

LIGA DE ENSINO DO RIO GRANDE DO NORTE
CENTRO UNIVERSITÁRIO DO RIO GRANDE DO NORTE PÓS-GRADUAÇÃO EM
DESENVOLVIMENTO DE SISTEMAS CORPORATIVOS

TÁCITO BARRETO NETO

**PROPOSTA DE DESENVOLVIMENTO DE UMA FERRAMENTA DE
ATENDIMENTO MÉDICO NA WEB**

NATAL/RN
2012

TÁCITO BARRETO NETO

**PROPOSTA DE DESENVOLVIMENTO DE UMA FERRAMENTA DE
ATENDIMENTO MÉDICO NA WEB**

Trabalho de conclusão de curso apresentado como parte dos requisitos para obtenção do grau de Especialista Desenvolvimento de Sistemas Corporativos na Faculdade Natalense para o Desenvolvimento do Rio Grande do Norte.

Orientador: Prof. Esp. Romulo Fagundes Cantanhede.

NATAL/RN
2012

TÁCITO BARRETO NETO

**PROPOSTA DE DESENVOLVIMENTO DE UMA FERRAMENTA DE
ATENDIMENTO MÉDICO NA WEB**

Trabalho de conclusão de curso apresentado
como parte dos requisitos para obtenção do
grau de Especialista em Desenvolvimento de
Sistemas Corporativos na Faculdade
Natalense para o Desenvolvimento do Rio
Grande do Norte.

Aprovada em: ____/____/____

BANCA EXAMINADORA

Romulo Cantanhede Fagundes (Esp)
Orientador

RESUMO

A proposta de desenvolvimento e documentação do sistema da Liga Contra o Câncer em plataforma web busca o avanço tanto nas tecnologias a serem utilizadas, como nos padrões adotados para os mesmos serem elaborados e executados, visando uma ampla iteratividade do sistema com os atores do cenário proposto que são pacientes, médicos, usuários e o próprio hospital, gerando uma maior escalabilidade de manutenção e crescimento dos processos existentes.

Palavras-chave: Java. Sistemas de informação. Scrum.

ABSTRACT

The proposition for development and documentation of Liga Contra o Cancer system in web platform searches for an advance in both, technology to be used and the patterns adopted to elaborate and execute it, targeting a wide iterability of the system with the proposed scenario actors, such as patients, doctors, users and the hospital itself, generating greater scalability maintenance and growth of existing processes.

Keywords: Java. Information Systems. Scrum.

LISTA DE FIGURAS

Figura 1 - Diagrama Entidade Relacionamento levantados do sistema atual.	10
Figura 2 - Swing interface para os administradores.	17
Figura 3 - Diagrama Representativa do MVC.....	19
Figura 4 - Representação Scrum.	23
Figura 5 – Cenário de Cadastro de paciente.....	25
Figura 6 - Cenário de Agendamento de Paciente.	27
Figura 7 – Cenário Registro Atendimento.	29
Figura 8 – Cenário Registro Serviço de Apoio.	33
Figura 9 – Cenário Atendimento Paciente.....	35
Figura 10 – Cenário de Monitoramento de Paciente.	37
Figura 11 – Cenário Geração Número Ficha.....	39
Figura 12 – Cenário Protocolo Arquivo Paciente.....	42
Figura 13 – Cenário Alteração Atendimento.	45
Figura 14 – Cenário Geração Alta/Encaminhamento Paciente.	47
Figura 15 - Diagrama de Classes.....	49
Figura 16 - Menu Responsável pelo processo de atendimento.....	52
Figura 17 - Tela de Monitoração de Atendimento	52
Figura 18 - Tela de Atendimento.	53
Figura 19 - Tela de cadastro do paciente	54

SUMÁRIO

1. INTRODUÇÃO	7
2. Objetivos	9
2.1. Objetivo Geral.....	9
2.2. Objetivos Específicos	9
2.2.1. DER(Diagrama Entidade Relacionamento) atual dos requisitos levantados.	10
3. Justificativa.....	11
4.Referencial Teórico	13
4.1. Introdução à plataforma Java.....	13
4.2. Historia do Java.....	13
4.3. JSF(JAVASERVER FACES).....	14
4.4. Hibernate.....	16
4.5. MVC	16
4.5.1. Contexto	16
4.5.2. Problema	17
4.5.3. Forças	18
4.5.4. Solução	18
4.5.5. Estrutura.....	19
4.5.6. Participantes e Responsabilidades	19
4.5.7. Estratégias	20
4.5.8. Consequências.....	21
4.6. SQL SERVER	21
4.7. Eclipse.....	22
4.8. Astah	22
4.9. Metodologia scrum	22
5. Caso de uso	24
5.1. Contextualização da empresa.....	24
6. Casos de uso, Cenário Cadastro Paciente.....	25
6.1. Descrição do caso de uso Cadastrar Paciente.....	25
7. Casos de uso, Cenário Agendamento Paciente.....	27
7.1. Descrição do caso de uso Agendar Paciente.....	27

8. Casos de uso, Cenário Registro Atendimento.....	29
8.1. Descrição do caso de uso Registrar Atendimento.....	30
9. Casos de uso, Cenário Registrar Serviço de Apoio	33
9.1. Descrição Caso de uso Registrar Serviço de Apoio.....	33
10. Casos de uso, Cenário Atender Paciente.....	35
10.1. Descrição caso de uso Atender Paciente.....	35
11. Casos de uso, Cenário Monitorar Atendimento.....	37
11.1. Descrição caso de uso Atender Paciente.....	37
12. Casos de uso, Cenário Gerar Número de Ficha	39
12.1. Descrição caso de uso Gerar Número de Ficha.....	40
13. Casos de uso, Cenário Protocolo Arquivo Paciente	42
13.1. Descrição caso de uso Protocolar Arquivo Paciente.....	43
14. Casos de uso, Alterar Atendimento	45
14.1. Descrição caso de uso Alterar Atendimento.....	45
15. Casos de uso, Gerar Alta/Encaminhamento Paciente	47
15.1. Descrição caso de uso Gerar Alta/Encaminhamento Paciente	47
16. Diagrama de Classes	49
17. Processo de Atendimento	49
18. Situação Atual	50
18.1. Sistema Atual	50
18.2. Telas do Sistema Atual.....	51
19. Proposta.....	54
20. Escopo do Projeto	55
20.1. Tempo do Projeto versus Custo	55
20.1.1. Quadro atual de dois funcionários.....	56
20.1.2. Quadro futuro com três funcionários	56
20.2. Plano de Migração	58
21. Conclusão	59
22. REFERÊNCIAS.....	60

1. INTRODUÇÃO

Os sistemas corporativos estão cada vez mais presentes nas empresas de mais variados campos, pela necessidade de uma gestão mais eficaz, confiabilidade e agilidade nas informações. As primeiras aplicações de tecnologia da informação para o apoio à gestão empresarial foram construídas totalmente sob demanda, de acordo com as necessidades de cada empresa.

