necessário) e justificativa. Neste trabalho, esse usuário é identificado com o perfil de **Solicitante**.

Após o preenchimento de todas as informações o sistema gera um documento no padrão de formulário do SUS para impressão.

Após o formulário, ao ser impresso, é encaminhado ao órgão executor para autorização, em seguida ele é reenviado ao hospital contendo a informação se foi autorizado ou não. Após o recebimento do formulário, o usuário do sistema deve informar estas informações no sistema, elas serão incluídas no formulário eletrônico,

na aplicação e a informação será registrada no sistema. A este usuário é identificado o perfil de **Administrativo** na aplicação.

Periodicamente o usuário, identificado com o perfil de **Administrativo Financeiro**, poderá consultar a aplicação e verificar o quantitativo dos formulários autorizados e não autorizados, de acordo com os critérios de CID 10 (Código Internacional de Doenças), Setor, Solicitante ou procedimento, para fins de cálculos nas contas médicas do hospital.

### **3. Objetivos do produto**

Desenvolver um sistema que preencha formulários de procedimentos ambulatoriais de forma semiautomática, possibilitando sua impressão conforme as normas de Autorização para Procedimentos de Alta Complexidade (APAC), conjuntamente permitindo a emissão de relatórios relacionados aos procedimentos solicitados, códigos das doenças diagnosticadas, setores responsáveis pelo preenchimento dos relatórios e usuários da aplicação.

### **4. Benefícios do produto**

As seguintes funcionalidades serão consideradas no sistema a ser desenvolvido:

- Controle do cadastro de utilizadores (inclusão, edição, exclusão e consulta);
- Preenchimento de forma semiautomática e emissão de formulários conforme as normas de Autorização para procedimentos de Alta Complexidade (APAC) do SUS (Sistema Único de Saúde);
- Emissão do quantitativo relacionado aos procedimentos utilizados, doenças diagnosticadas, setores e usuários do sistema;
- Registro das principais atividades desempenhadas pelos usuários do sistema.

### **5. Modelo de Casos de Uso**

Existem 4 atores interagindo com o sistema, todos são usuários, porém cada um tem seu perfil identificado pelo sistema: ator Administrador, ator Solicitante, ator Administrativo e ator Administrativo Financeiro.

- **Ator Administrativo Financeiro:** usuário do sistema com perfil Administrativo Financeiro responsável pela emissão dos relatórios da aplicação.
- **Ator Administrativo:** usuário do sistema com perfil Administrativo responsável por registrar se o formulário de procedimentos ambulatoriais fora autorizado ou não, caso sim, informando os dados da autorização.
- **Ator Solicitante:** usuário do sistema com o perfil Solicitante responsável pelo preenchimento e impressão de um novo formulário de procedimento ambulatorial.
- **Ator Administrador:** usuário do sistema com o perfil Administrador responsável pelo controle de usuários da aplicação (inclusão, edição, desativação e consulta) e registros das atividades desempenhadas por eles. Além disso, o usuário com perfil de Administrador pode desempenhar todas as funções de qualquer outro perfil descrito no acima.

Os casos de uso identificados correspondem a subsistemas que serão decompostos em outros diagramas, conforme verifica a seguir (ver Figura 1).

**Controle de Usuários:** envolve as funções do usuário com perfil de Administrador, ou seja, o cadastramento, edição, desativação ou busca de usuários.

**Registro de Formulários:** envolve funções dos usuários com perfil de Solicitante ou Administrativo. Caso o usuário seja perfil Solicitante, suas funções serão as de cadastrar, editar (seguindo as suas regras de edição conforme descrito no seu respectivo caso de uso) e buscar formulários de autorização de procedimentos. Caso o usuário possua o perfil de Administrativo, possuirá a apenas a função de edição (seguindo suas regras de edição conforme descrito no seu respectivo caso de uso) de formulários de autorização de procedimentos.

**Registro de Autorização de Formulários:** envolve funções do usuário com perfil Administrativo, fazendo o registro de autorização dos formulários de autorização de procedimentos.

**Emissão de Relatórios:** envolve as funções do usuário com perfil Administrativo Financeiro, emitindo o quantitativo (CID 10, Setor, Solicitante, Procedimento) de formulários autorizados e não autorizados pelo SUS.

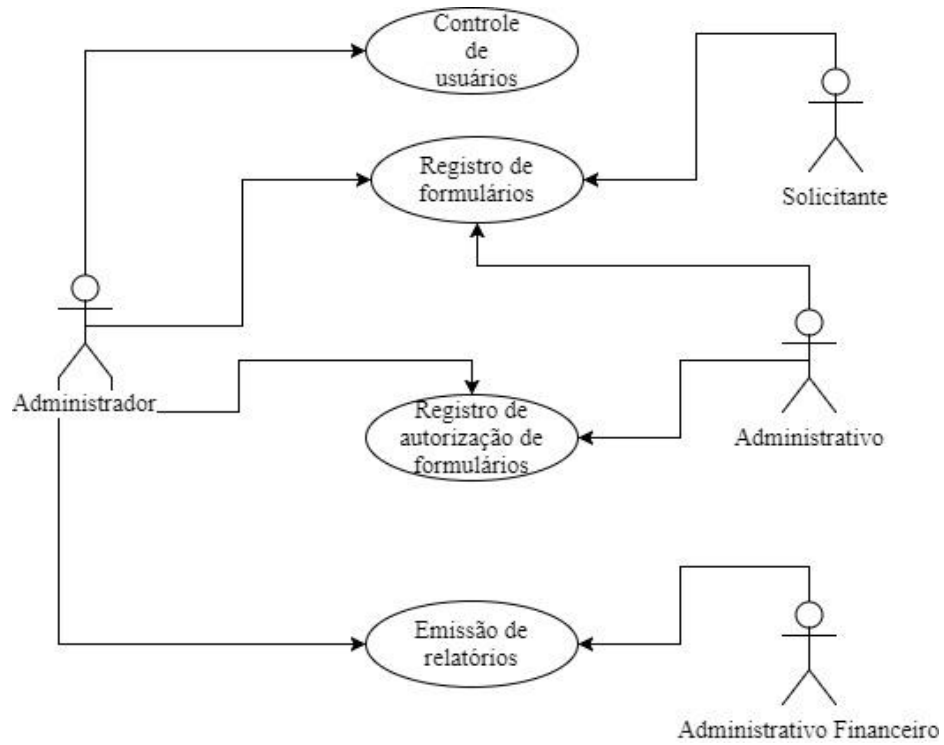
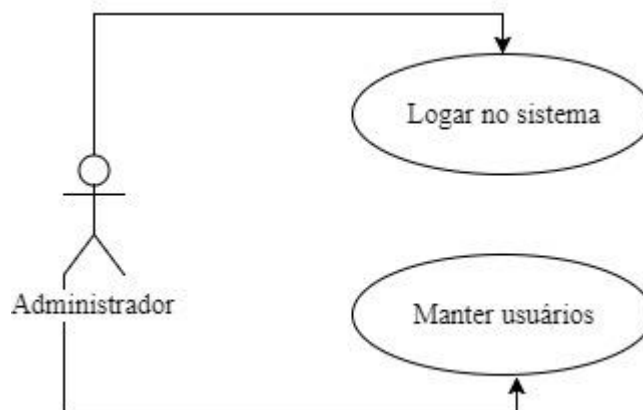


Figura 1. Diagrama de Casos de Uso de Contexto.

## 5.1 Subsistema Controle de Usuários



**Figura 2. Diagrama de Casos de Uso do Subsistema Controle de Usuários.**

### **CASO DE USO 01: LOGAR NO SISTEMA**

**Descrição:** O usuário deverá entrar com login e senha e o sistema deverá validar suas credenciais permitindo o acesso ou não ao sistema.

**Atores primários:** usuário com perfil Administrador, Solicitante, Administrativo ou Administrativo Financeiro.

**Precondições:** o usuário deve possuir cadastro no AD (Active Directory) do Hospital Universitário João de Barros Barreto e no banco de dados da aplicação.

#### **Fluxo Principal:**

1. O usuário deverá entrar com as credenciais de login e senha pré-cadastradas.
2. O sistema processará as informações e validar o login.
3. Caso o login seja válido, o usuário entra no sistema.
4. Caso o login seja inválido o usuário não terá acesso ao sistema.

**Pós-Condições:** nenhuma.

**Prioridade de Desenvolvimento:** 1.

**Frequência de uso:** diário.

### **CASO DE USO 02: MANTER USUÁRIOS.**

**Descrição:** Este caso de uso permitirá criação, edição, busca e desativação de usuários do sistema.

**Ator primário:** usuário do sistema com o perfil de Administrador.

**Precondições:** O usuário deverá estar logado com o perfil de Administrador.

**Fluxo principal:**

1. O usuário seleciona a opção <<Usuário>> no menu principal.
2. O sistema exibe as opções que podem ser selecionadas: Cadastrar, editar e buscar usuários.
3. O usuário indica a opção desejada: Cadastrar, buscar e editar.
4. Se o usuário desejar continuar com alguma operação, o caso de uso retorna para o passo 2, caso contrário o caso de uso termina.

**Fluxo alternativo (3):** Cadastrar um novo usuário no sistema.

- a. O usuário solicita o cadastro de um novo usuário através da opção <<Cadastrar>>.
- b. O sistema exibe uma tela com as informações do usuário a serem preenchidas (Nome, CPF, Login, Senha, CNS, Setor, Perfil).
- c. O usuário preenche as informações e seleciona a opção <<Finalizar>>.
- d. O sistema verifica se o login e senha são compatíveis com as mesmas credenciais do AD (Active Directory) do HUUJBB ou HUBFS.
- e. Se forem válidos, sistema efetua o cadastro do novo usuário no sistema caso contrário, exibe uma mensagem de falha.

**Pós-Condições:** O sistema exibe uma mensagem de cadastro realizado com sucesso ou falha.

**Prioridade de Desenvolvimento:** 2

**Frequência de Uso:** Eventual.

**Fluxo alternativo (3):** Busca e edição de usuários no sistema.

- a. O usuário solicita a busca e edição de usuários através da opção <<Buscar / Editar>>.
- b. O sistema exibe uma tela com os campos para informar o nome do usuário que deseja fazer a busca e também exibe uma lista com o nome, login e opções (Visualizar, editar e desativar) de todos os usuários cadastrados no sistema por ordem alfabética.
- c. Caso o usuário deseje fazer a busca, ele deverá preencher o campo de Busca de usuários
- d. O sistema retornará uma lista com usuários que contenham as mesmas informações prescritas no campo de busca.
- e. Caso o usuário deseje visualizar, editar ou desativar um usuário do sistema, deverá selecionar uma das opções ao lado do nome do usuário.

**Pós-condições:** nenhuma.

**Prioridade de Desenvolvimento:** 2

**Frequência de uso:** Eventual.

## **5.2 Subsistema Registro de Formulários**

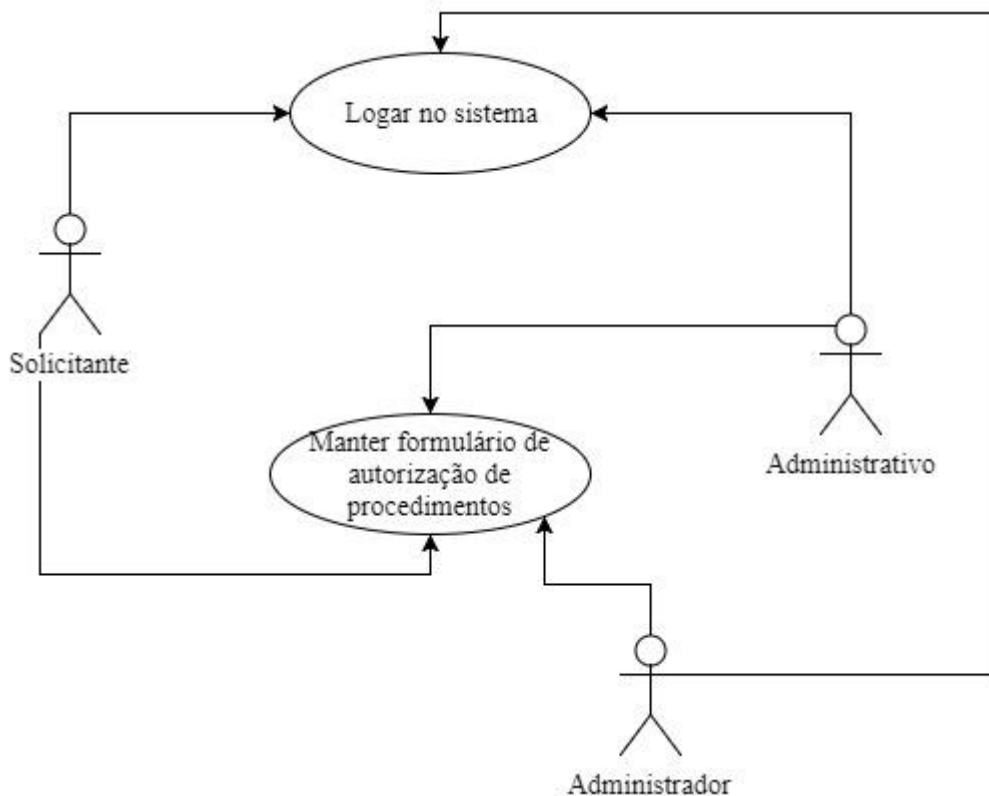


Figura 3. Diagrama de Casos de Uso do Subsistema Registro de Formulários

### CASO DE USO 03: MANTER FORMULÁRIO DE AUTORIZAÇÃO DE PROCEDIMENTOS.

**Descrição:** O usuário com perfil de Solicitante mantém o cadastro de um novo formulário de autorização de procedimento (Inclusão, edição e consulta de formulários).

**Atores primários:** Solicitante, Administrativo

**Precondições:** o usuário deverá estar logado no sistema com perfil de Solicitante.

#### Fluxo Principal:

1. O usuário seleciona a opção <<Formulário>> no menu principal.
2. O sistema exibe as opções que podem ser selecionadas: A inclusão de um novo formulário, a edição de um formulário e a consulta de formulários.
3. O usuário indica a opção desejada: Inclusão, edição ou consulta.

4. Se o usuário desejar continuar com alguma operação, o caso de uso retorna para o passo 2, caso contrário o caso de uso termina.

**Fluxo alternativo (3):** inclusão de novo formulário de autorização de procedimento.

- a O usuário solicita um novo formulário através da opção <<Novo>>.
- b O sistema exibe um formulário de entrada das informações do Laudo para solicitação/Autorização de procedimento Ambulatorial no padrão do Sistema Único de Saúde (SUS).
- c O usuário preenche, de forma semiautomática, os dados do formulário ( Os campos de Identificação do Paciente, Procedimento Solicitado, Procedimento(s) Secundários(s) e Justificativa do(s) Procedimento(s) Solicitado(s).
- d Após o preenchimento dos campos do formulário, o usuário terá duas opções: Limpar e Salvar.
- e Caso o usuário opte por Limpar, o sistema apaga todos os dados fornecidos e o fluxo retorna para o passo b.
- f Caso o usuário opte por Salvar, o sistema verifica se todas as informações obrigatórias foram fornecidas

Se sim, o sistema persiste as informações no banco de dados exibindo a mensagem de sucesso e o número de identificação do formulário gerado pelo banco de dados,

Senão, o sistema exibe a mensagem de erro, informando qual informação precisa ser preenchida, fimse.

**Pós-Condições:** O sistema exibe uma mensagem de sucesso ou falha na operação.

**Prioridade de desenvolvimento:** 3

**Frequência de uso:** diário

**Fluxo alternativo (3):** Buscar formulário de autorização de procedimento.

- a. O usuário solicita a busca de formulário através da opção <<Buscar>>.
  - b. O sistema exibe uma tela com a opção <<Buscar formulário>>.
  - c. O usuário preenche o campo com o número de identificação do formulário desejado.
  - d. O sistema verifica se o número de identificação do formulário é válido,
  - e. Se sim, o sistema exibe as informações do formulário.
- Senão, o sistema exibe uma mensagem informando que o formulário correspondente ao número de identificação não foi encontrado, fimse.

**Pós-Condições:** nenhuma.

**Prioridade de Desenvolvimento:** 1.

**Frequência de uso:** eventual

**Fluxo alternativo (3):** Editar formulário de autorização de procedimento.

- a. O usuário solicita a edição de um formulário através da opção <<Editar>>.
- b. O sistema exibe uma tela com a opção <<Buscar formulário>>.
- c. O usuário preenche o campo com o número de identificação do formulário que deseja editar.
- d. O sistema verifica se existe um formulário com o número de identificação informado.
- e. O sistema verifica o perfil do usuário.
- f. O sistema verifica o status do formulário.
- g. Caso o perfil de Solicitante seja identificado,  
Se o status do formulário for como Novo ou Salvo, o sistema exibe todas os campos do formulário, porém permitindo apenas a edição das seções de Identificação do Paciente, Procedimento Solicitado, Procedimentos Secundários, Justificativa Do(s) Procedimento(s) Solicitado(s) do formulário,  
Senão, o sistema exibe apenas as informações do formulário, mas não permite edição, fimse.
- h. Caso o perfil de Administrativo seja identificado,

Se o status do formulário for como Autorizado ou Não Autorizado, o sistema exibe todas os campos do formulário, permitindo apenas a edição da seção Autorização do formulário,

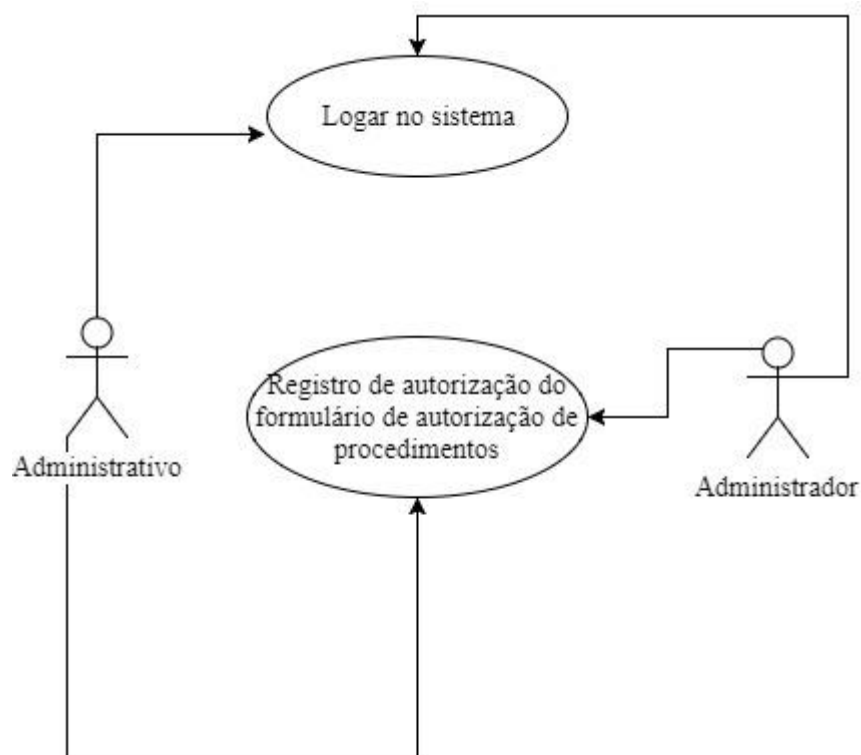
Senão, o sistema exibe apenas as informações do formulário, mas não permite edição, fimse.

**Pós-Condições:** o sistema exibe a mensagem de dados atualizados.

**Prioridade de Desenvolvimento:** 2

**Frequência de uso:** eventual.

### 5.3 Subsistema Registro de Autorização de Formulários



**Figura 4. Diagrama de Casos de Uso do Subsistema Registro de Autorização de Formulários de Procedimentos**

## **CASO DE USO 04: REGISTRO DE AUTORIZAÇÃO DO FORMULÁRIO DE AUTORIZAÇÃO DE PROCEDIMENTOS.**

**Descrição:** O usuário com perfil Administrativo informa os dados da autorização de procedimentos (Campo Autorização).

**Ator primário:** Administrativo

**Precondições:** o usuário deverá estar logado no sistema com perfil de Administrativo.

**Fluxo principal:**

1. O usuário seleciona a opção <<Formulário>> no menu principal.
2. O sistema exibe a opção que pode ser selecionada: Registrar Autorização.
3. O sistema exibe uma tela com a opção <<Buscar formulário>>.
4. O usuário preenche o campo o número de identificação do formulário que deseja fazer a autorização.
5. O sistema verifica se existe um formulário com o número de identificação informado.
6. O sistema verifica o status do formulário,  
Se o status for como Impresso ou Enviado para o SUS, o sistema exibe as opções (Se de Autorização) do formulário para o preenchimento,  
Senão, o sistema exibe apenas as informações do formulário, mas não permite registro de autorização, fimse.
7. O usuário preenche os campos da autorização, onde o campo N° de autorização (APAC) é o único campo obrigatório.

**Pós-condições:** o sistema exibe a mensagem de autorização registrada com sucesso.

**Prioridade de Desenvolvimento:** 3

Frequência de uso: diário.

#### 5.4 Subsistema Emissão de Relatórios.

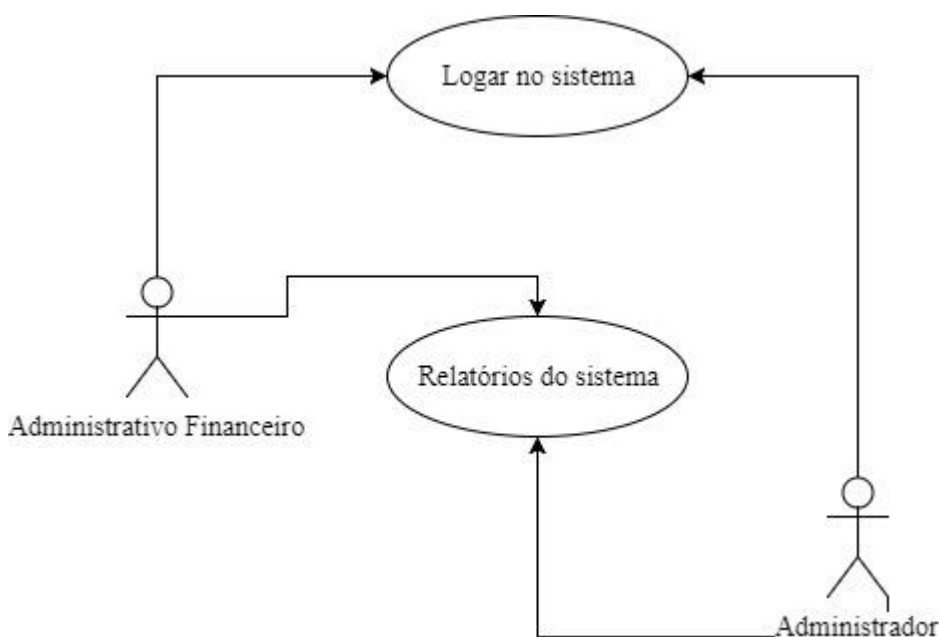


Figura 5. Diagrama de Casos de Uso do Subsistema Relatórios do Sistema.

#### CASO DE USO 05: RELATÓRIOS DO SISTEMA.

**Descrição:** Este caso de uso permitirá que o sistema forneça o quantitativo (CID 10, Setor, Solicitante, Procedimento) de formulários autorizados e não autorizados pelo SUS.

**Ator Primário:** usuário com perfil Administrativo Financeiro.

**Precondições:** o usuário deverá estar logado com o perfil Administrativo Financeiro.

#### Fluxo Principal:

1. O usuário seleciona a opção <<Relatórios>> no menu principal.
2. O sistema exibe uma tela com as opções de relatórios que podem ser selecionadas: CID 10, Setor, Solicitante e Procedimento.
3. O usuário indica a opção desejada: CID 10, Setor, Solicitante e Procedimento.

4. Se o usuário desejar continuar com alguma operação, o caso de uso retorna para o passo 2, caso contrário o caso de uso termina.

**Pós-Condições:** nenhuma.

**Frequência de Uso:** eventual.

**Prioridade de Desenvolvimento:** 1.

## **6. Requisitos Não Funcionais**

**6.1 Segurança:** O sistema deve prover facilidade para autenticação dos usuários, com a digitação de senha para acesso, o usuário deve possuir cadastro no AD ( Active Directory ) do Hospital Universitário João de Barros Barreto e no banco de dados da aplicação.

**6.2 Desempenhos:** Todas as operações deverão ser executadas em tempo hábil para otimização do tempo de atendimento ao paciente.

**6.3 Interfaces:** A interface do sistema deverá obedecer a exigência de boa usabilidade, já que os usuários precisam realizar suas atividades de forma rápida e prática.

**6.4 Portabilidades:** O sistema deve permitir ampla portabilidade, de modo a executar nos principais navegadores: Chrome, FireFox, EDGE e Opera.

## **7. Restrições do Software**

O software a ser desenvolvido deverá rodar em qualquer ambiente; a linguagem de programação escolhida para a implementação deve garantir código eficiente e compacto, o banco de dados (relacional) a ser utilizado deve ser MySQL 5.2.

**APÊNDICE C – DOCUMENTO DE ESPECIFICAÇÃO DE ANÁLISE DO APAC  
BARROS BARRETO**

# DOCUMENTO DE ESPECIFICAÇÃO DE ANÁLISE – APAC BARROS BARRETO

## 1. Introdução

Este documento contém a especificação de análise do Sistema APAC Barros Barreto (Sistema de Solicitação de Autorização de Procedimentos Ambulatoriais de Alta Complexidade). Abrange o modelo de classes (diagrama de pacotes, de classes e dicionário de dados) e o modelo comportamental (diagrama de sequência).

## 2. Modelo de Classes

Este modelo é útil para a identificação de classes (e seu agrupamento em pacotes), de relacionamentos, de atributos e de operações.

Nem sempre é possível identificar todas essas características nesse ponto da análise, sendo necessário recorrer ao modelo comportamental para complementar a especificação. Os resultados obtidos nessas duas fases, no entanto, encontram-se condensados nos diagramas apresentados adiante.

### 2.1 Diagrama de Pacotes

Este diagrama objetiva garantir uma visualização de mais alto nível do sistema provendo um agrupamento lógico das classes.

Para que se alcance o objetivo proposto, deve-se partir da análise do problema e da solução detalhada nos casos de uso. Abaixo encontra-se o diagrama de pacotes do Sistema APAC Barros Barreto, que possui os subsistemas Controle de Usuários, Registro de Formulários, Registro de Autorização de Formulários e Emissão de Relatórios, subsistemas que estão distribuídos nos pacotes de Aplicação, view, bean, dao, entidades, util, util.execao e entidades.relatorios. A linha pontilhada denota a dependência dos subsistemas.

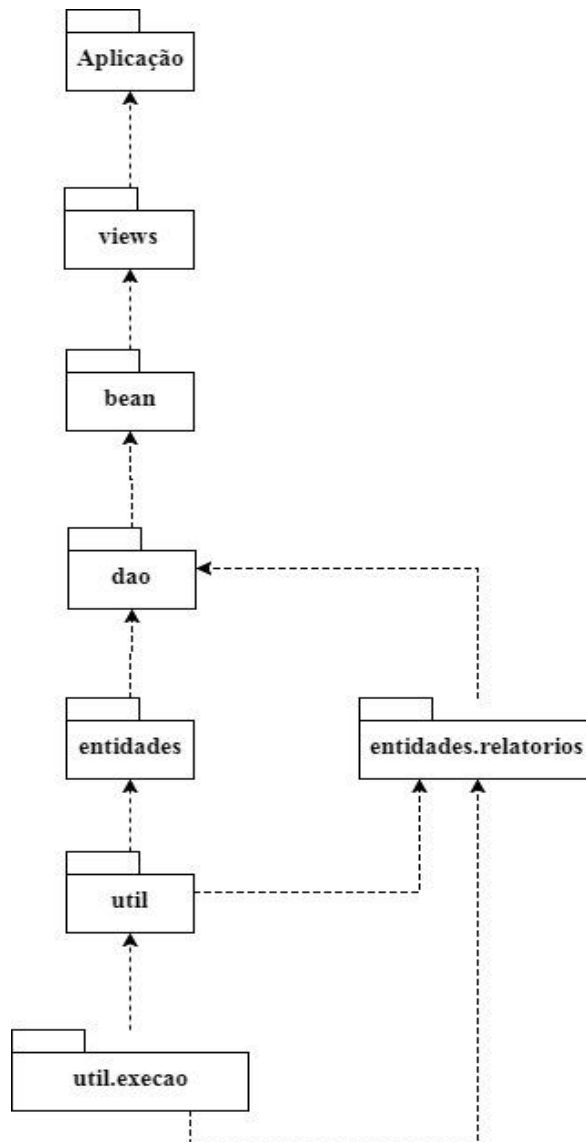


Figura 1. Diagrama de Pacotes do APAC Barros Barreto

## 2.2 Diagrama de Classes

A figura abaixo mostra o diagrama de classes geral referente aos pacotes descritos no diagrama acima. Em seguida estão as especificações de cada método utilizado no diagrama.



+Conjunto de métodos CRUD.

### **Cid**

+Conjunto de métodos CRUD.

### **Entidade**

+Conjunto de métodos CRUD.

### **Estabelecimento\_de\_saude**

+Conjunto de métodos CRUD.

### **Formulario\_has\_procedimento\_sus**

+Conjunto de métodos CRUD.

### **Formulario**

+Conjunto de métodos CRUD.

+buscaCid(cid : String) : List(cid)

+buscaForm(id : String) : formulario

+buscaProcedimento(busca : String) : procedimento\_sus

+buscaProntuarioAghu(n\_prontuario : String) : paciente

+verificaPerfil(perfil : int) : boolean

+log(logado : Usuario, acao : int) : void

### **Paciente**

+Conjunto de métodos CRUD.

### **Proc\_justificativa**

+Conjunto de métodos CRUD.

### **Procedimento**

+Conjunto de métodos CRUD.

### **Procedimentos\_sus**

+Conjunto de métodos CRUD.

### **Solicitante**

+Conjunto de métodos CRUD.

+buscaAtivos() : void

+log(logado: usuario, acao: int): void

+verificaPerfil(perfil: int): void

+verificarSolicitanteCadastrado() : boolean

### **Status**

+Conjunto de métodos CRUD.

### **Usuario**

+logar(void) : boolean

+logarBDLocal(void) : boolean

+sair(void) : void

+verificarUsuLog(void) : boolean

## **2.4 Dicionário de Dados**

altura: \* Altura do paciente \*

Tipo: Float

**AUDITORIA** = {auditoria}

**Auditoria** = \*registro das atividades realizadas no sistema\*

@id\_auditoria + data + procedimento\_realizado + usuario\_id\_usuario +  
tipo\_entidade + id\_entidade\_alvo + procedimento\_id + campo\_antigo + campo\_novo

autenticacao: \* Autenticação gerada pelo sistema do formulário de solicitação APAC

\*

Tipo: String(300)

**AUTORIZACAO = {autorizacao}**

**Autorizacao** = \*registro de dados da autorizacao do formulario\*

@id\_autorizacao + autorizador + cod\_org\_emissor + tipo\_doc + num\_doc + data +  
num\_autorizacao + data\_val\_ini + data\_val\_fim

autorizacao\_id\_autorizacao: \* Código da autorizacao do formulário de solicitação  
APAC \*

Tipo: Inteiro(10)

autorizador: \* Nome do profissional autorizador do formulário de solicitação APAC \*

Tipo: String(150)

bairro: \* Bairro onde reside o paciente \*

Tipo: String(150)

campo\_antigo: \* Descrição do campo antes de ser alterado no registro de atividades  
\*

Tipo: String(100)

campo\_novo: \* Descrição do campo após ser alterado no registro de atividades \*

Tipo: String(100)

cep: \* CEP do paciente \*

Tipo: String(22)

**CID = {cid}**

**Cid** = \*registro de dados do código internacional de doenças\*

@id\_cid + descricao

cid\_id\_causas\_assoc: \* Código do CID 10 das causas associadas da justificativa \*

Tipo: Inteiro(10)

cid\_id\_secundario: \* Código do CID 10 secundario da justificativa \*

Tipo: Inteiro(10)

cid\_id\_principal: \* Código do CID 10 principal da justificativa \*

Tipo: Inteiro(10)

cned: \* Código nacional do estabelecimento de saúde \*

Tipo: Inteiro(45)

cod\_ibge\_municipio: \* Código do município estabelecido pelo Instituto Brasileiro de Geografia e Estatística (IBGE) \*

Tipo: String(22)

cod\_org\_emissor: \* Código do órgão emissor da autorização do formulário de solicitação APAC \*

Tipo: String(10)

codigo: \* Código do procedimento SUS \*

Tipo: String(11)

cor: \* Cor do paciente \*

Tipo: String(2)

data: \* Data de criação do formulário de solicitação APAC; data da autorização do formulário de solicitação APAC; data da auditoria \*

Tipo: Date(10)

Tipo: Date(10)

Tipo: Date(10)

data\_nascimento: \* Data de nascimento do paciente \*

Tipo: Date(10)

data\_val\_ini: \* Início do período de validade da autorização do formulário de solicitação APAC \*

Tipo: Date(10)

data\_val\_fim: \* Final do período de validade da autorização do formulário de solicitação APAC \*

Tipo: Date(10)

descricao: \* Descrição do CID 10; descrição do procedimento do registro de atividades (log do sistema) \*

Tipo: String(250)

Tipo: String(45)

estabelecimento\_de\_saude\_id\_executante: \* Código do estabelecimento de saúde executante do formulário de solicitação APAC \*

Tipo: Inteiro(10)

estabelecimento\_de\_saude\_id\_solicitante: \* Código do estabelecimento de saúde solicitante do formulário de solicitação APAC \*

Tipo: Inteiro(10)

**ENTIDADE** = {entidade}

**Entidade** = \*registro do nome da entidade da classe Auditoria\*

@id + nome

etnia: \* Etnia do paciente \*

Tipo: String(2)

**ESTABELECIMENTO\_DE\_SAUDE** = {estabelecimento\_de\_saude}

**Estabelecimento\_de\_saude** = \*registro de dados do estabelecimento de saúde

executante e/ou solicitante do formulário de solicitação APAC\*

@id\_estabelecimento\_de\_saude + cned + nome

**FORMULARIO = {formulario}**

**Formulario = \*registro de dados do formulário de solicitação de APAC\***

@id\_formulario + data + autenticacao + paciente\_id\_paciente +  
solicitante\_id\_solicitante + estabelecimento\_de\_saude\_id\_solicitante +  
proc\_justificativa\_id + status\_id\_status + autorizacao\_id\_autorizacao

formulario\_id\_formulario: \* Código do formulário atribuído a entidade  
Formulario\_has\_procedimento\_sus para sua identificação \*

Tipo: Inteiro(10)

**FORMULARIOS\_HAS\_PROCEDIMENTO\_SUS =**  
{formularios\_has\_procedimento\_sus}

**Formularios\_has\_procedimento\_sus = \*estabelece a ligação entre as entidades  
Formulario e Procedimento\_sus, registrando o número de vezes que este  
procedimento fora solicitado\***

@formulario\_id\_formulario + procedimento\_sus\_id\_procedimento\_sus + quantidade

id: \* Código atribuído a entidade Procedimento para sua identificação; código  
atribuído a Entidade para sua identificação \*

Tipo: Inteiro(10)

id\_auditoria: \* Código atribuído a entidade Auditoria para sua identificação \*

Tipo: Inteiro(10)

id\_autorizacao: \* Código atribuído a entidade Autorizacao para sua identificação \*

Tipo: Inteiro(10)

id\_cid: \* Código atribuído a entidade Cid para sua identificação \*

Tipo: Inteiro(10)

id\_entidade\_alvo: \* Nome da entidade que sofre a ação no registro de atividades \*

Tipo: String(45)

id\_estabelecimento\_de\_saude: \* Código atribuído a entidade Estabelecimento\_de\_saude para sua identificação \*

Tipo: Inteiro(10)

id\_formulário: \* Código atribuído a entidade Formulario para sua identificação \*

Tipo: Inteiro(10)

id\_paciente: \* Código atribuído a entidade Paciente para sua identificação \*

Tipo: Inteiro(10)

id\_proc\_justificativa: \* Código atribuído a entidade Proc\_justificativa para sua identificação \*

Tipo: Inteiro(10)

id\_procedimento\_sus: \* Código atribuído a entidade Procedimento\_sus para sua identificação \*

Tipo: Inteiro(10)

id\_solicitante: \* Código atribuído a entidade Solicitante para sua identificação \*

Tipo: Inteiro(10)

id\_status: \* Código atribuído a entidade Status para sua identificação \*

Tipo: Inteiro(10)

id\_usuario: \* Código atribuído a entidade Usuario para sua identificação \*

Tipo: Inteiro(10)

login: \* Identificação do usuário do sistema \*

Tipo: String(60)

logradouro: \* Logradouro do paciente \*

Tipo: String(150)

municipio: \* Município onde reside o paciente \*

Tipo: String(150)

nome: \* Nome do paciente; nome do solicitante; nome do estabelecimento de saúde; nome do procedimento SUS; nome do usuário do sistema; nome da entidade do sistema \*