Na década de 70(setenta) o ambiente de desenvolvimento era o mainframe, com grandes equipes de desenvolvimento e projetos de longo prazo que demandavam grandes investimentos. Com o passar do tempo, começaram a surgir sistemas prontos no mercado que puderam ser aplicados a múltiplas empresas. Era o surgimento dos softwares de gestão empresarial. Muito tímidos inicialmente, eram ilhas de automação em empresas extremamente carentes de suporte adequado de sistemas. Os primeiros sistemas prontos surgiram para atender as áreas administrativo-financeiras, nas funções de contabilidade e tesouraria.

Pouco a pouco esses diversos sistemas isolados foram convergindo para soluções integradas, evitando o processo extremamente trabalhoso de integração através de “*interfaceamento*”. Os sistemas passaram a compartilhar bases de dados comuns que permitiram a disponibilização da mesma informação para todas as áreas da empresa.

As funcionalidades desses sistemas foram avançando. Na década de 70(setenta) já surgem os primeiros sistemas integrados com funcionalidades industriais. A técnica do MRP, utilizada em aplicativos isolados foi integrada a módulos que faziam o controle de estoques, a gestão dos pedidos de vendas e de compras e outros.

Nos anos 80(oitenta) houve grande aumento no número de fornecedores de softwares integrados de gestão e o crescimento crescente da sua funcionalidade. Esses sistemas passaram a ser identificados com a sigla ERP – Enterprise Resource Planning.

Em paralelo a esse crescimento uma área que vem crescendo de forma bastante acentuada, é a Informática Médica ou Informática em Saúde (em Inglês Medical Informatics) é definida por Blois e Shortliffe (1990) como "um campo de

rápido desenvolvimento científico que lida com armazenamento, recuperação e uso da informação, dados e conhecimentos biomédicos para a resolução de problemas e tomada de decisão".

"A Saúde é uma das áreas onde há maior necessidade de informação para a tomada de decisões. A Informática Médica é o campo científico que lida com recursos, dispositivos e métodos para otimizar o armazenamento, recuperação e gerenciamento de informações biomédicas. O crescimento da Informática Médica como uma disciplina deve-se, em grande parte: aos avanços nas tecnologias de computação e comunicação, à crescente convicção de que o conhecimento médico e as informações sobre os pacientes são ingerenciáveis por métodos tradicionais baseados em papel, e devido à certeza de que os processos de acesso ao conhecimento e tomada de decisão desempenham papel central na Medicina moderna. "

Grande parte dos SIS (Sistemas de Informação em Saúde), são desenvolvidos em aplicativos Desktop, de forma cliente servidor. Porém com a expansão da internet, avanço em equipamentos como notebooks, tablet's e smartphones existem uma carência nesses sistemas na web.

2. Objetivos

2.1. Objetivo Geral

A proposta aqui apresentada é de desenvolvimento e documentação de uma ferramenta em plataforma Web, que irá auxiliar no atendimento dos pacientes da Liga Norte-Rio-Grandense contra o Câncer. Visando atender as 3(três) unidades de Natal-RN , e uma que se encontra no município de Caicó-RN.

2.2. Objetivos Específicos

- Levantamento da situação atual dos sistemas utilizados pela LIGA.
- Levantamento de requisitos, de um sistema de gerenciamento web, para controle das informações utilizado no sistema atual.
- Proposta de desenvolvimento, e sugestão de tecnologias utilizadas para o desenvolvimento da plataforma web.

3. Justificativa

A Liga Contra o Câncer tem em média 1200(um mil e duzentos) funcionários de mais variados cargos que utilizam o sistema em seu dia-a-dia.

A central de marcação recebe cerca de 1700(um mil e setecentos) ligações ao dia fazendo entre 1200(um mil e duzentos) a 1500(um mil e quinhentos) agendamentos. O setor de recepção onde é gerado o atendimento do paciente é o que precisa de mais agilidade para melhor atendimento, que fica em torno de 1700 atendimentos ao dia.

Esses números representam a utilização de apenas 2(dois) módulos do sistema, no entanto o sistema contém os módulos de: Caixa, módulo onde ocorre os pagamentos dos serviços. Prescrição, que gera os pedidos de materiais e medicamentos à farmácia, que também tem um módulo onde são confirmadas as prescrições para montagem de kits, devoluções, relatórios estratégicos. PEP(Prontuário Eletrônico do Paciente) onde o médico têm o controle médico do paciente. Centro-cirúrgico, que gerencia o controle das cirurgias. Patologia clínica, responsável pela digitação dos laudos de análises clínicas e disponibilização dos mesmos, Esterilização, que controla toda movimentação de materiais cirúrgicos entre as unidades de atendimento médico e próprio setor de esterilização. Faturamento, onde são faturadas as contas médicas para envio de produção aos convênios. Auditoria de contas, que revisam todas as contas digitadas pelo faturamento. Financeiro, que controla as contas à pagar e receber. Solicitação de material, onde todos os setores utilizam para consumo de materiais de expediente e diversos. Almoxarifado, onde chegam as solicitações e são feita as liberações de materiais para os setores. Compras onde são administradas as ordens de compras. Suprimentos, que controla o estoque. Chefia de recepção para levantamento de dados pra tomadas de decisões e gestor, módulo totalmente voltado para o gerenciamento de acesso de usuários, setores, máquinas e outros, onde somente a T.I (Tecnologia da Informação) tem acesso.