Tipo: String(150)

Tipo: String(150)

Tipo: String(300)

Tipo: String(150)

Tipo: String(45)

nome\_abrev: \* Nome abreviado do procedimento SUS \*

Tipo: String(200)

nome\_mae: \* Nome da mãe do paciente \*

Tipo: String(150)

nome\_responsavel: \* Nome do responsável do paciente \*

Tipo: String(150)

num\_autorizacac: \* Número da autorização do formulário de solicitação APAC \*

Tipo: String(20)

num\_doc: \* Número do documento (CNS/CPF) do profissional autorizador do formulário de solicitação APAC; número do documento (CNS/CPF) do profissional solicitante do formulário de solicitação APAC\*

Tipo: String(16)

Tipo: String(16)

num\_prontuario: \* Número do prontuário do paciente \*

Tipo: String(22)

num\_residencia: \* Número da residência do paciente \*

Tipo: String(10)

observacoes: \* Observações da justificativa para a solicitação de APAC \*

Tipo: String(1000)

**PACIENTE = {paciente}**

**Paciente = \*registro de dados do paciente\***

@id\_paciente + num\_prontuario + nome + sexo + cnd + data\_nascimento + cor +  
etnia + nome\_mae + telefone\_mae + nome\_responsavel + telefone\_responsavel +  
peso + altura + logradouro + num\_residencia + bairro + municipio +  
cod\_ibge\_municipio + uf + cep

paciente\_id\_paciente: \* Código do paciente do formulário de solicitação APAC \*

Tipo: Inteiro(10)

perfil: \* Perfil de acesso do usuário do sistema \*

Tipo: Inteiro(1)

peso: \* Peso do paciente \*

Tipo: Float

**PROC\_JUSTIFICATIVA = {proc\_justificativa}**

**Proc\_justificativa = \*registro de dados da justificativa do formulário de solicitação APAC\***

@id\_proc\_justificativa + observacoes + cid\_id\_principal + cid\_id\_secundario +

cid\_id\_causas\_assoc

proc\_justificativa\_id: \* Código da justificativa do procedimento do formulário de solicitação APAC \*

Tipo: Inteiro(10)

**PROCEDIMENTO** = {procedimento}

**Procedimento** = \*registro da descrição do procedimento da classe Auditoria\*

@id + descricao

procedimento\_id: \* Código de identificação do procedimento realizado no registro de atividades \*

Tipo: Inteiro(10)

procedimento\_realizado: \* Descrição do procedimento realizado no registro de atividades \*

Tipo: String(200)

**PROCEDIMENTO\_SUS** = {procedimento\_sus}

**Procedimento\_sus** = \*registro dos dados do procedimento do formulário de solicitação APAC\*

@id\_procedimento\_sus + codigo + nome + nome\_abrev

procedimento\_sus\_id\_procedimento\_sus: \* Código do procedimento SUS atribuído a entidade Formulario\_has\_procedimento\_sus para sua identificação \*

Tipo: Inteiro(10)

quantidade: \* Quantidade de procedimentos solicitados \*

Tipo: Inteiro(10)

senha: \* Senha do usuário do sistema \*

Tipo: String(50)

setor: \* Setor em que o usuário do sistema está alocado \*

Tipo: String(100)

sexo: \* Sexo do paciente \*

Tipo: String(2)

**SOLICITANTE** = {solicitante}

**Solicitante** = \*registro dos dados do solicitante do formulário de solicitação APAC\*

@id\_solicitante + nome + tipo\_doc + num\_doc + usuario\_id\_usuario

solicitante\_id\_solicitante: \* Código do solicitante do formulário de solicitação APAC \*

Tipo: Inteiro(10)

**STATUS** = {status}

**Status** = \*registra o status do formulário de solicitação APAC\*

@id\_status + status

status: \* Descrição do status do formulário \*

Tipo: String(150)

status\_id\_status: \* Código do status do formulário de solicitação APAC \*

Tipo: Inteiro(10)

telefone\_mae: \* Telefone da mãe do paciente \*

Tipo: String(26)

telefone\_responsavel: \* Telefone do responsável do paciente \*

Tipo: String(26)

tipo\_doc: \* Tipo do documento do profissional responsável pela autorização do formulário de solicitação APAC; tipo do documento do profissional solicitante do formulário de solicitação APAC \*

Tipo: Inteiro(1)

Tipo: Inteiro(1)

tipo\_entidade\_id: \* Código da entidade que sofre a ação no registro de atividades \*

Tipo: Inteiro(10)

uf: \* Unidade Federativa onde reside o paciente \*

Tipo: String(2)

**USUARIO** = {usuario}

**Usuario** = \*registro de dados do usuário da aplicação\*

@id\_usuario + nome + login + senha + setor + perfil

usuario\_id\_usuario: \* Código do usuário do sistema, o qual está vinculado o cadastro do solicitante; código do usuário que realizou alguma atividade no registro de atividades \*

Tipo: Inteiro(10)

Tipo: Inteiro(10)

## 2.5 Diagrama de Sequência

### 2.4.1 Caso de Uso 01: Logar no Sistema

A figura 3 detalha os caminhos que o ator usuário (Administrador, Solicitante, Administrativo e Administrativo Financeiro) terá de seguir para conseguir *logar* no sistema. A sequência dos passos inicia com o ator inserindo seus dados no formulário de *login*; a submissão desses dados chama o método **logar** que está presente na classe **UsuarioBean** que, por sua vez, está situada no pacote **Bean**;

esse método usa a função **buscar** que está presente na classe **UsuarioDao** que, por sua vez, está situada no pacote **Dao**. A função tem dois retornos, podendo ser nulo (não encontrou o usuário e o sistema retorna para a **tela de login**) ou trará os dados do usuário do banco de dados da aplicação; em seguida o método **conexaoLdapBarros** que está situado na classe **FabricaDeConexoes** situada no pacote **Util**. Essa função validará os dados do usuário no banco de dados do Hospital Universitário João de Barros Barreto retornando um valor booleano, caso os dados sejam validados com sucesso, a **tela inicial** do sistema é exibida.

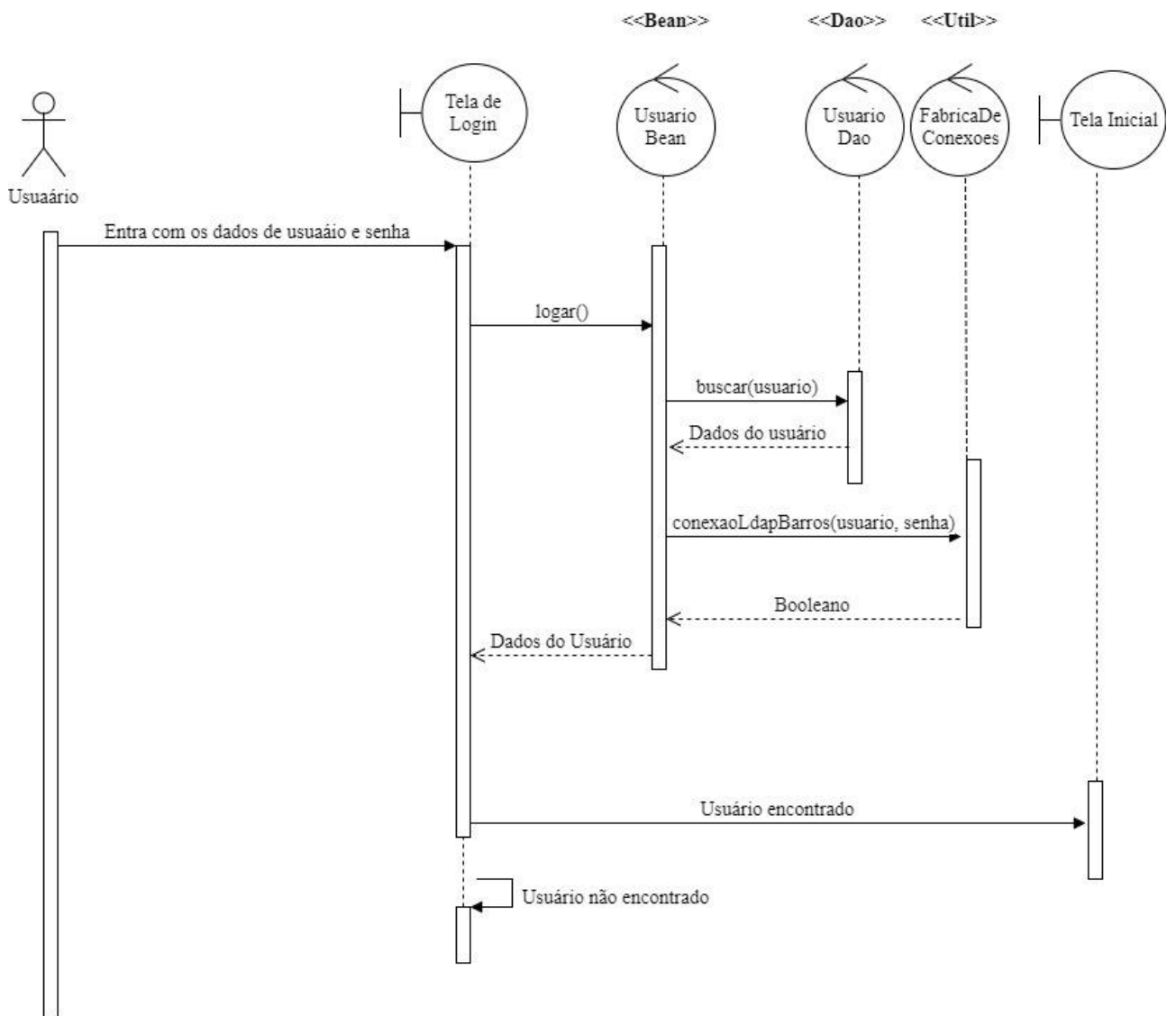


Figura 3. Diagrama de seqüência do funcionamento do caso de uso 01: logar no sistema

## 2.4.2 Caso de Uso 02: Manter Usuários // Cenário: Cadastrar Usuários

A figura 4 detalha os caminhos que o ator Administrador terá de seguir para conseguir cadastrar um novo usuário no sistema. A sequência dos passos inicia com o ator inserindo os dados do usuário no formulário da **tela de cadastro**; a submissão desses dados chama o método **btCadastrar** que está presente na classe **SolicitanteBean** que, por sua vez, está situada no pacote **Bean**; esse método chama a função **salvar** que está presente na classe **SolicitanteDao** situada no pacote **Dao**. A função retorna um valor booleano; caso o valor seja falso, exibe a mensagem “Erro ao cadastrar usuário”; caso o valor seja verdadeiro, o sistema exibe a mensagem “Cadastro realizado com sucesso”.

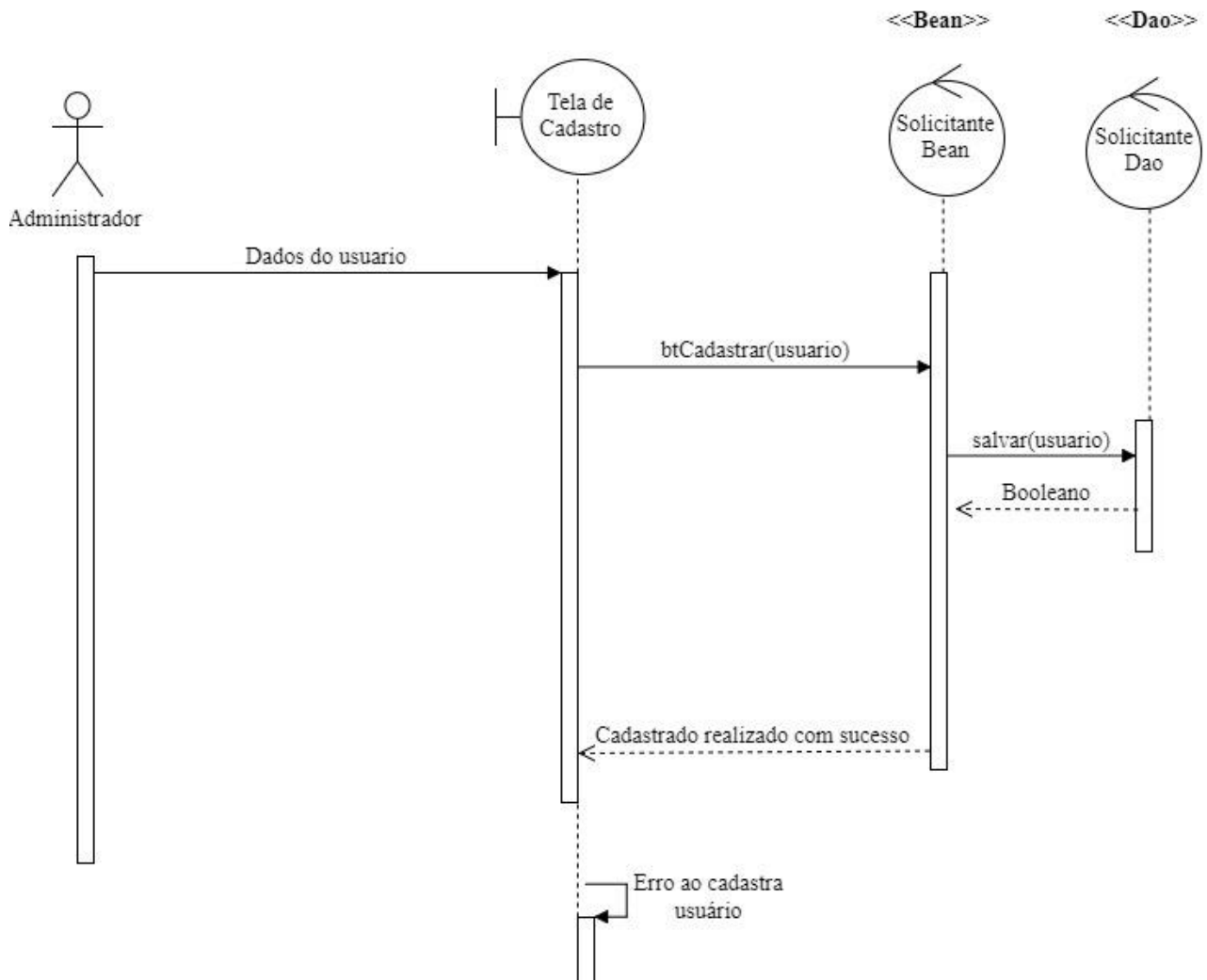


Figura 4. Diagrama de sequência do funcionamento do caso de uso 02: manter usuários // Cenário cadastrar usuário

### 2.4.3 Caso de Uso 02: Manter Usuários // Cenário: Editar Usuários

A figura 5 detalha os caminhos que o ator Administrador terá de seguir para conseguir editar dados de um usuário no sistema. A sequência dos passos inicia com o ator inserindo os dados do usuário no formulário da **tela de edição de usuário**; a submissão desses dados chama o método **btAtualizar** que está presente na classe **SolicitanteBean** que, por sua vez, está situada no pacote **Bean**; esse método chama a função **atualizar** que está presente na classe **SolicitanteDao** situada no pacote **Dao**. A função retorna um valor booleano; caso o valor seja falso, exibe a mensagem “Falha ao atualizar dados do usuário”; caso o valor seja verdadeiro, o sistema exibe a mensagem “Dados Atualizados”.

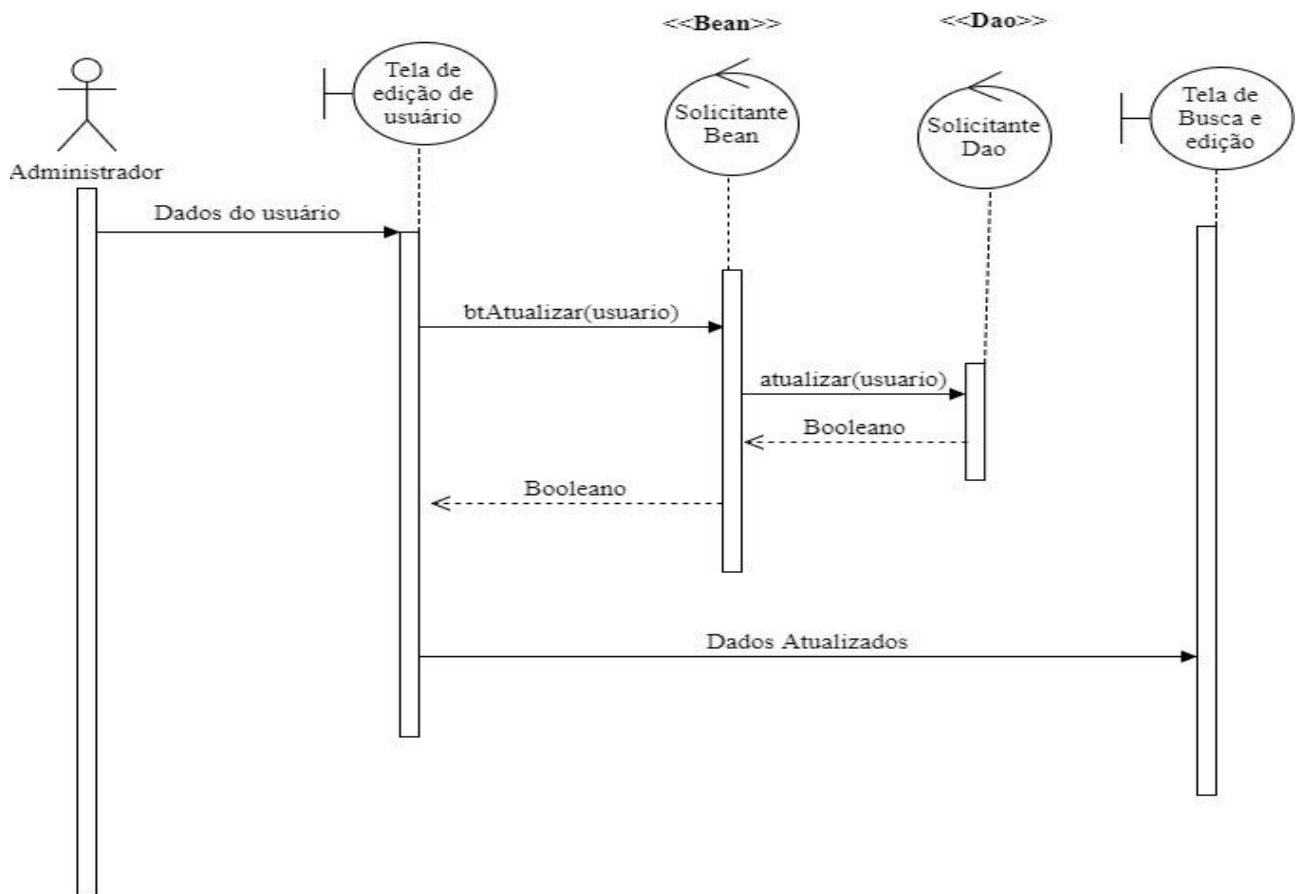


Figura 5. Diagrama de sequência do funcionamento do caso de uso 02: manter usuários // Cenário: editar usuários

#### 2.4.4 Caso de Uso 02: Manter Usuários // Cenário: Desativar Usuários

A figura 6 detalha os caminhos que o ator Administrador terá de seguir para conseguir desativar um usuário no sistema. A sequência dos passos inicia com o ator inserindo os dados do usuário no formulário da **tela de busca e edição de usuário**; a submissão desses dados chama o método **btDelConfirmSimdIlgDesativar** que está presente na classe **SolicitanteBean** que, por sua vez, está situada no pacote **Bean**; esse método chama a função **atualizar** que está presente na classe **SolicitanteDao** situada no pacote **Dao**. A função retorna um valor booleano; caso o valor seja falso, exibe a mensagem “Falha ao desativar usuário”; caso o valor seja verdadeiro, o sistema exibe a mensagem “Usuário desativado com sucesso”.

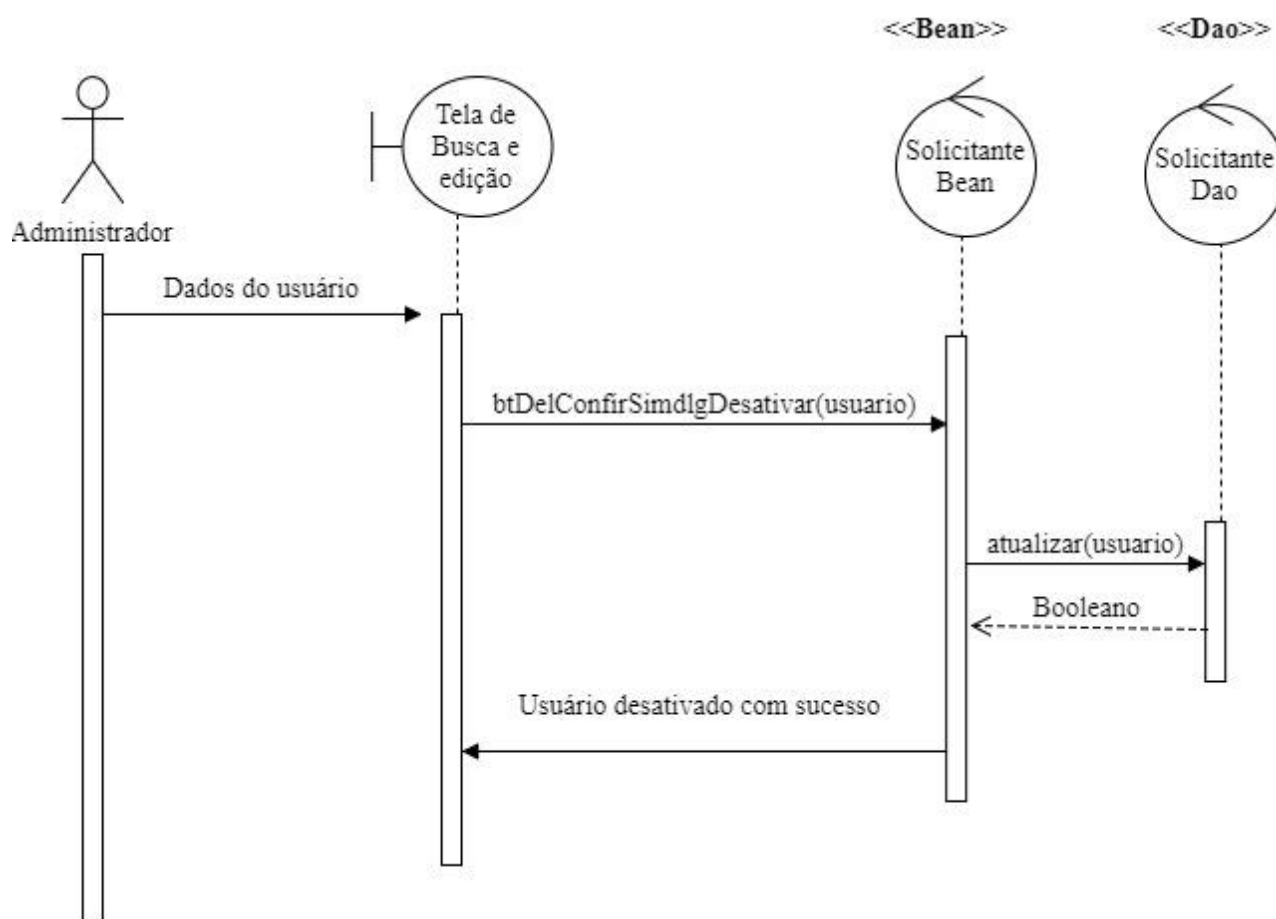


Figura 6. Diagrama de sequência do funcionamento do caso de uso 02: manter usuários // Cenário: Desativar usuários

### 2.4.5 Caso de Uso 02: Manter Usuários // Cenário: Buscar Usuários

A figura 7 detalha os caminhos que o ator Administrador terá de seguir para conseguir buscar dados de um usuário no sistema. A sequência dos passos inicia com o ator inserindo os dados do usuário no formulário da **tela de busca e edição de usuário**; a submissão desses dados chama o método **buscarAtivos** que está presente na classe **SolicitanteBean** que, por sua vez, está situada no pacote **Bean**; esse método chama a função **buscar** que está presente na classe **SolicitanteDao** situada no pacote **Dao**. A função retorna a lista com os usuários do sistema.

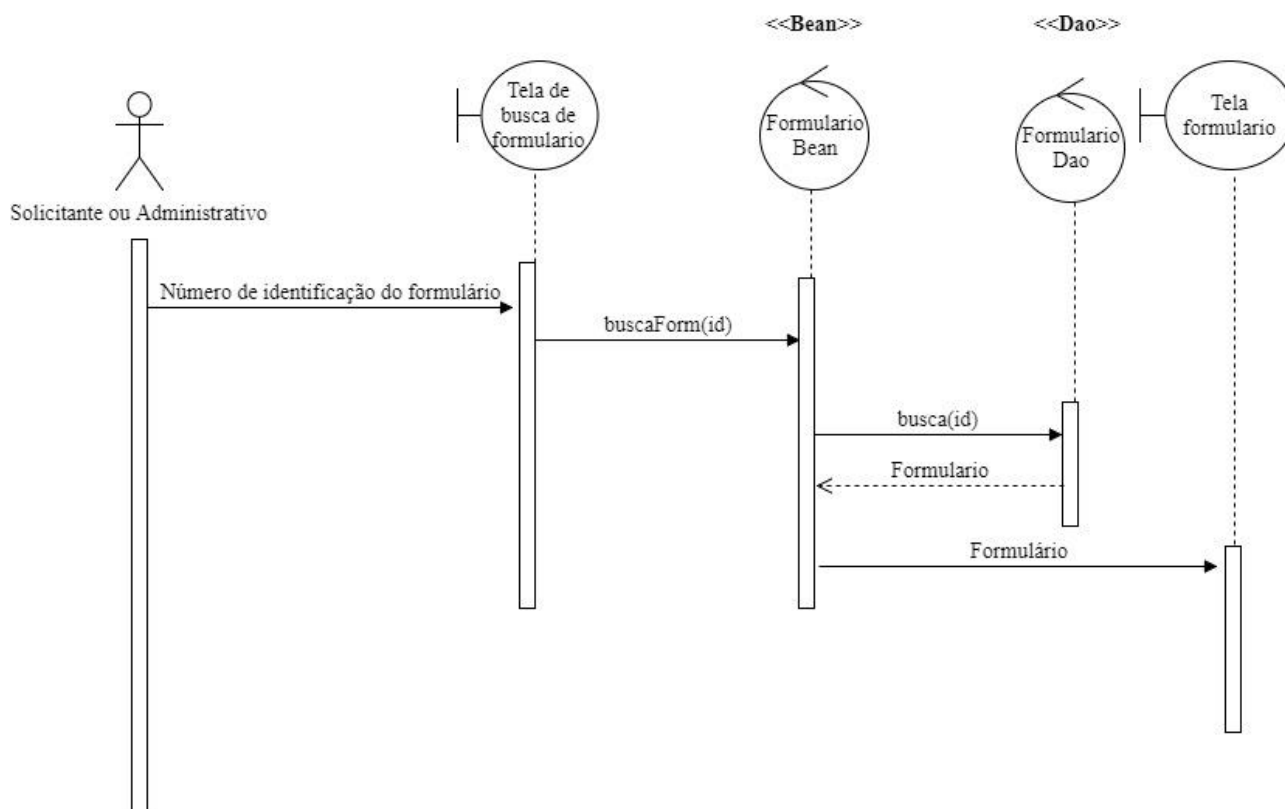
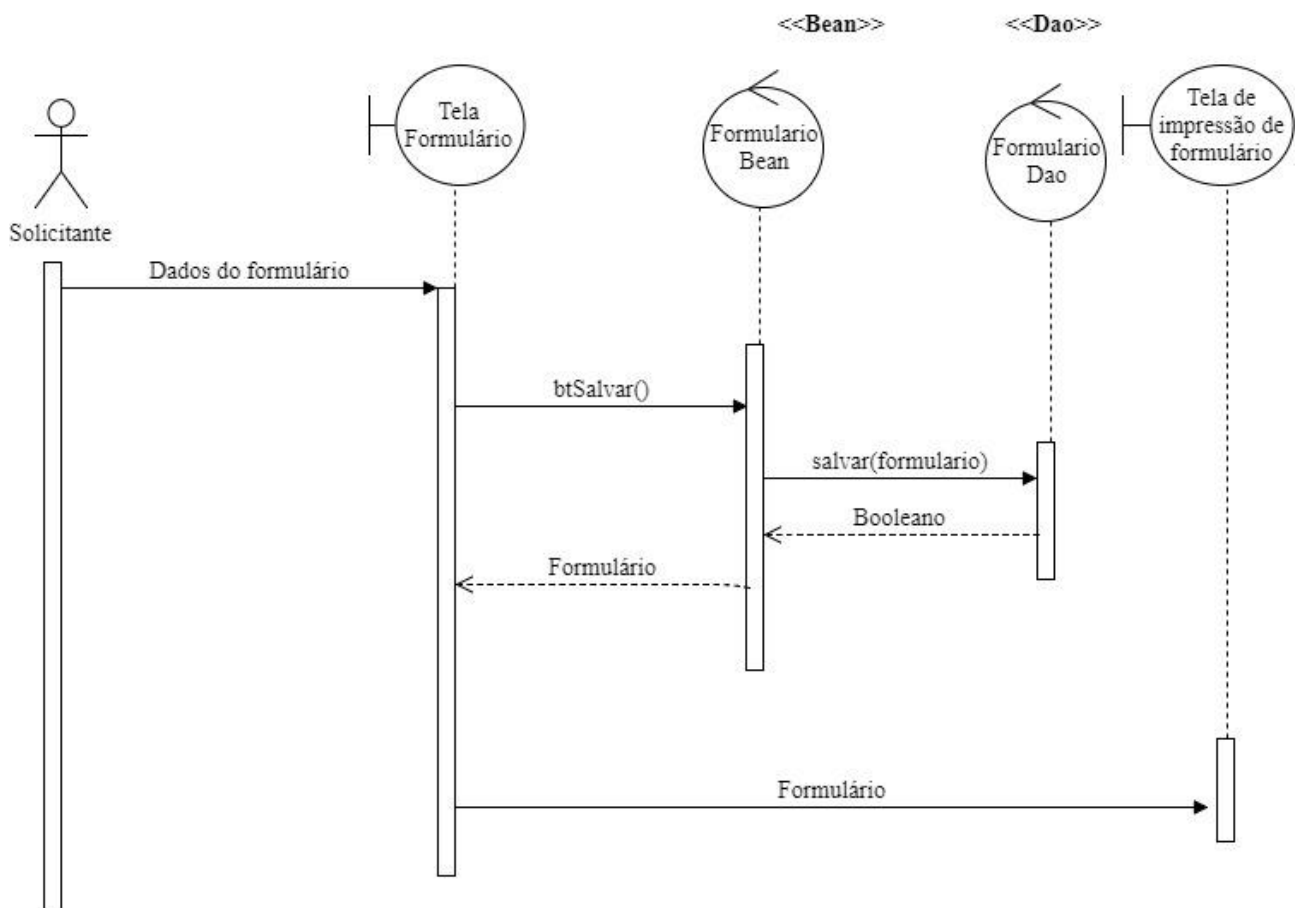


Figura 7. Diagrama de sequência do funcionamento do caso de uso 02: manter usuários // Cenário: buscar usuários

### 2.4.6 Caso de Uso 03: Manter Formulário de Autorização de Procedimentos // Cenário: Criação de um Novo Formulário de Autorização de Procedimentos

A figura 8 detalha os caminhos que o ator Solicitante terá de seguir para conseguir criar um formulário de autorização de procedimentos no sistema. A sequência dos

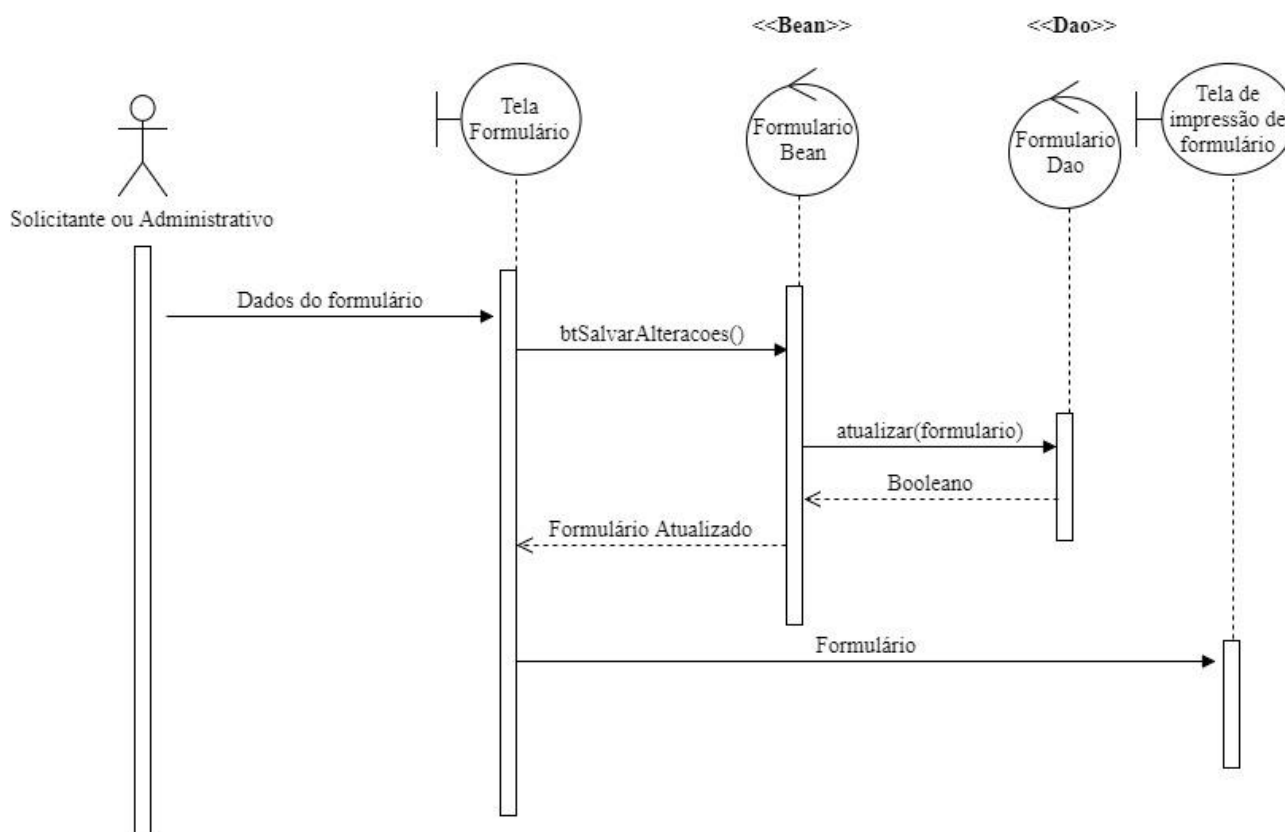
passos inicia com o ator inserindo os dados do formulário na **tela formulário**; a submissão desses dados chama o método **btSalvar** que está presente na classe **FormularioBean** que, por sua vez, está situada no pacote **Bean**; esse método chama a função **salvar** que está presente na classe **FormularioDao** situada no pacote **Dao**. A função retorna um valor booleano; caso o valor seja falso, exibe a mensagem “Falha ao salvar dados do formulário”; caso o valor seja verdadeiro, o sistema exibe a mensagem “Formulário salvo com sucesso”; então a opção para imprimir o formulário é habilitada no sistema.