Sendo assim, o sistema é um dos órgãos vitais para funcionamento do hospital, porém existem vários problemas a sanar. Pelo fato do sistema ter sido desenvolvido em plataforma desktop, há uma série de procedimentos a executar em

uma chegada de computadores novos ou formatação dos mesmos, o que dificulta na liberação dos tais para os setores. Com perda de conexão entre as unidades, o sistema não pode ficar off-line por causa da centralização da base e falta de replicação de servidores. Outro problema dar-se a continuidade do sistema, por não haver a documentação, dificultando o entendimento de novos profissionais tanto da parte de codificação quanto a de análise de requisitos. Com uma nova análise dos processos diante do cenário tecnológico atual, é possível desenvolver uma ferramenta mais dinâmica, inteligente, rápida, documentada e com alta disponibilidade (24/7), tudo isso utilizando técnicas de análise com diagramas de UML, mapeamento das classes para ser possível a utilização de variados bancos de dados dependendo da necessidade, programação em Java utilizando framework JSF, possibilitando a reutilização de código e trabalhando no padrão MVC (Model-View-controller).

4.Referencial Teórico

4.1. Introdução à plataforma Java.

Java é uma linguagem de programação orientada a objetos desenvolvida na década de 90(noventa) por uma equipe de programadores chefiada por James Gosling, na empresa Sun Microsystems. Diferentemente das linguagens convencionais, que são compiladas para código nativo, a linguagem Java é compilada para um "bytecode" que é executado por uma máquina virtual. A linguagem de programação Java é a linguagem convencional da Plataforma Java, mas não sua única linguagem.

4.2. Historia do Java

Os microprocessadores têm um impacto profundo em dispositivos inteligentes eletrônicos voltados para o consumidor. Reconhecendo isso, a Sun Microsystems, em 1991, financiou um projeto de pesquisa corporativa interna com codinome Green, que resultou no desenvolvimento de uma linguagem baseada em C++ que seu criador, James Gosling, chamou de Oak em homenagem a uma árvore de carvalho vista por sua janela na Sun. Descobriu-se mais tarde que já havia uma linguagem de computador chamada Oak. Quando uma equipe da Sun visitou sua cafeteria local, o nome Java (cidade de origem de um tipo de café importado) foi sugerido.

O projeto Green passou por algumas dificuldades. O mercado de dispositivos eletrônicos inteligentes voltados para o consumo popular não estava se desenvolvendo, no início da década de 1990, tão rápido como a Sun havia antecipado. O projeto corria o risco de ser cancelado. Por uma casualidade, a World Wide Web explodiu em popularidade em 1993 e a equipe da Sun viu o imediato potencial de utilizar o Java para adicionar conteúdo dinâmico, como interatividades e animações, às páginas da Web. Isso deu nova vida ao projeto.

A Sun anunciou o Java formalmente em uma importante conferência em maio de 1995. O Java chamou a atenção da comunidade de negócios por causa do

enorme interesse na Word WideWeb. O Java é agora utilizado para desenvolver aplicativos corporativos de grande porte, aprimorar a funcionalidade de servidores Web, fornecer aplicativos para dispositivos voltados para o consumo popular (por exemplo, telefones celulares, pagers e PDAs) e para muitos outros propósitos.

4.3. JSF(JAVASERVER FACES)

JSF é uma tecnologia que incorpora características de um framework MVC para WEB e de um modelo de interfaces gráficas baseado em eventos. Por basear-se no padrão de projeto MVC, uma de suas melhores vantagens é a clara separação entre a visualização e regras de negócio (modelo).

A tecnologia JavaServer Faces inclui:

- Um conjunto de APIs para representar componentes UI e gerenciar seu estado, tratamento de eventos e de validação de entrada, definição de navegação de página e apoiar a internacionalização e acessibilidade;
- O JavaServer Pages (JSP) biblioteca de marcas personalizadas para expressar uma interface de JavaServer Faces dentro de uma página JSP.

Projetado para ser flexível, o JavaServer Faces utiliza a tecnologia UI padrão existente e conceitos web-tier, não limitando os desenvolvedores a uma linguagem de marcação especial, protocolo ou dispositivo cliente. As classes de componentes UI, incluído a tecnologia JavaServer Faces encapsula as funcionalidades dos componentes não específicos do cliente, possibilitando assim os componentes UI JavaServer Faces a serem prestados para vários dispositivos de cliente. Ao combinar a funcionalidade do componente de UI com renderizadores personalizados que definem os atributos de renderização de um componente UI específico, os desenvolvedores podem construir marcas personalizadas a um dispositivo cliente em particular. Como uma conveniência, a tecnologia JavaServer Faces fornece um processador personalizado e uma biblioteca de tags JSP personalizadas para processamento de um cliente de HTML, permitindo que os desenvolvedores de Java

Platform, Enterprise Edition (Java EE) possam usar a tecnologia JavaServer Faces em suas aplicações.

Como objetivo principal, a facilidade de uso, a arquitetura do JavaServer Faces define claramente uma separação entre a lógica da aplicação e apresentação, sendo fácil de conectar a camada de apresentação para o código do aplicativo. Este projeto permite que cada membro de uma equipe de desenvolvimento de aplicações web possam se concentrar em sua peça do processo de desenvolvimento, e também fornece um modelo de programação simples para ligar as peças. Por exemplo, os desenvolvedores de páginas web sem conhecimentos de programação podem usar as tags JavaServer Faces componente UI para vincular código da aplicação a partir de uma página web sem escrever qualquer script.

Desenvolvida através do Java Community Process sob JSR - 314, a tecnologia JavaServer Faces estabelece o padrão para a construção de interfaces de usuário do lado do servidor. Com as contribuições do grupo de peritos, o JavaServer Faces APIs estão sendo projetados para que possam ser aproveitados por ferramentas que tornarão o desenvolvimento de aplicações web ainda mais fácil. Vários fornecedores de ferramentas foram respeitados e membros do grupo de peritos JSR-314, que desenvolveu a especificação JavaServer Faces 1.0. Estes fornecedores estão empenhados em apoiar a tecnologia JavaServer Faces em suas ferramentas, promovendo assim a adoção do JavaServer Faces padrão de tecnologia.