**Figura 8. Diagrama de seqüência do funcionamento do caso de uso 03: manter formulário de autorização de procedimentos // Cenário: criação de um novo formulário de autorização de procedimentos**

**2.4.7 Caso de Uso 03: Manter Formulário de Autorização de Procedimentos // Cenário: Edição de Formulário de Autorização de Procedimentos**

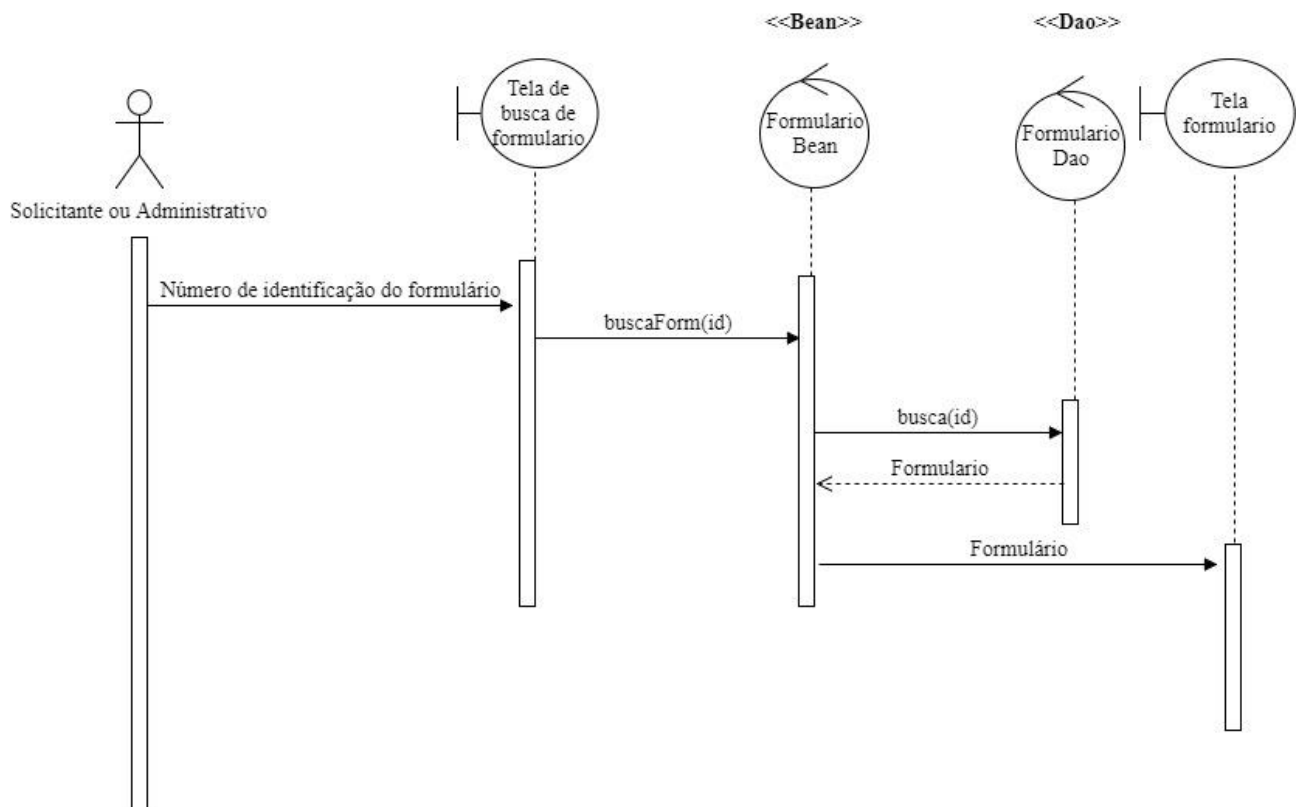
A figura 9 detalha os caminhos que o ator Solicitante ou Administrativo terá de seguir para conseguir editar um formulário de autorização de procedimentos no sistema. A sequência dos passos inicia com o ator inserindo os dados do formulário na **tela formulário**; a submissão desses dados chama o método **btSalvar** que está presente na classe **FormularioBean** que, por sua vez, está situada no pacote **Bean**; esse método chama a função **atualizar** que está presente na classe **FormularioDao** situada no pacote **Dao**. A função retorna um valor booleano; caso o valor seja falso, exibe a mensagem “Falha ao atualizar dados do formulário”; caso o valor seja verdadeiro, o sistema exibe a mensagem “Formulário atualizado com sucesso”; caso o status desse formulário no sistema esteja como **Salvo**, opção para imprimir o formulário é habilitada no sistema.



**Figura 9. Diagrama de sequência do funcionamento do caso de uso 03: manter formulário de autorização de procedimentos // Cenário: edição do formulário de autorização de procedimentos**

**2.4.8 Caso de Uso 03: Manter Formulário de Autorização de Procedimentos // Cenário: Buscar Formulário de Autorização de Procedimentos**

A figura 10 detalha os caminhos que o ator Solicitante ou Administrativo terá de seguir para conseguir buscar um formulário de autorização de procedimentos no sistema. A sequência dos passos inicia com o ator inserindo os dados do formulário na **tela de busca de formulário**; a submissão desses dados chama o método **buscaForm** que está presente na classe **FormularioBean** que, por sua vez, está situada no pacote **Bean**; esse método chama a função **buscar** que está presente na classe **FormularioDao** situada no pacote **Dao**. A função retorna o formulário; o formulário é exibido na **tela formulario**.

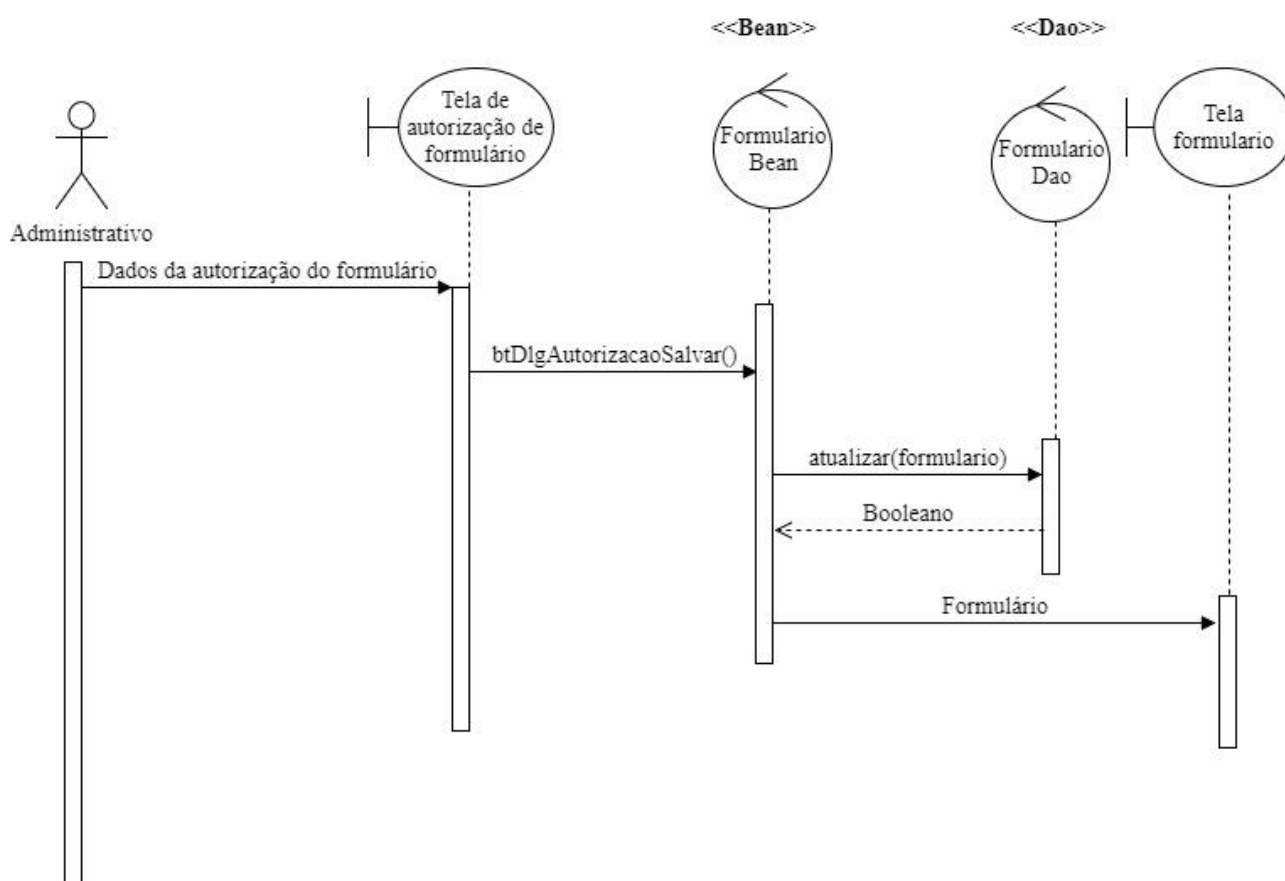


**Figura 10. Diagrama de sequência do funcionamento do caso de uso 03: manter formulário de autorização de procedimentos // Cenário: buscar do formulário de autorização de procedimentos**

#### 2.4.9 Caso de Uso 04: Registro de Autorização de Formulário de Autorização de Procedimentos

A figura 11 detalha os caminhos que o ator Administrativo terá de seguir para

conseguir autorizar um formulário de autorização de procedimentos no sistema. A sequência dos passos inicia com o ator inserindo os dados da autorização do formulário na **tela de autorização de formulário**; a submissão desses dados chama o método **btDlgAutorizacao** que está presente na classe **FormularioBean** que, por sua vez, está situada no pacote **Bean**; esse método chama a função **atualizar** que está presente na classe **FormularioDao** situada no pacote **Dao**. A função retorna um valor booleano; caso o valor seja falso, exibe a mensagem “Erro ao registrar autorização do formulário”; caso o valor seja verdadeiro, o sistema exibe a mensagem “Autorização registrada com sucesso” exibindo o formulário na **tela formulario**.



**Figura 11. Diagrama de sequência do funcionamento do caso de uso 04: registro de autorização de formulário de autorização de procedimentos**

#### 2.4.10 Caso de Uso 05: Relatórios do Sistema

A figura 12 detalha os caminhos que o ator Administrativo Financeiro terá de seguir

para conseguir emitir relatórios do sistema. A sequência dos passos inicia com o ator inserindo os dados do filtro do relatório na **tela de relatórios**; a submissão desses dados chama o método **pesquisaRel** que está presente na classe **RelatorioBean** que, por sua vez, está situada no pacote **Bean**; esse método chama a função **quantitativo** que está presente na classe **RelatorioDao** situada no pacote **Dao**. A função retorna o relatório desejado.

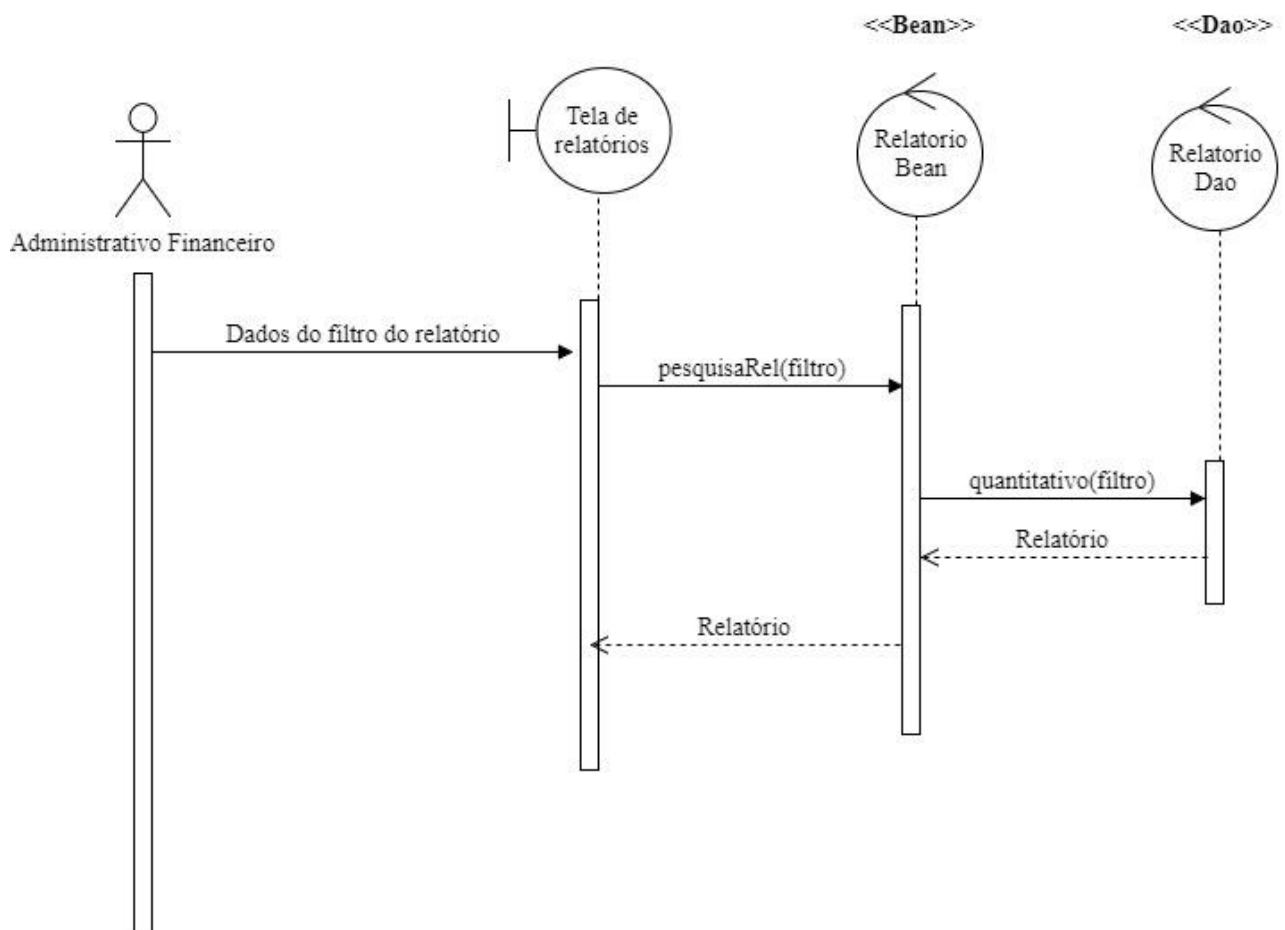


Figura 12. Diagrama de sequência do funcionamento do caso de uso 05: relatórios do sistema

## 2.5 Diagrama de Máquina de Estados

A figura abaixo demonstra o diagrama de máquina de estados da criação de um novo formulário. O formulário pode assumir os estados de Salvo, Impresso, Autorizado, Não autorizado, Finalizado e Excluído.

Durante a criação de um novo formulário, através do método **btSalvar**, o

formulário assume o estado de Salvo. A partir disso, o formulário estará apto a mudar para os próximos Estados.

A partir do momento que o formulário é impresso, ele assume o estado de Impresso através do método **btImprimir**, nesse momento o formulário não pode ser mais excluído ou editado.

O formulário, a partir do momento que é impresso, informando os dados da autorização o formulário pode assumir os estados de Autorizado ou Não Autorizado, através do método **btDlgAutorizacaoSalvar**.

Após o formulário ser finalizado com a autorização, ele assume o estado de Finalizado, através do método **btFormularioSalvar**.

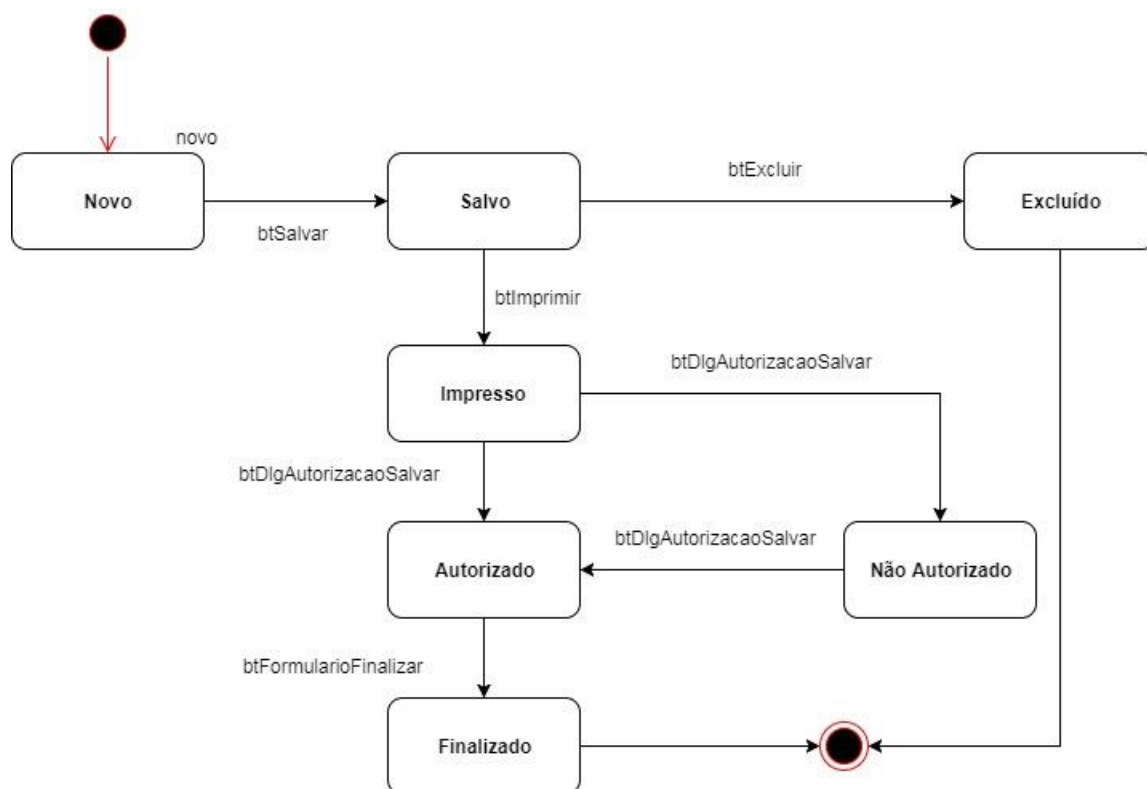


Figura 13. Diagrama de máquina de estados do Formulário

### 3. BIBLIOGRAFIA

FURTADO, Alfredo Braga; COSTA JR, Júlio Valente da. “**PRÁTICA DE ANÁLISE E PROJETO DE SISTEMAS**”. Editora da Universidade Federal do Pará. Belém. 2010.

**APÊNDICE D – DOCUMENTO DE ESPECIFICAÇÃO DE PROJETO DO  
APAC BARROS BARRETO**

# **APAC Barros Barreto, Sistema de Solicitação de Autorização de Procedimentos Ambulatoriais de Alta Complexidade**

## **Documento de Especificação de Projeto**

### **1. Introdução**

Este documento contém a Especificação de Projeto para o Sistema APAC Barros Barreto. Nele será especificada a arquitetura do sistema, por meio de diagramas de pacotes e dos respectivos diagramas de classes para os componentes identificados. Para dar mais riqueza de detalhes ao projeto.

É importante definir nesta fase a plataforma de implementação e o mecanismo de persistência de dados utilizados, para que o restante do projeto seja construído de forma realista, visando a implementação final. Neste caso, o sistema será desenvolvido utilizando a linguagem de programação *Java EE* em conjunto com o *framework PrimeFaces*, banco de dados MySQL e plataforma de desenvolvimento *NetBeans*.

### **2. Plataforma computacional**

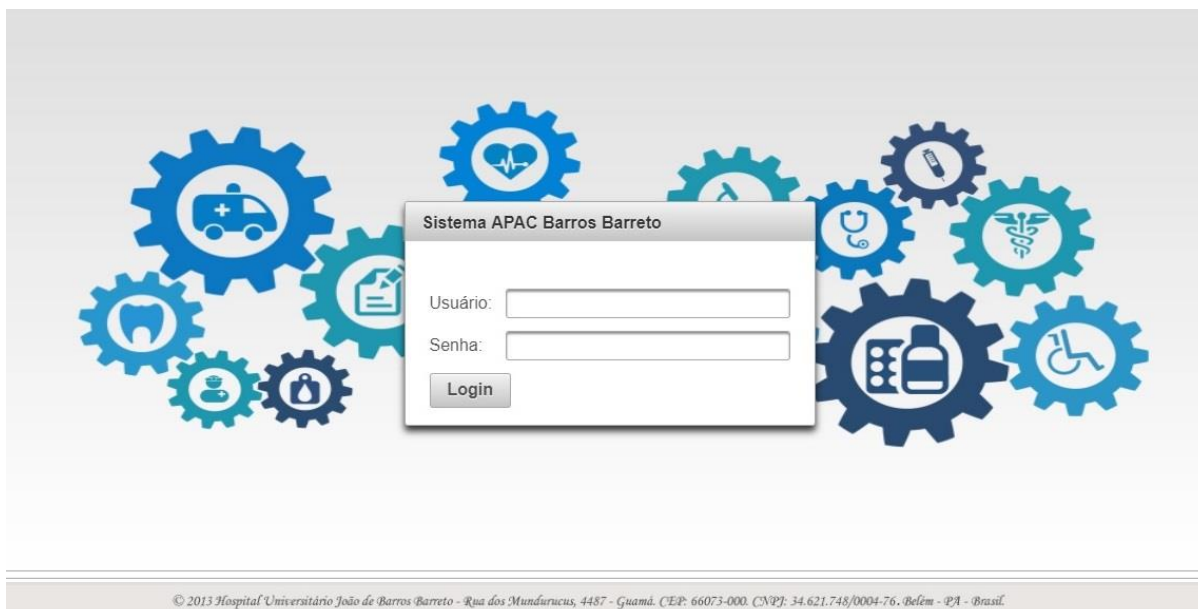
Tratando-se de uma aplicação *web* o equipamento de acesso deve possuir acesso à intranet do Hospital Universitário João de Barros e as configurações mínimas para execução dos navegadores Google Chrome, Microsoft EDGE e Mozilla FireFox.

### **3. Arquitetura do sistema**

A arquitetura do sistema já fora definida no documento de Especificação de Análise, nos diagramas de pacotes, especificando cada pacote em seu determinado caso de uso e diagrama de classe.

### **4. Projeto de interação humano-computador**

Nessa seção são apresentadas as telas que serão utilizadas pelo sistema.



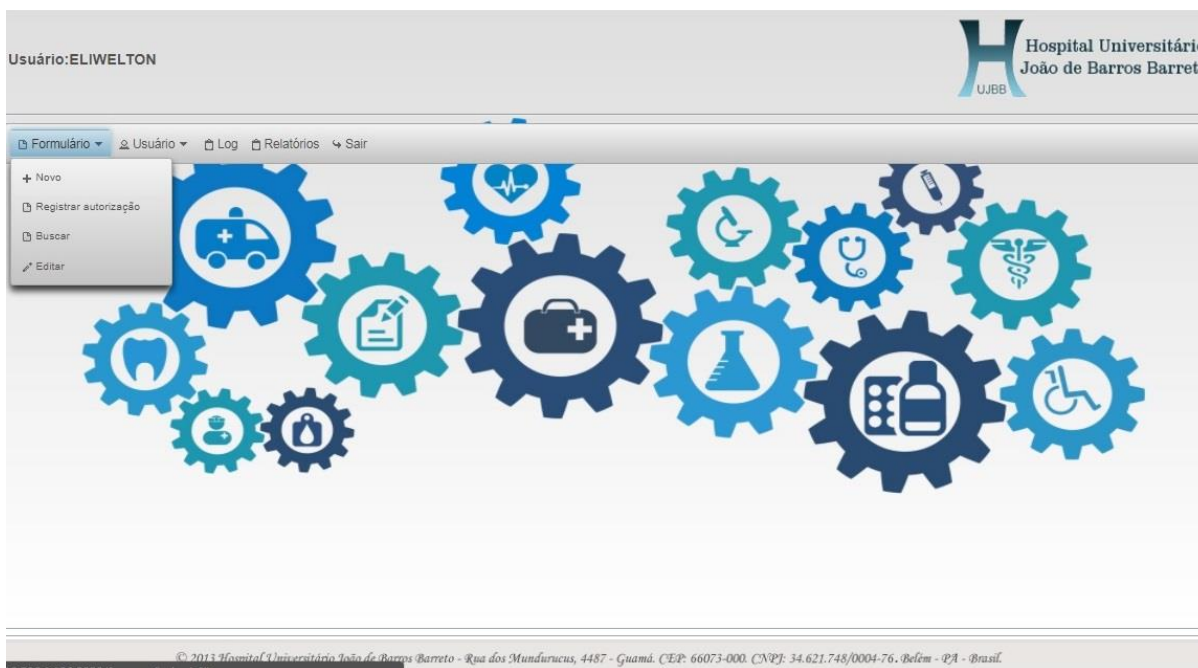
**Figura 1. Tela de Login do APAC Barros Barreto.**

A figura 1 demonstra o protótipo de tela de login para autenticação dos usuários afim de utilizarem a ferramenta. O login e senha são, obrigatoriamente, os mesmos do acesso a intranet do Hospital Universitário João de Barros Barreto, porém estes precisam ser cadastrado pelo sistema.



**Figura 2. Tela principal do APAC Barros Barreto.**

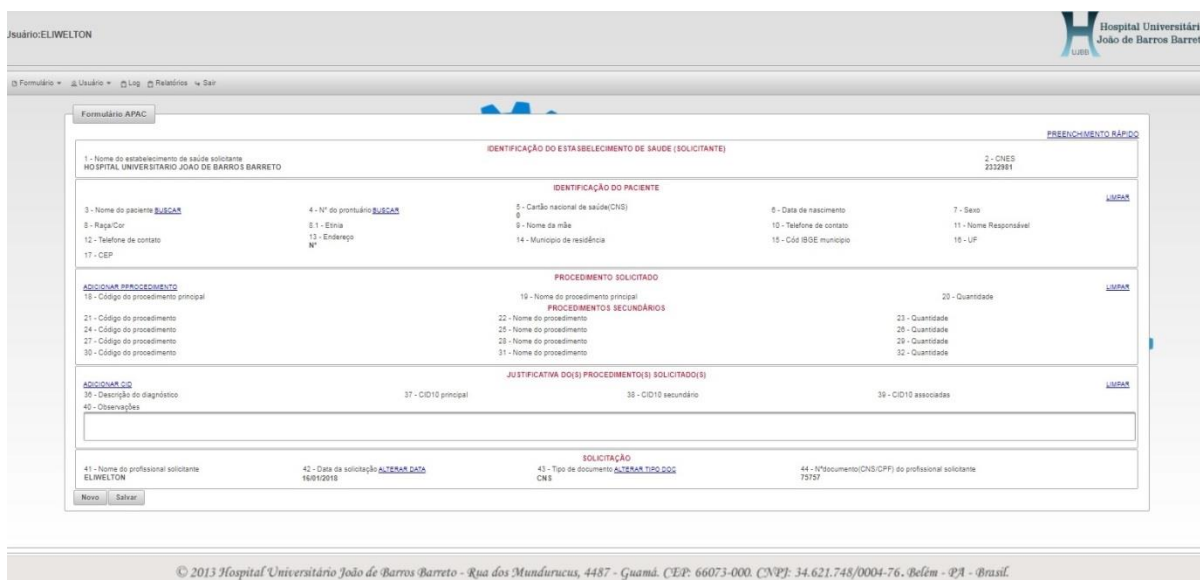
A figura 2 demonstra a tela principal do sistema, apresentando os menus de cada subsistema, estes por sua vez estão vinculados a cada um dos perfis do sistema, portanto, cada menu só é exibido a seu determinado perfil de usuário.



**Figura 3. Menu de Formulário do APAC Barros Barreto.**

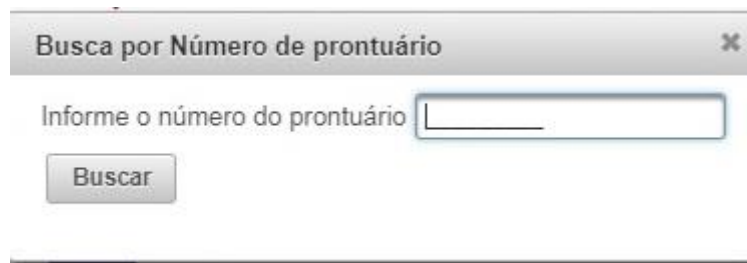
A figura 3 mostra o menu relacionado aos subsistemas de criação de novos formulários APAC e registro de autorização destes formulários. Nele existem as opções:

- **Novo:** Responsável pela criação de um novo formulário APAC. Apenas o usuário com perfil de Solicitante poderá efetuar a criação de um novo formulário APAC.
- **Registrar Autorização:** responsável pelo registro de autorizações. Apenas o usuário com perfil de Administrativo poderá efetuar este registro.
- **Buscar:** Efetua a busca de um formulário.
- **Editar:** responsável pela edição de formulários APAC. Usuários com perfil de Solicitante e Administrativo podem interagir com esta função, porém interagem de forma diferente. Conforme descrito nos casos de uso, existem situações específicas para os usuários.



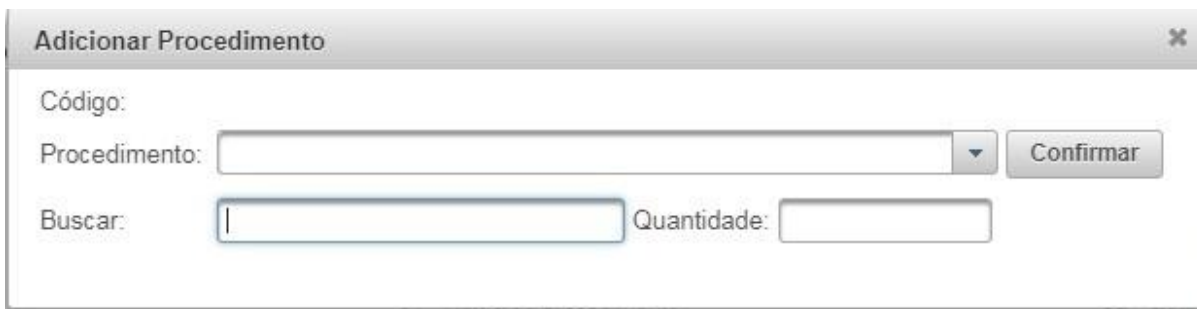
**Figura 4. Tela de Preenchimento de um novo formulário de solicitação APAC.**

A figura 4 demonstra como é a criação de um novo formulário APAC.



**Figura 5. Pop-up para inserir prontuário para preenchimento dos dados do paciente no formulário de solicitação APAC.**

A figura 5 demonstra a janela para inserir os dados do prontuário do paciente, prontuário este que está previamente cadastrado no sistema principal do Hospital Universitário João Barros Barreto. O sistema manda uma requisição para o banco da aplicação e esta retorna os dados do paciente, preenchendo o campo das informações do paciente de forma automática.



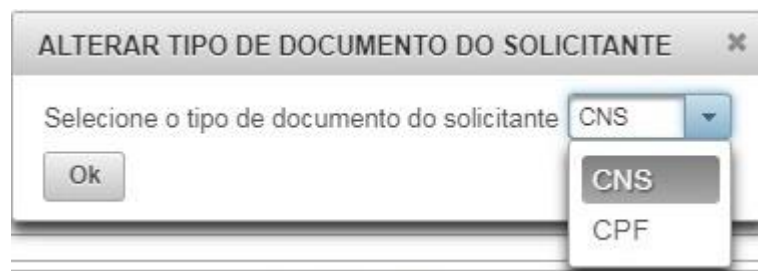
**Figura 6. Pop-up para inserção do procedimento no formulário de solicitação APAC.**

Janela para inserção do(s) procedimento(s) para o qual se quer autorização. O campo Procedimento é para o usuário procurar, através de uma lista completa dos procedimentos, aquele que deseja. O campo de busca pode utilizar as opções de digitar o código do procedimento ou digitar o nome do procedimento, em seguida é exibido o procedimento solicitado. O campo quantidade é para descrever a quantidade de procedimentos que precisam ser solicitadas autorizações.



**Figura 7. Pop-up para inserção do CID 10 no formulário de solicitação APAC.**

A figura 7 demonstra a janela para inserção do CID 10 (Código Internacional de Doenças) relacionado ao procedimento para o qual se quer autorização. O campo selecione Cid 10 exibe uma lista com todos os CIDs 10 cadastrados no sistema principal do hospital. O campo busca utiliza as opções de digitar o código do CID 10 ou digitar o nome da doença.



**Figura 8. Pop-up para alterar o tipo do documento do Solicitante do formulário de solicitação APAC.**

A figura 8 demonstra a janela para selecionar o tipo do documento do usuário com perfil de Solicitante. Este podendo ser o CNS (Conselho Nacional de Saúde) ou CPF (Cadastro de Pessoa Física). Selecionando o tipo de documento, este é preenchido automaticamente no campo de documento do formulário de solicitação APAC, pois ambos os documentos estão previamente cadastrados no sistema principal do hospital, o sistema faz uma solicitação e o banco de dados do sistema principal do o hospital retorna as informações.



**Figura 9. Pop-up para alterar a data do formulário de solicitação APAC.**

A figura 9 demonstra apenas uma janela para alterar a data do formulário. Por padrão o formulário já vem com a data do atual dia da criação do novo formulário de solicitação APAC.

**Figura 10. Pop-up para inserção rápida dos campos do formulário de solicitação APAC.**

A figura 10 demonstra a janela para inserção rápida dos campos do formulário de solicitação APAC. Cada campo corresponde aos campos com mesmo nome do formulário. Esta opção facilita ainda mais o processo de solicitação.

**Figura 11. Pop-up para buscar paciente por nome.**

A figura 11 demonstra a janela para, no caso de não saber o número de prontuário do paciente, efetuar uma busca através de nome. O campo Nome do Paciente existe a lista com os nomes que estão cadastrados no sistema. O campo busca é onde será digitado o nome do paciente para efetuar a busca.

**Figura 12. Pop-up de buscar formulário de solicitação APAC.**

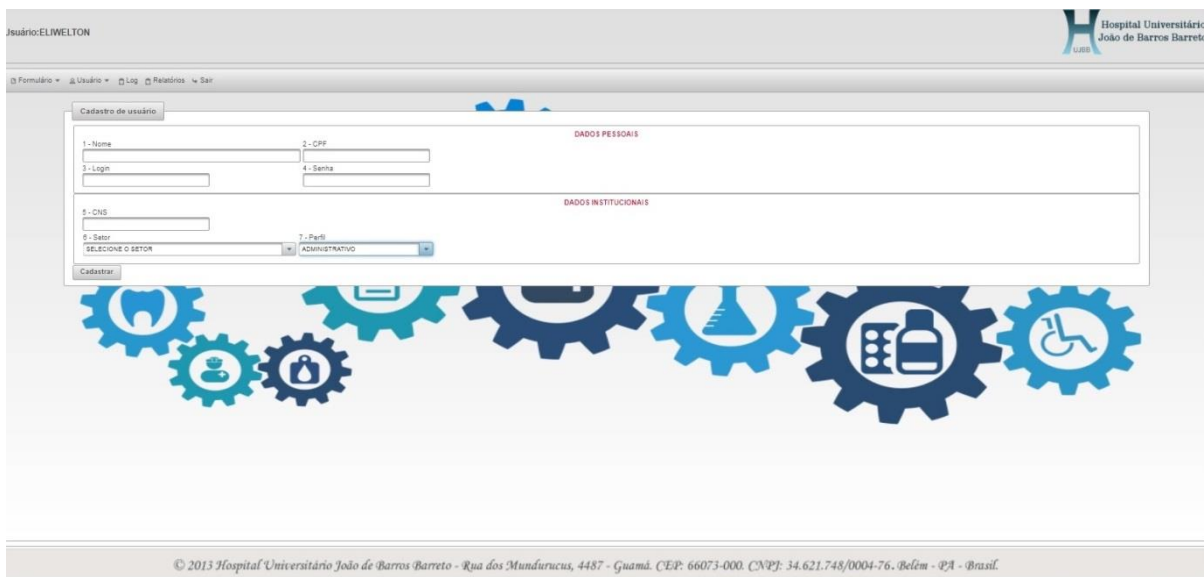
A figura 12 demonstra a janela para efetuar uma busca de um formulário de solicitação APAC específico. No campo descrito o Solicitante ou Administrativo informa o número do formulário e, caso esteja cadastrado, o formulário é exibido na tela.



**Figura 13. Menu de usuário do APAC Barros Barreto.**

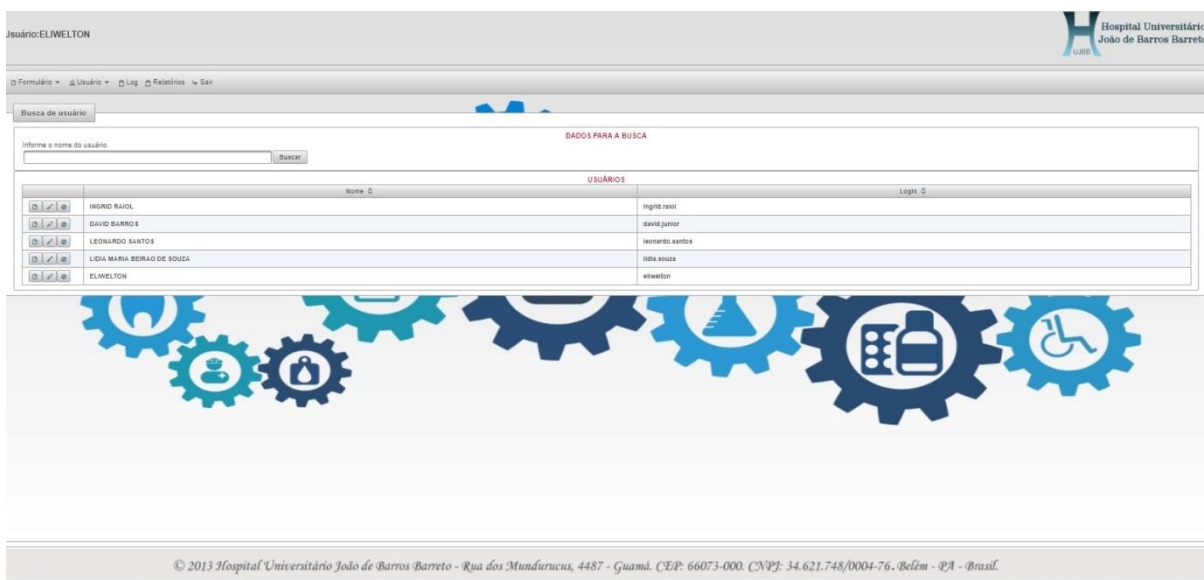
A figura 13 demonstra o subistema relacionado a criação de usuários, que por sua vez só pode ser visualizado por usuários vinculados ao perfil de Administrador. Nele existem a opção:

- **Cadastrar:** responsável por direcionar a tela de cadastro de novos usuários.
- **Buscar / Editar:** responsável por direcionar para tela de busca e edição de usuários do sistema.