4.4. Hibernate

O Hibernate, como todas as outras ferramentas de mapeamento objeto / relacional, requer metadados que governam a transformação dos dados de uma representação para outro. Hibernate Annotations oferece anotação baseada em metadados de mapeamento.

A especificação do JPA reconhece o interesse e sucesso do objeto transparente / paradigma relacional. Ele padroniza as APIs básicas e os metadados necessários para qualquer objeto / relacional, mecanismo de persistência. *Hibernate Entity Manager* implementa as interfaces de programação e as regras de ciclo de vida, tal como definido pela especificação de persistência JPA e, juntamente com *Hibernate Annotations* oferece um completo (independente) para solução de persistência JPA no topo do Núcleo Hibernate maduro.

4.5. MVC

4.5.1. Contexto

Como exemplo, um aplicativo apresenta um conteúdo em numerosas páginas contendo diversos dados. Além disso, a equipe de engenharia responsável pela concepção, implementação e manutenção do aplicativo é composto de indivíduos com diferentes habilidades.

4.5.2. Problema

Agora, mais do que nunca, aplicações empresariais têm a necessidade de apoiar vários tipos de usuários com vários tipos de interfaces. Por exemplo, uma loja online pode exigir uma frente HTML para clientes Web, uma frente WML para clientes sem fio, um Java™ (JFC) / Swing interface para os administradores, e um serviço Web baseado em XML para fornecedores. A figura 2(dois) representa essa descrição.

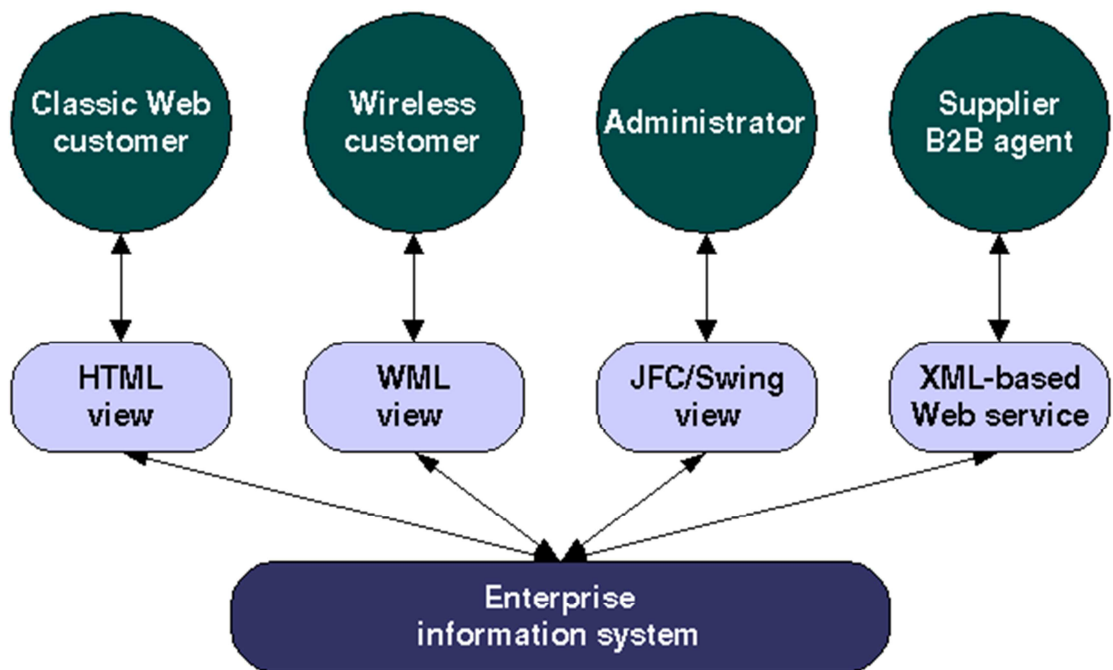


Figura 2 - Swing interface para os administradores.

FILIPPE-PINTO, **Model-View-Controller Architecture**. Disponível em: <<http://www.filipe-pinto.com/books/Software-Engineering/mvc-model-view-controller-architecture>> Acesso em 12 dez. 2011.

Ao desenvolver uma aplicação para suportar um único tipo de cliente, às vezes é benéfico para entrelaçar acesso a dados e lógica de regras de negócio com interface lógica específica para apresentação e controle. Tal abordagem, no entanto, é inadequada quando aplicado a sistemas corporativos que precisam suportar múltiplos tipos de clientes. Diferentes aplicações precisam ser desenvolvidas, uma para cada tipo de suporte e interface do cliente. Não específicos de interface, o código é duplicado em cada aplicação, resultando em esforços duplicados na implementação (muitas vezes da variedade de copiar e colar), bem como testes e manutenção. A tarefa de determinar o que é caro para duplicar em si, uma vez interface específica e não interface de código específico está interligada. Duplicações dos esforços são inevitavelmente imperfeitas. Lentamente, mas seguramente, os aplicativos que são supostos para fornecer a mesma funcionalidade evoluir para sistemas diferentes.

4.5.3. Forças

- Os dados de uma empresa precisam ser acessados quando se apresenta em diferentes pontos de vista: por exemplo, HTML, WML, JFC / Swing, XML;
- Os dados de uma empresa precisam ser atualizados por meio de interações diferentes: por exemplo, as seleções de um link em uma página HTML ou de um cartão WML, cliques de botão em uma GUI JFC / Swing, mensagens SOAP escrito em XML;
- Apoiar vários tipos de pontos de vista e as interações não deveria afetar os componentes que fornecem a funcionalidade principal do aplicativo corporativo.

4.5.4. Solução

Através da aplicação do Model-View-Controller arquitetura (MVC) para um JavaTM 2 Platform, Enterprise Edition (J2EETM aplicação), você separa do núcleo da funcionalidade do modelo de negócio da apresentação e lógica de controle que utiliza essa funcionalidade. Tal separação permite múltiplas visões de compartilhar o

mesmo modelo de dados corporativos, o que dá mais apoio de vários clientes fáceis de implementar, testar e manter.