**Figura 14. Tela de cadastro de um novo usuário no sistema.**

A figura 14 demonstra a tela de criação de um novo usuário do sistema. Demonstrando, também, os campos relacionados a criação de um novo usuário.



**Figura 15. Tela de busca e edição de usuários do APAC Barros Barreto**

A figura 15 demonstra a tela de busca e edição de usuários já cadastrados no APAC Barros Barreto. Ao entrar na opção de busca e edição de usuários, uma lista com todos os usuários da aplicação é exibida, contendo seus nomes completos e logins

de utilização. Para cada usuário existem três opções:

- **Visualizar:** exibe uma janela contendo os dados do usuário.
- **Editar:** edita os dados do usuário.
- **Desativar:** desativa um usuário do sistema.

O campo busca é utilizado para efetuar uma busca de determinado usuário da aplicação, ao efetuar esta busca, a lista abaixo é retornada contendo o nome do usuário específico.



A janela de edição de dados de usuário, intitulada "Editar dados", é dividida em duas seções principais: "DADOS PESSOAIS" e "DADOS INSTITUCIONAIS".

**DADOS PESSOAIS:**

- 1 - Nome: Campo de texto contendo "DAVID BARROS".
- 2 - CPF: Campo de texto vazio.
- 3 - Login: Campo de texto contendo "david.junior".

**DADOS INSTITUCIONAIS:**

- 4 - CNS: Campo de texto contendo "0".
- 5 - Setor: Menu suspenso com o texto "SELECIONE O SETOR".
- 6 - Perfil: Menu suspenso com o texto "ADMINISTRATIVO FINANCEIRO".

Na base da janela, há dois botões: "Cancelar" e "Atualizar".

**Figura 16. Pop-up de edição de usuário do sistema.**

A figura 16 demonstra a janela de edição de usuários. Na imagem é possível visualizar os campos que são possíveis edição.



A janela de confirmação de desativação de usuário, intitulada "Confirmação", contém o seguinte texto:

Deseja realmente desativar esse usuário?

Na base da janela, há dois botões: "Não" e "Sim".

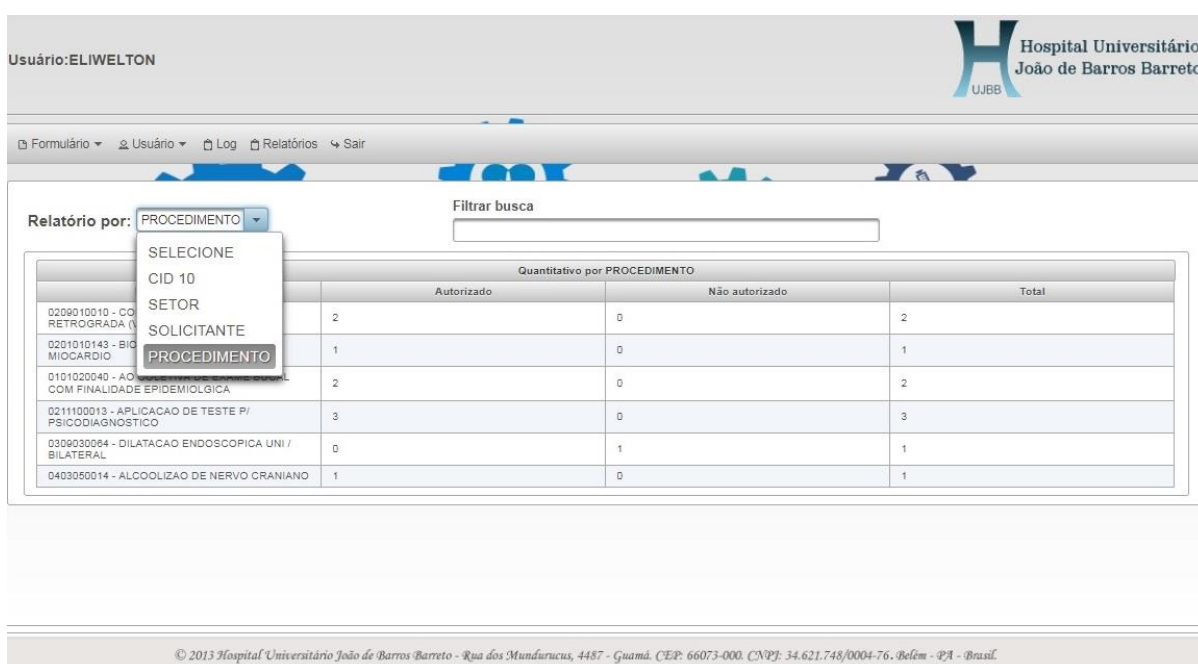
**Figura 17. Pop-up de confirmação de desativação de usuário do sistema.**

A figura 17 demonstra a janela de confirmação de desativação de um usuário do sistema.



**Figura 18. Pop-up de visualização dos dados de usuários do sistema.**

A figura 18 demonstra a janela exibida ao visualizar os dados de um usuário.



**Figura 19. Tela de relatórios do APAC Barros Barreto**

A figura 19 demonstra a tela de exibição do subsistema de emissão de formulários do sistema. Na imagem podemos visualizar que existe 4 tipos de quantitativos que a aplicação fornece:

- **CID 10:** Quantitativo relacionado aos CID 10 que foram utilizados em todos os formulários de solicitação APAC.
- **Setor:** Quantitativo das solicitações feitas por cada Setor do hospital.
- **Solicitante:** Quantitativo de todos os Solicitantes de formulário de solicitação APAC.

- **Procedimento:** Quantitativo de todos os procedimentos que foram solicitadas autorizações.

**Figura 20. Pop-up registro de autorização de formulário de solicitação APAC.**

A figura 20 demonstra a janela relacionada ao subsistema de registro de autorização de formulário de solicitação APAC. Somente o usuário com perfil Administrativo pode interagir com esta tela, nela estão os campos para os quais o usuário deverá preencher informando os dados da autorização concedida pelo órgão executante.

LOG

**REGISTROS DE ATIVIDADES**

Data	Usuário	Ação	Afetado
15/01/2018	david.junior	imprimir	formulario:87
15/01/2018	david.junior	novo	formulario:87
11/01/2018	david.junior	editar	solicitante:11
11/01/2018	david.junior	registro de autorizacao	formulario:86
11/01/2018	david.junior	imprimir	formulario:86
11/01/2018	david.junior	novo	formulario:86
05/01/2018	david.junior	desativar	solicitante:9
22/12/2017	ingrid.raiol	registro de autorizacao	formulario:85
22/12/2017	ingrid.raiol	imprimir	formulario:85
22/12/2017	ingrid.raiol	novo	formulario:85
22/12/2017	ingrid.raiol	registro de autorizacao	formulario:84
22/12/2017	ingrid.raiol	registro de autorizacao	formulario:84
22/12/2017	ingrid.raiol	imprimir	formulario:84
22/12/2017	ingrid.raiol	novo	formulario:84
21/12/2017	david.junior	editar	solicitante:12
21/12/2017	david.junior	editar	solicitante:12
18/12/2017	david.junior	registro de autorizacao	formulario:64
18/12/2017	david.junior	registro de autorizacao	formulario:63
18/12/2017	david.junior	registro de autorizacao	formulario:61
18/12/2017	david.junior	registro de autorizacao	formulario:60

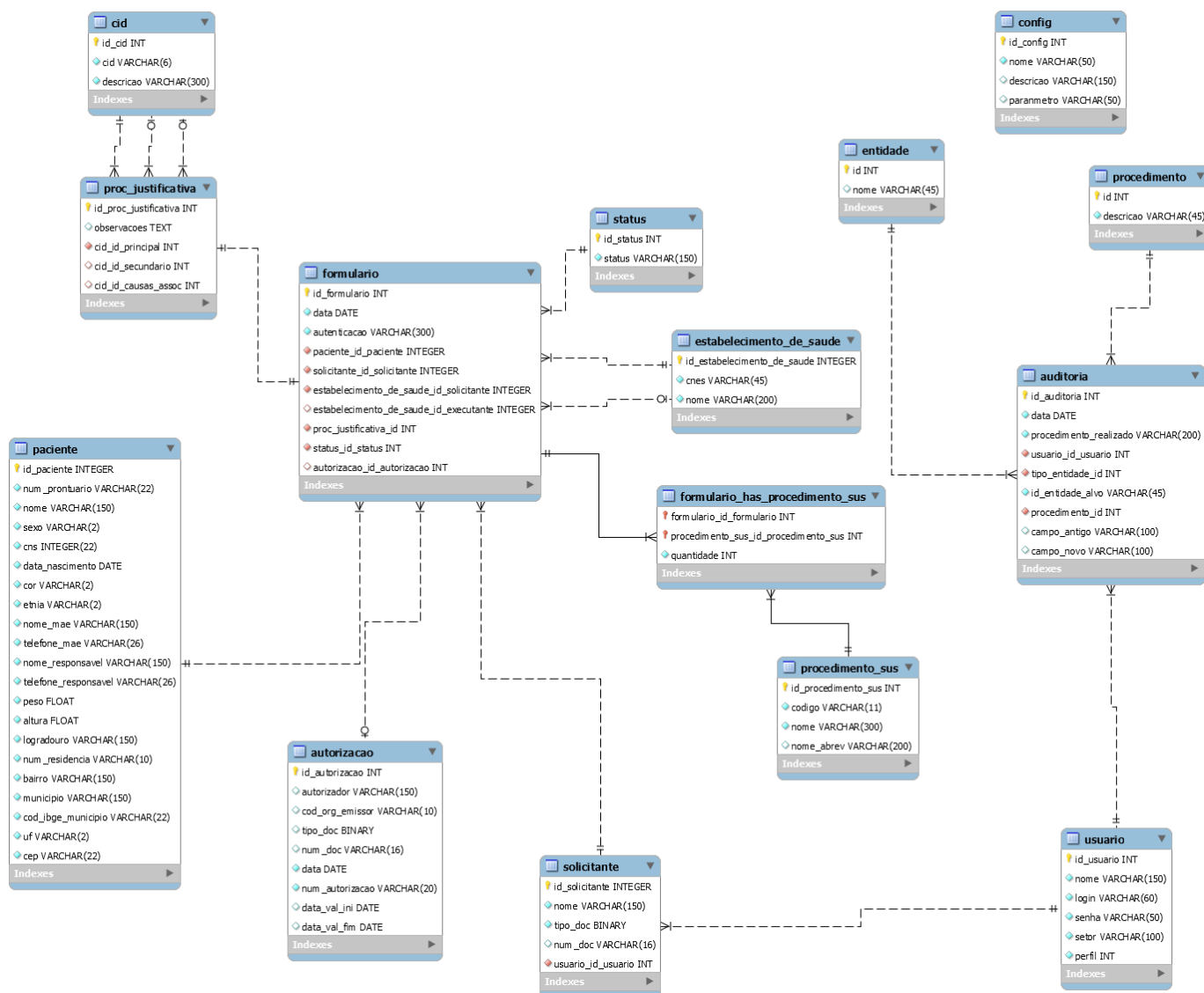
© 2013 Hospital Universitário João de Barros Barreto - Rua dos Mundurucus, 4487 - Guamá. CEP: 66073-000. CNPJ: 34.621.748/0004-76. Belém - PA - Brasil.

Figura 21. Tela de registro de atividades do APAC Barros Barreto.

A figura 21 demonstra a tela de registro de atividades da aplicação.

## 5. Projeto de banco de dados

A Figura 22 apresenta o diagrama relacional do sistema APAC Barros Barreto, usando uma extensão da UML para representar diagramas deste tipo.0,



**Figura 22. Diagrama de entidade relacional do sistema APAC Barros Barreto**

cid(**id\_cid**, cid, descricao);

**id\_cid** chave primária;

proc\_justificativa(**id\_proc\_justificativa**, observacoes, cid\_id\_principal,  
cid\_id\_secundario, cid\_id\_causas\_assoc);

**id\_proc\_justificativa** chave primária;

cid\_id\_principal chave estrangeira referenciando a chave primária **id\_cid** da  
tabela **cid**;

cid\_id\_secundario chave estrangeira referenciando a chave primária **id\_cid**  
da tabela **cid**;

cid\_id\_causas\_assoc chave estrangeira referenciando a chave primária  
**id\_cid** da tabela **cid**;

formulario(**id\_formulario**, data, autenticacao, paciente\_id\_paciente,  
solicitante\_id\_solicitante, estabelecimento\_de\_saude\_id\_solicitante,  
estabelecimento\_de\_saude\_id\_executante, proc\_justificativa\_id, status\_id\_status,  
autorizacao\_id\_autorizacao);

**id\_formulario** chave primária;

paciente\_id\_paciente chave estrangeira referenciando a chave primária  
**id\_paciente** da tabela **paciente**;

solicitante\_id\_solicitante chave estrangeira referenciando a chave primária  
**id\_solicitante** da tabela **solicitante**;

estabelecimento\_de\_saude\_id\_solicitante chave estrangeira referenciando a  
chave primária **id\_estabelecimento\_de\_saude** da tabela  
**estabelecimento\_de\_saude**;

estabelecimento\_de\_saude\_id\_executante chave estrangeira referenciando a  
chave primária **id\_estabelecimento\_de\_saude** da tabela  
**estabelecimento\_de\_saude**;

proc\_justificativa chave estrangeira referenciando a chave primária  
**id\_proc\_justificativa** da tabela **proc\_justificativa**;

status\_id\_status chave estrangeira referenciando a chave primária **id\_status**  
da tabela **status**;

autorizacao\_id\_autorizacao chave estrangeira referenciando a chave primária

**id\_autorizacao** da tabela **autorizacao**;

paciente(**id\_paciente**, num\_prontuario, nome, sexo, cns, data\_nascimento, cor, etnia, nome\_mae, telefone\_mae, nome\_responsavel, peso, altura, logradouro, num\_residencia, bairro, municipio, cod\_ibge\_municipio, uf, cep);

**id\_paciente** chave primária;

autorizacao(**id\_autorizacao**, autorizador, cod\_org\_emissor, tipo\_doc, num\_doc, data, num\_autorizacao, data\_val\_ini, data\_val\_fim);

**id\_autorizacao** chave primária;

solicitante(**id\_solicitante**, nome, tipo\_doc, num\_doc, usuario\_id\_usuario);

**id\_solicitante** chave primária;

usuario\_id\_usuario chave estrangeira referenciando a chave primária

**id\_usuario** da tabela **usuario**;

usuario(**id\_usuario**, nome, login, senha, setor, perfil);

**id\_usuario** chave primária;

procedimento\_sus(**id\_procedimento\_sus**, codigo, nome, nome\_abrev);

**id\_procedimento\_sus** chave primária;

formulario\_has\_procedimento\_sus(**formulario\_id\_formulario**, **procedimento\_sus\_id\_procedimento\_sus**, quantidade);

**formulario\_id\_formulario**, **procedimento\_sus\_id\_procedimento\_sus** chave primária composta;

formulario\_id\_formulario chave estrangeira referenciando a chave primária

**id\_formulario** da tabela **formulario**;

procedimento\_sus\_id\_procedimento\_sus chave estrangeira referenciando a chave primária **id\_procedimento\_sus** da tabela **procedimento\_sus**;

status(**id\_status**, status);

**id\_status** chave primária;

estabelecimento\_de\_saude(**id\_estabelecimento\_de\_saude**, cnes, nome);

**id\_estabelecimento\_de\_saude** chave primária;

entidade(**id**, nome);

**id** chave primária;

auditoria(**id\_auditoria**, data, procedimento\_realizado, usuario\_id\_usuario, tipo\_entidade\_id, id\_entidade\_alvo, procedimento\_id, campo\_antigo, campo\_novo);

**id\_auditoria** chave primária;

usuario\_id\_usuario chave estrangeira referenciando a chave primária **id\_usuario** da tabela **usuario**;  
tipo\_entidade\_id chave estrangeira referenciando a chave primária **id\_entidade** da tabela **entidade**;  
procedimento\_id chave estrangeira referenciando a chave primária **id** da tabela **procedimento**;  
procedimento(**id**, descricao);  
**id** chave primária;  
config(**id\_config**, nome, descricao, parametro);  
**id\_config** chave primária.

## 6. Projeto de componentes

### ● Especificação do Componente NOVO FORMULÁRIO

- 3) FUNÇÃO: Criar um novo formulário de solicitação de Autorização de Procedimentos Ambulatoriais de Alta Complexidade (APAC).
- 4) PSEUDOCÓDIGO DO PROGRAMA NOVO FORMULÁRIO

7. Início.

Fazer inicializações

Receber número de prontuário ou nome do paciente

localizar prontuário ou nome do paciente em BANCO DE DADOS DO AGHU

SE localizou o paciente ENTÃO

Preenche os dados do paciente no formulário

SENÃO

Exibir a mensagem “Prontuário não cadastrado no BANCO DE DADOS DO AGHU”

FIMSE

8. Receber código do procedimento ou nome do procedimento ambulatorial

localizar procedimento ou nome do procedimento em

PROCEDIMENTO\_SUS

SE localizou o procedimento ENTÃO

Preenche os dados do procedimento no formulário

SENÃO

Exibir a mensagem “Procedimento não cadastrado no banco de dados”

FIMSE

9. Receber código do cid ou nome do CID

localizar código ou nome do cid em CID

SE localizou o cid ENTÃO

Preenche os dados do cid no formulário

SENÃO

Exibir a mensagem “CID não cadastrado no banco de dados”

FIMSE

10. Executar verificarPreenchimento

SE todos os campos foram preenchidos de forma correta ENTÃO

Executar Salvar

SE o formulário foi salvo com sucesso em FORMULÁRIO

ENTÃO

Exibe a mensagem “Formulário N° (número do formulário) salvo com sucesso”

SENÃO

Exibe a mensagem “Erro ao salvar formulário no banco de dados”

FIMSE

SENÃO

Exibir mensagem “Por favor, preencha todos os campos corretamente”

FIMSE

11. Finalizações

12. Fim.

verificarPreenchimento

Início.