4.5.5. Estrutura

Como é possível ver na figura 3(três), uma representação do padrão Model-View-Controller.

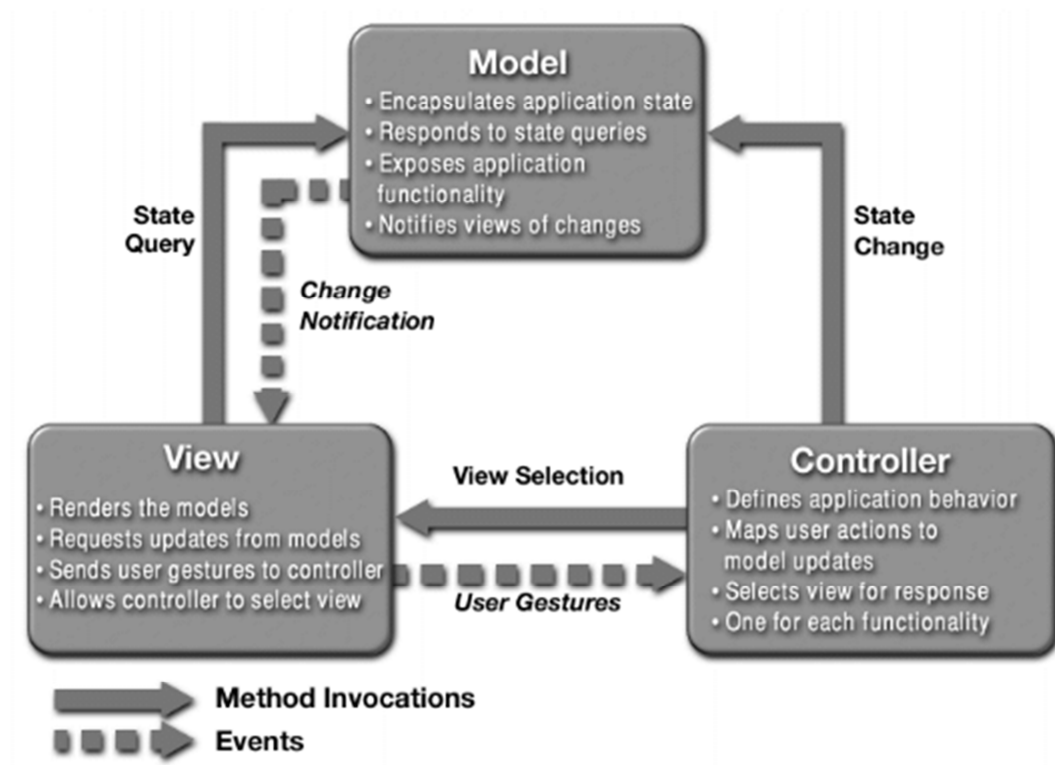


Figura 3 - Diagrama Representativa do MVC

ORACLE, **MVC**. Disponível em: <<http://www.oracle.com/technetwork/articles/javase/mvc-136693.html#1>> Acesso em 15 dez. 2011.

4.5.6. Participantes e Responsabilidades

A arquitetura MVC tem suas raízes no Smalltalk, onde foi originalmente aplicada para mapear a entrada tradicional, processamento e saída de tarefas para o modelo de gráfica do usuário interação. No entanto, é fácil mapear esses conceitos no domínio de aplicações multi-tier empresa.

- **Modelo** - O modelo representa os dados da empresa e as regras de negócio que governam o acesso e atualizações de dados. Muitas vezes, o modelo serve como uma aproximação a um processo de software do mundo real, tão simples técnicas de modelagem do mundo real se aplicam quando a definição do modelo;
- **Ver** - O ponto de vista torna o conteúdo de um modelo. Ele acessa os dados da empresa através do modelo e especifica como os dados devem ser apresentados. É de responsabilidade da visão para manter a consistência na sua apresentação, quando o modelo é alterado. Isto pode ser conseguido através de um modelo push, onde a vista se registra com o modelo de notificações de alterações, ou um modelo pull, onde a visão é responsável por chamar o modelo quando for necessário para recuperar os dados mais atuais;
- **Controlador** - O controlador traduz interações com a visão em ações a serem realizadas pelo modelo. Em um cliente GUI stand-alone, as interações do usuário podem ser cliques de botão ou seleções de menu, enquanto que em uma aplicação Web, eles aparecem como solicitações GET e POST HTTP. As ações realizadas pelo modelo incluem ativar processos de negócio ou alterar o estado do modelo. Com base na interação do usuário e o resultado das ações do modelo, o controlador responde selecionando uma visão adequada.

4.5.7. Estratégias

- Web-based clientes, tais como navegadores. JavaServer Pages TM (JSP TM) páginas para tornar o ponto de vista Servlet como o controlador, e Enterprise JavaBeans TM (EJB TM) componentes como o modelo;
- Centralizar o controlador. Em vez de ter múltiplos servlets como controladores, um Servlet principal é usado para tornar o controle mais gerenciável. O Front Controller padrão descreve essa estratégia com mais detalhes.

4.5.8. Consequências

- Reutilização de componentes de modelo. A separação do modelo e a visualização permite múltiplas visões para usar o modelo de empresa. Consequentemente, os componentes de um aplicativo corporativo do modelo são mais fáceis de implementar, testar e manter, uma vez que todo o acesso ao modelo passa por esses componentes;
- Suporte mais fácil para novos tipos de clientes. Para suportar um novo tipo de cliente, basta escrever uma opinião e algumas lógicas do controlador e conectá-los para o aplicativo empresarial existente.
- Aumento da complexidade do projeto.

4.6. SQL SERVER

O SGBD(Sistema Gerenciador de Banco de Dados) Sql Server está presente na liga contra o câncer desde da versão 2000. Ferramenta versátil, que proporciona integridade e rapidez nas informações. Uma das vantagens é a rede de usuários que utiliza, proporcionando mais segurança ao futuro do sistema.

O Sql Server é muito mais do que um simples banco de dados utilizados para armazenar dados. Dentro da plataforma de dados do Sql Server existem vários serviços que podem ser usados para construir qualquer aplicativo concebível dentro de uma empresa, o que dar-se muita segurança a utilização do mesmo no hospital da liga contra o Câncer. Dentro do mecanismo de banco de dados central, você encontrará serviços para armazenar, manipular, fazer backup e restaurar dados. O mecanismo de banco de dados central também contém recursos de segurança avançados para proteger seus investimentos, junto com serviços para garantir a máxima disponibilidade. Sua infraestrutura de dados pode ser ampliada para manipular texto não estruturado, junto com o sincronismo de várias cópias de um banco de dados.

4.7. Eclipse

Plataforma de desenvolvimento da linguagem Java, é considerada por uma grande maioria de desenvolvedores Java como a melhor IDE para a linguagem Java, conta a característica de ser uma IDE rápida em relação a outra IDE como por exemplo o NetBeans, tanto em carregamento como uso de seus assistentes de código, possuindo uma grande gama de plug-ins, podendo se adaptar a qualquer situação de desenvolvimento.

4.8. Astah

A ferramenta Jude Community é uma boa ferramenta e modelagem UML gratuita. Por ser uma versão Community possui algumas limitações. Há uma versão Professional disponível, mas os recursos presentes na versão Community podem suprir a necessidade de grande parte dos artefatos necessários no dia-a-dia.

Dentre os recursos da ferramenta estão:

- Suporte a UML 2.1;
- Diagramas de Classe, Caso de Uso, Sequência, Atividade, Comunicação, Máquina de Estado, Componentes, Implantação, Estrutura de Composição, Objetos e Pacotes;
- Ajustes de alinhamento e tamanho dos diagramas;
- Impressão dos diagramas (com a marca d'água da ferramenta);
- Exportação das imagens dos diagramas (com a marca d'água da ferramenta).

4.9. Metodologia scrum

Scrum é um framework ágil para a conclusão de projetos complexos. Scrum originalmente foi formalizada para projetos de desenvolvimento de software, mas funciona bem para qualquer escopo de trabalho. As possibilidades são infinitas. O framework Scrum é enganosamente simples.

O Quadro de Scrum em 30 segundos:

- Um proprietário do produto cria uma lista de desejos priorizados chamado product backlog;
- Durante planejamento do sprint , a equipe pega um pequeno pedaço da parte superior do que na lista de desejos, um backlog de sprint , e decide como implementar essas peças;
- A equipe tem certa quantidade de tempo da Sprint , para completar o seu trabalho - geralmente duas a quatro semanas, mas atende a cada dia para avaliar o seu progresso (scrum diário);
- Ao longo do caminho, o ScrumMaster mantém a equipe focada em seu objetivo.
- No final do sprint, o trabalho deve ser potencialmente possível de o entregar para o cliente;
- O sprint termina com uma revisão de sprint e retrospectiva;
- Como o próximo sprint começa, a equipe escolhe outro pedaço do product backlog e começa a trabalhar novamente.

A figura 4(quatro) representa o ciclo dessa metodologia.

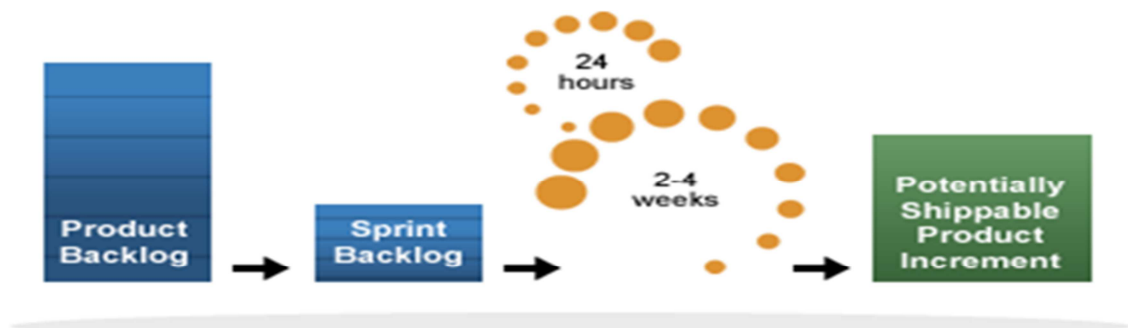


Figura 4 - Representação Scrum.

SCRUMBRASIL, **SCRUM**. Disponível em: <<http://scrumbrasil.blogspot.com.br/2011/09/scrum-em-30-segundos.html>> Acesso em 15 dez. 2011.

O ciclo se repete até que os itens no productbacklog tenham sido concluídos, o orçamento esteja esgotado, ou um prazo chega. Qual destes marcos marca o fim do trabalho é inteiramente específico para o projeto. Não importa qual risco para

trabalhá-lo, Scrum garante que a obra mais valiosa seja concluída quando o projeto terminar.

5. Caso de uso

5.1. Contextualização da empresa

A LIGA NORTE-RIO-GRANDENSE CONTRA O CÂNCER – LNRCC, fundada e, 17 de julho de 1949, declarada de utilidade pública pela Lei Federal nº 86.871, de 25 de janeiro de 1982, Lei Estadual nº 157, de 05 de outubro de 1949 e pela Lei Municipal nº 3.254, 26 de outubro de 1981, registrada no Conselho Nacional de Assistência social – CNAS e filiada à Sociedade Brasileira de Cancerologia, é uma Sociedade Civil sem fins lucrativos de caráter científico-social-filantrópico, com personalidade jurídica de direito privado, tendo por objetivo a prevenção e o combate ao câncer no âmbito do Estado do Rio Grande do Norte. Para atendimento de seus objetivos a LNRCC possui em pleno funcionamento quatro unidades hospitalares: Hospital Luiz Antônio, Policlínica, Ambulatório CECAN e a unidade de Oncologia do Seridó. A LNRCC realizou o albergamento da Casa de Apoio ao Paciente com Câncer (Casa Irmã Gabriela), com objetivo fornecer melhores acomodações aos pacientes do interior em fase de tratamento.