SE Id\_estabelecimento\_saude\_solicitante = -1 ENTÃO

Exibe a mensagem “Erro - Informe o estabelecimento de saúde solicitante!”

Retorna valor Falso

FIMSE

SE Num\_prontuario = Nulo ENTÃO

Exibe a mensagem “Erro - Informe os dados do paciente!”

Retorna valor Falso

FIMSE

SE codigo\_procedimento = Nulo ENTÃO

Exibe a mensagem “Erro - Informe os dados procedimento principal!”

Retorna valor Falso

SE Id\_cid\_principal = -1 ENTÃO

Exibe a mensagem “Erro – Informe o cid”

Fim.

## 6.1 Rotina Gráfica

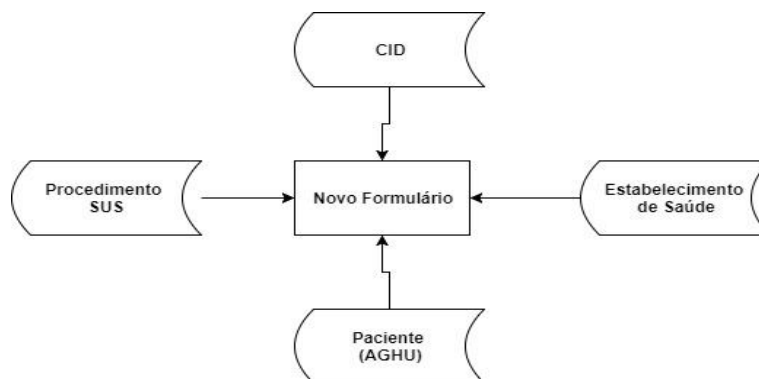


Figura 23. Rotina gráfica do componente Novo Formulário

## 13. Plano de Teste de Validação

Tipo de Teste:

**Teste de Validação de Software**

## 1 Abordagens do Teste

---

*A técnica utilizada neste plano de teste é teste baseado em casos de uso. Nesta técnica, os cenários de teste são encontrados a partir dos casos de uso do software, definidos no Documento de Especificação de Casos de Uso. Cada cenário de teste encontrado é levado em conta no momento dos testes.*

## 2 Definição do Ambiente

---

Recursos (Nome)	Tipo/Características
Sistema Operacional	Windows 7
Banco de Dados	MySQL
Navegador	Google Chrome

## 3 Envolvidos

---

Envolvido	Papel
Eliwelton Gomes Paes	Elaborador do Plano de testes Desenvolvedor.
David Barros de Oliveira Junior	Desenvolvedor e Testador

## 4 Definir Casos de Teste

---

Código	01	Nome	Logar no Sistema
<b>Caso de Uso/Componentes a serem testados</b>	UC1 – Logar no sistema.		

<b>Descrição</b>	Este teste tem como finalidade validar a funcionalidade que permite o usuário entrar com seu usuário e senha no sistema, este validando os dados permitindo acesso.
<b>Responsável</b>	Eliwelton Gomes Paes
<b>Pré-condições</b>	O usuário deve entrar com usuário e senha previamente cadastrados na aplicação.
<b>Pós-condições</b>	Nenhuma.

<b>Passos do Teste</b>
1 – Entrar com usuário e senha;
2 – O sistema autentica as credenciais do usuário;
3 – A página inicial do sistema é exibida.

<b>Cenários de Teste (de Caso de Uso)</b>	
<b>Cenário</b>	<b>Resultado Esperado</b>
Cenário 01: Entrando com usuário e senha previamente cadastrados no sistema.	O sistema redireciona para a tela principal.
Cenário 02: Entrando com usuário e senha que não estão cadastrados no sistema.	O sistema exibe a mensagem: “Usuário não encontrado”
Cenário 03: Entrando com caracteres especiais no usuário e senha.	O sistema exibe a mensagem: “Usuário não encontrado”

<b>Código</b>	02	<b>Nome</b>	Cadastrar Usuários
<b>Caso de Uso/Componentes a serem testados</b>	UC2 – Manter Usuários		
<b>Descrição</b>	Permite o cadastrado de usuários no sistema.		
<b>Responsável</b>	Eliwelton Gomes Paes		

<b>Pré-condições</b>	O usuário deve estar autenticado com o perfil Administrador.
<b>Pós-condições</b>	O sistema exibe uma mensagem de cadastro realizado com sucesso ou falha..

<b>Passos do Teste</b>	
1 - Selecionar a opção do menu principal <<Usuário>> em seguida clicar na opção <<Cadastrar>>;	
2 - O sistema exibe uma tela com os campos necessários para a criação de um novo usuário no sistema;	
3 - Adicionar nome, CPF, login, senha, CNS e selecionar o setor do usuário com seu respectivo perfil;	
4 - Confirmar a criação de usuário clicando no botão <<Cadastrar>>;	
5 - O sistema insere um novo usuário no sistema;	

<b>Cenários de Teste (de Caso de Uso)</b>	
<b>Cenário</b>	<b>Resultado Esperado</b>
Cenário 01: Cadastrar um novo usuário preenchendo corretamente todos os campos.	O sistema realiza o cadastro de um novo usuário no sistema.
Cenário 02: Cadastrando um novo usuário sem preencher um dos campos.	O sistema informa que o campo indicado precisa ser preenchido.
Cenário 03: Cadastrando um usuário com login e senha que não estão previamente cadastrados no sistema principal do hospital.	O sistema informa um erro: "Usuário não encontrado".

<b>Código</b>	03	<b>Nome</b>	Busca e Edição de usuários
<b>Caso de Uso/Componente</b>	UC2 – Manter Usuários		

<b>s a serem testados</b>	
<b>Descrição</b>	Permite a busca de edição de usuários do sistema.
<b>Responsável</b>	David Barros de Oliveira Junior
<b>Pré-condições</b>	O usuário deve estar autenticado com o perfil Administrador.
<b>Pós-condições</b>	Nenhuma.

<b>Passos do Teste</b>	
1 - Selecionar a opção do menu principal <<Usuário>> em seguida clicar na opção <<Busca e Edição>>.	
2 - O sistema exibe uma tela com o campo de busca e uma lista com todos os usuários cadastrados no sistema;	
3 – Digitar o nome de usuário o qual se deseja fazer a busca no campo específico;	
4 – O sistema exibe uma lista com o usuário especificado no campo;	
5 – Selecionar uma das opções no ao lado dos nomes dos usuários do sistema <<Visualizar>> <<Editar>> <<Desativar>>;	
6 – Ao selecionar a opção <<Visualizar>> uma janela contendo as informações do usuário é exibida;	
7 – Ao selecionar a opção <<Editar>> uma janela contendo os campos para edição é exibida;	
5 – Ao selecionar a opção <<Desativar>> uma janela de confirmação de desativação é exibida.	

<b>Cenários de Teste (de Caso de Uso)</b>	
<b>Cenário</b>	<b>Resultado Esperado</b>
Cenário 01: Buscar um usuário do sistema.	O sistema exibe uma lista com o usuário o qual está especificado no campo de busca.

Cenário 02: Editar um usuário.	O sistema conclui a alteração, atualizando os dados do usuário.
Cenário 03: Visualizar dados de um usuário.	O sistema exibe uma janela com os dados do usuário.
Cenário 04: Desativar usuário.	O sistema exibe uma janela com a confirmação de desativação do usuário. Caso confirmado, o sistema modifica a chave do usuário no banco de dados, fazendo com que ele não seja mais visualizado na lista de usuários. Caso não, o sistema retorna para a tela com a lista de usuários.

<b>Código</b>	04	<b>Nome</b>	Criação de um novo formulário de solicitação APAC
<b>Caso de Uso/Componentes a serem testados</b>	UC3 – Manter Formulário de Solicitação de Autorização de Procedimentos.		
<b>Descrição</b>	O usuário com perfil de Solicitante mantém o cadastro de um novo formulário de autorização de procedimento (Inclusão, edição e consulta de formulários).		
<b>Responsável</b>	Eliwelton Gomes Paes		
<b>Pré-condições</b>	O usuário precisa estar autenticado com o perfil Solicitante.		
<b>Pós-condições</b>	O sistema exibe uma mensagem de sucesso ou falha na operação.		

<b>Passos do Teste</b>
1 - Selecionar a opção <<+Novo>> no menu Formulário da aplicação;
2 - O sistema exibe uma tela contendo todos os campos padronizados de acordo com o formulário aceito pelo SUS.
3 - Adicionar os itens desejados que serão incluídos no formulário.
4 - Confirmar e salvar o formulário clicando no botão <<Salvar>>.

<b>Cenários de Teste (de Caso de Uso)</b>	
<b>Cenário</b>	<b>Resultado Esperado</b>
Cenário 01: Criar um novo formulário de solicitação de procedimentos.	O sistema processa todas as operações, salva o formulário e exibe o mesmo para impressão de acordo com o modelo aceito pelo SUS.

<b>Código</b>	05	<b>Nome</b>	Editar um formulário.
<b>Caso de Uso/Componentes a serem testados</b>	UC3 – Manter Formulário de Solicitação de Autorização de Procedimentos.		
<b>Descrição</b>	O usuário com perfil de Solicitante mantém o cadastro de um novo formulário de autorização de procedimento (Inclusão, edição e consulta de formulários).		
<b>Responsável</b>	Eliwelton Gomes Paes		
<b>Pré-condições</b>	O usuário deve estar autenticado com perfil Solicitante.		
<b>Pós-condições</b>	O sistema exibe a mensagem de dados atualizados.		

<b>Passos do Teste</b>
1 - Selecionar a opção <<Editar>> no menu Formulário da aplicação;
2 - O sistema exibe a tela de busca de formulário, o qual o usuário deve informar o número do formulário que deseja editar através da opção <<Buscar Formulário>>;
3 - O sistema exibe a tela de contendo os campos que podem ser editados no formulário;
4 - O sistema processa todas as alterações e salva o novo formulário.

<b>Cenários de Teste (de Caso de Uso)</b>	
<b>Cenário</b>	<b>Resultado Esperado</b>
Cenário 01: Editando um formulário com status Novo ou Salvo.	O sistema permite a edição das seções de Identificação do Paciente, Procedimento Solicitado, Procedimentos Secundários, Justificativa Do(s) Procedimento(s) Solicitado(s) do formulário, após a edição de qualquer um dos campos citados, o sistema registra as alterações e atualiza o formulário editado.
Cenário 02: Editando um formulário que esteja com status Impresso, Finalizado, Autorizado ou Não autorizado.	O sistema não permite edição de nenhum campo do formulário com estes status.

<b>Código</b>	06	<b>Nome</b>	Autorizar ou Não autorizar um Formulário
<b>Caso de</b>	UC4 – Registro de Autorização do Formulário de		

<b>Uso/Componentes a serem testados</b>	Autorização de Procedimento.
<b>Descrição</b>	Permite ao usuário com perfil Solicitante, a edição (conforme as regras descritas no seu respectivo caso de uso) de um formulário de solicitação de autorização de procedimento.
<b>Responsável</b>	Eliwelton Gomes Paes
<b>Pré-condições</b>	O usuário deve estar autenticado com perfil Administrativo.
<b>Pós-condições</b>	O sistema exibe a mensagem de autorização registrada com sucesso.

<b>Passos do Teste</b>
1 - O usuário seleciona a opção << Registrar Autorização>> no menu Formulário da aplicação;
2 - O sistema exibe uma tela com a opção <<Buscar formulário>>;
3 - O usuário preenche o campo o número de identificação do formulário que deseja fazer a autorização;
4 - O usuário preenche os campos da autorização, onde o campo N° de autorização (APAC) é o único campo obrigatório.

<b>Cenários de Teste (de Caso de Uso)</b>	
<b>Cenário</b>	<b>Resultado Esperado</b>
Cenário 01: Preenchendo todos os campos referentes a autorização ou não autorização de um formulário de solicitação APAC.	O sistema registra todas as operações, modificando o status do formulário para Autorizado ou Não autorizado.
Cenário 02: O usuário preenche os campos da autorização, exceto o campo N° de autorização	O sistema informa que o campo N° de Autorização é

(APAC).	obrigatório e precisa ser preenchido para prosseguir.

<b>Código</b>	07	<b>Nome</b>	Emitir relatórios do sistema
<b>Caso de Uso/Componentes a serem testados</b>	UC5 – Relatórios do Sistema.		
<b>Descrição</b>	Fornecimento do quantitativo (CID 10, Setor, Solicitante, Procedimento) de formulários autorizados e não autorizados pelo SUS.		
<b>Responsável</b>	David Barros de Oliveira Junior		
<b>Pré-condições</b>	O usuário deverá estar logado com o perfil Administrativo Financeiro.		
<b>Pós-condições</b>	Nenhuma.		

<b>Passos do Teste</b>
1 - O usuário seleciona a opção <<Relatórios>> no menu principal;
2 - O sistema exibe uma tela com as opções de relatórios que podem ser selecionadas: CID 10, Setor, Solicitante e Procedimento;
3 - O usuário indica a opção desejada: CID 10, Setor, Solicitante e Procedimento.
4 - O sistema fornece o quantitativo indicado.

<b>Cenários de Teste (de Caso de Uso)</b>	
<b>Cenário</b>	<b>Resultado Esperado</b>
Cenário 01: Emissão do quantitativo de CID 10.	O sistema fornece o quantitativo indicado.

Cenário 02: Emissão do quantitativo de Procedimentos solicitados.	O sistema fornece o quantitativo indicado.
Cenário 03: Emissão do quantitativo de Solicitantes.	O sistema fornece o quantitativo indicado.
Cenário 04: Emissão do quantitativo de Setores.	O sistema fornece o quantitativo indicado.

## 5 Resultados dos Testes

---

<b>Cenário</b>	<b>Resultado Obtido</b>
Código 01 – Cenário 01	Sucesso. O sistema logou no sistema com as credenciais autenticadas.
Código 01 – Cenário 02	Sucesso. O sistema apresentou a mensagem: “Usuário não encontrado”.
Código 01 – Cenário 03	Sucesso. O sistema apresentou a mensagem: “Usuário não encontrado”.
Código 02 – Cenário 01	Sucesso. O sistema realiza o cadastro de um novo usuário no sistema.
Código 02 – Cenário 02	Sucesso. O sistema informa que o campo indicado precisa ser preenchido.
Código 02 – Cenário 03	Sucesso. O sistema informou a mensagem: “Usuário não encontrado”.
Código 03 – Cenário 01	Sucesso. O sistema exibiu a lista com o usuário o qual está indicado no campo de busca.
Código 03 – Cenário 02	Sucesso. O sistema conclui a alteração e atualizando os dados do usuário.
Código 03 – Cenário 03	Sucesso. O sistema exibe a janela com os dados do usuário.
Código 03 – Cenário 04	Sucesso. O sistema exibe uma janela com a confirmação de desativação do usuário.

Código 04 – Cenário 01	Sucesso. O sistema salva o formulário e exibe o mesmo para impressão de acordo com o modelo aceito pelo SUS.
Código 05 – Cenário 01	Sucesso. O sistema permitiu a edição das seções de Identificação do Paciente, Procedimento Solicitado, Procedimentos Secundários, Justificativa Do(s) Procedimento(s) Solicitado(s) do formulário.
Código 05 – Cenário 02	Sucesso. O sistema não permite edição de nenhum campo do formulário com status que não sejam Novo e Salvo.
Código 06 – Cenário 01	Sucesso. O sistema registrou todas as operações e modificou o status do formulário para Autorizado ou Não autorizado.
Código 06 – Cenário 02	Sucesso. O sistema informa que o campo N° de Autorização é obrigatório.
Código 07 – Cenário 01	Sucesso. O sistema fornece o quantitativo indicado.
Código 07 – Cenário 02	Sucesso. O sistema fornece o quantitativo indicado.
Código 07 – Cenário 03	Sucesso. O sistema fornece o quantitativo indicado.
Código 07 – Cenário 04	Sucesso. O sistema fornece o quantitativo indicado.

## 6 Resumo dos Testes

---

*Os testes validação indicaram que todas as funcionalidades do sistema estão funcionando corretamente.*

## APÊNDICE E – QUESTIONÁRIO DE AVALIAÇÃO ONLINE

# Formulário de Avaliação - APAC Barros Barreto

1. Endereço de e-mail \*

\_\_\_\_\_

2. Usabilidade: Ao acessar o sistema APAC Barros Barreto, é possível identificar o objetivo da ferramenta?

*Marcar apenas uma oval.*

- Sim
- Sim, porém algumas coisas não ficaram claras
- Não

3. Usabilidade: O APAC Barros Barreto possui uma aparência agradável e intuitiva?

*Marcar apenas uma oval.*

- Sim, sua aparência é agradável e consigo entender as suas funcionalidades pelo designer;
- Sim, sua aparência é agradável mas não consigo entender exatamente as suas funcionalidades;
- Não é agradável, mas é intuitivo;
- Não.

4. Funcionalidade: O APAC Barros Barreto cumpre o que promete fazer, ou seja, fornece a criação de um formulário APAC e imprime no padrão SUS?

*Marcar apenas uma oval.*

- Sim;
- Sim, em partes;
- Não;
- Outro: \_\_\_\_\_

5. Funcionalidade: Em algum momento, durante o uso do sistema, ocorreu alguma falha inesperada?

*Marcar apenas uma oval.*

- Nenhuma vez;
- 1 vez;
- 2 vezes;
- Entre 5 e 10 vezes;
- Mais de 10 vezes;

6. Confiabilidade: Todos os botões do APAC Barros Barreto direcionam para páginas corretas?

*Marcar apenas uma oval.*

- Sim;
- Não.

**7. Confiabilidade: O sistema consegue orientar durante sua utilização? Isto é, caso você não consiga executar determinada tarefa, o sistema indica o que pode estar acontecendo?**

*Marcar apenas uma oval.*

- Sim;
- Algumas vezes, sim;
- Não.

**8. Eficiência: O APAC Barros Barreto consegue processar as requisições com tempo de resposta aceitável?**

*Marcar apenas uma oval.*

- Sim, estou satisfeito com o tempo de resposta;
  - Ele demora um pouco com algumas requisições;
  - Não, demora bastante.
-