A LNRCC vem expandindo suas atividades, sempre voltada para o social, ingressando na área de ensino, através do Departamento de Pesquisa Ensino e Ação Comunitária – DEPECOM, sendo ministradas disciplinas curriculares do Curso de Medicina da UFRN; cursos básicos de oncologias, ciclos de palestras e simpósios direcionados a oncologia, estágios, residência médica e trabalhos científicos de pesquisa.

A Liga Contra o Câncer é o único Centro de Alta Complexidade em Oncologia II (CACON II) do Rio Grande do Norte. Isso significa dispor de todas especialidades médicas e multidisciplinares, além de todos tratamentos e equipamentos necessários à atenção oncológica. Atualmente, são mais de 1.000 colaboradores alocados nas quatro unidades da instituição, empenhados em oferecer excelência em termos de atendimento.

6. Casos de uso, Cenário Cadastro Paciente

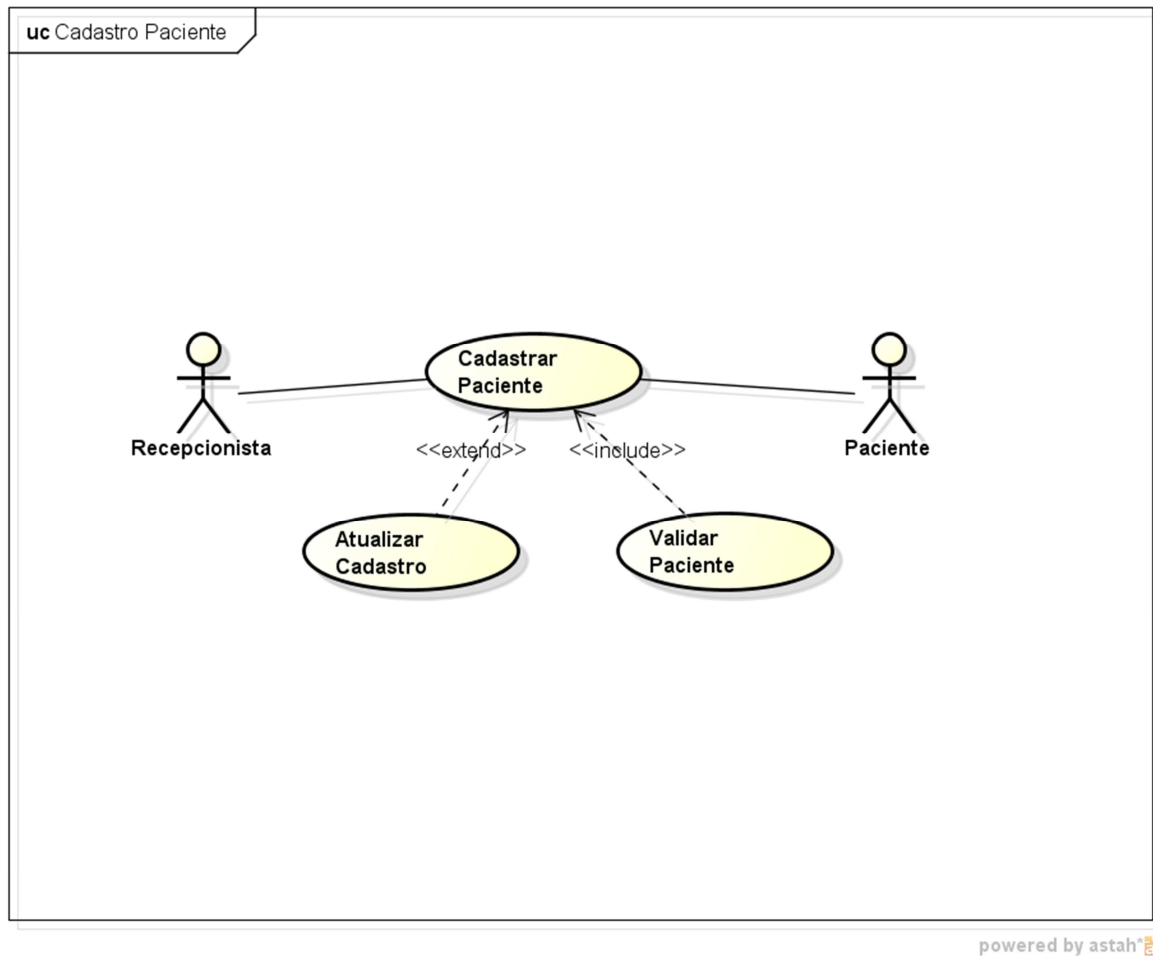


Figura 5 – Cenário de Cadastro de paciente.

Fonte: Resultado da pesquisa do autor.

6.1. Descrição do caso de uso Cadastrar Paciente

Caso de uso: Cadastrar Paciente.

- a. **Descrição:** Este caso de uso descreve o processo de cadastramento de um paciente no sistema.
- b. **Ator:** Recepcionista, Paciente.
- c. **Pré-condições:** O paciente deve apresentar Identidade, CPF, e deve está presente, não podendo ser feito o cadastro por telefone ou outros meios.

d. Pós-condições: Sempre que o paciente voltar a realizar outro atendimento, o cadastro deve ser conferido e atualizado.

e. Fluxo ótimo:

1. O Recepcionista entra no sistema pelo setor de atendimento com suas devidas permissões.
2. No menu de cadastros, o sistema disponibilizará uma tela específica.
3. O recepcionista insere os dados no sistema.
4. O sistema Verifica os dados.
A1: Dados Incorretos.
A2: Campos não preenchidos.
5. Sistema salva os dados.

f. Fluxo Alternativo:

A1: Dados Incorretos.

1. O sistema demonstra uma mensagem explicativa descrevendo o erro e o que fazer.
2. O recepcionista toma as medidas necessárias para sanar o problema.
3. O sistema retorna para o passo 2(dois) do fluxo ótimo para retificar o impedimento apresentado.

A2: Campos não preenchidos.

1. O sistema demonstra uma mensagem explicativa descrevendo qual campo se faz necessário preencher.
2. O recepcionista verifica a informação.
3. O sistema retorna para o passo 2(dois) do fluxo ótimo para inserir o campo obrigatório.

7. Casos de uso, Cenário Agendamento Paciente

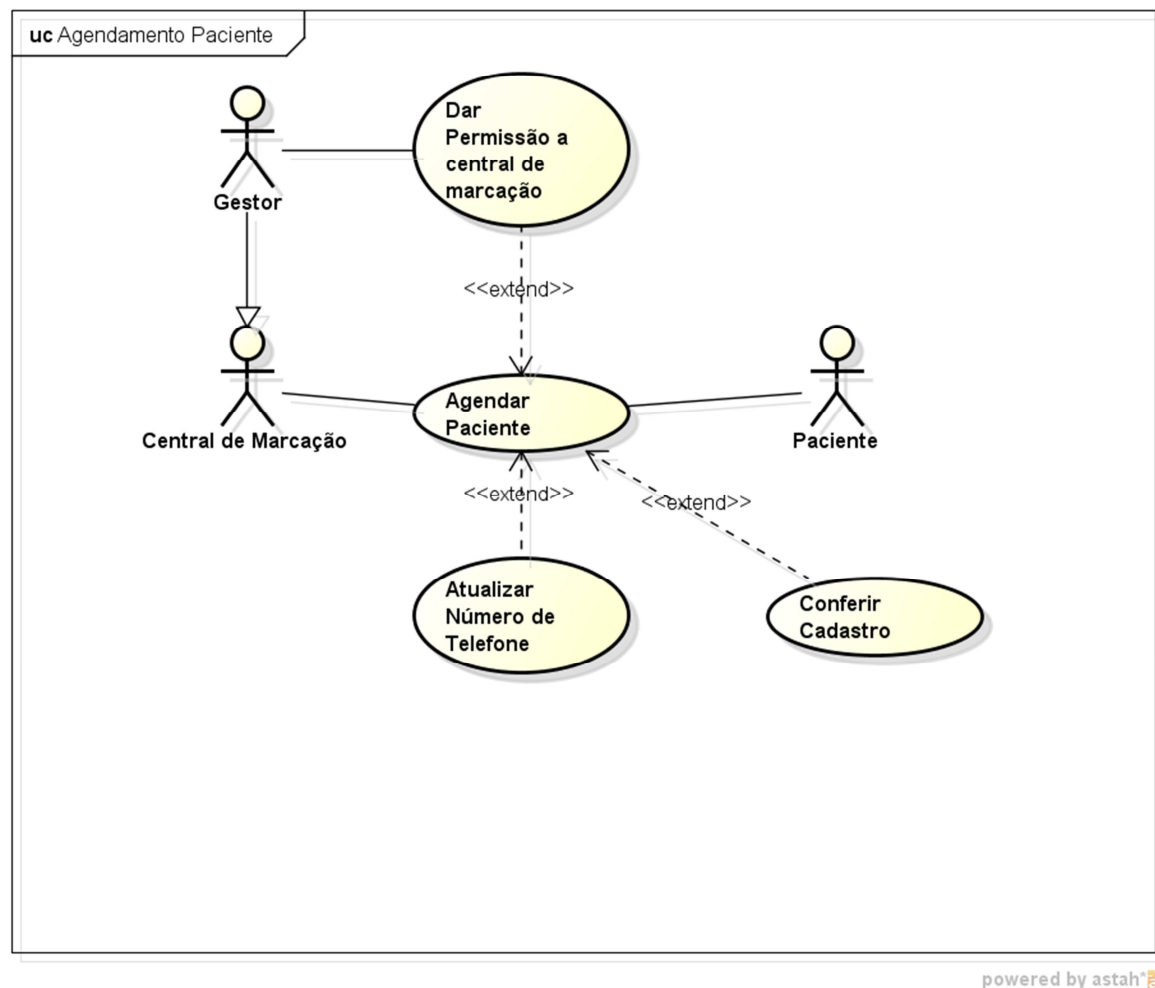


Figura 6 - Cenário de Agendamento de Paciente.

Fonte: Resultado da pesquisa do autor.

7.1. Descrição do caso de uso Agendar Paciente

Caso de uso: Agendar Paciente.

- a. **Descrição:** Este caso de uso descreve o processo de agendamento de consultas, procedimentos, cirurgias do hospital.
- b. **Ator:** Central de Marcação, Gestor, Paciente.
- c. **Pré-condições:** O paciente precisa informar no mínimo o seu nome e telefone, caso o mesmo não tenha cadastro no sistema.

d. Pós-condições: Sempre que o paciente voltar a realizar um novo agendamento, a central de marcação precisa conferir se o mesmo já tem cadastro no sistema e atualizar o número de telefone.

e. Fluxo ótimo:

1. O profissional que atua na central de marcação entra no sistema pela unidade que ele deseja realizar o agendamento com suas devidas permissões.
2. No menu de monitoração, o sistema disponibilizará uma tela específica de marcação de agenda.
3. O recepcionista insere os dados no sistema.
4. O sistema Verifica os dados.
 - A1:** Dados Incorretos.
 - A2:** Campos não preenchidos.
 - A3:** Falta de acesso às entidades de agendamento.
5. Sistema salva os dados.

f. Fluxo Alternativo:

A1: Dados Incorretos.

1. O sistema demonstra uma mensagem explicativa descrevendo o erro e o que fazer.
2. O profissional da central de marcação toma as medidas necessárias para sanar o problema.
3. O sistema retorna para o passo 2(dois) do fluxo ótimo para retificar o impedimento apresentado.

A2: Campos não preenchidos.

1. O sistema demonstra uma mensagem explicativa descrevendo qual campo se faz necessário preencher.
2. O profissional da central de marcação verifica a informação.
3. O sistema retorna para o passo 2(dois) do fluxo ótimo para inserir o campo obrigatório.

A3: Falta de acesso às unidades de agendamento.

1. O sistema demonstra uma mensagem explicativa informando à falta de acesso.

2. O profissional da central de marcação informa ao gestor a falta de acesso.
3. O gestor analisa as informações dando ou não permissão ao seu funcionário.
4. O sistema retorna para o passo 2(dois) do fluxo ótimo para inserir o campo obrigatório.

8. Casos de uso, Cenário Registro Atendimento

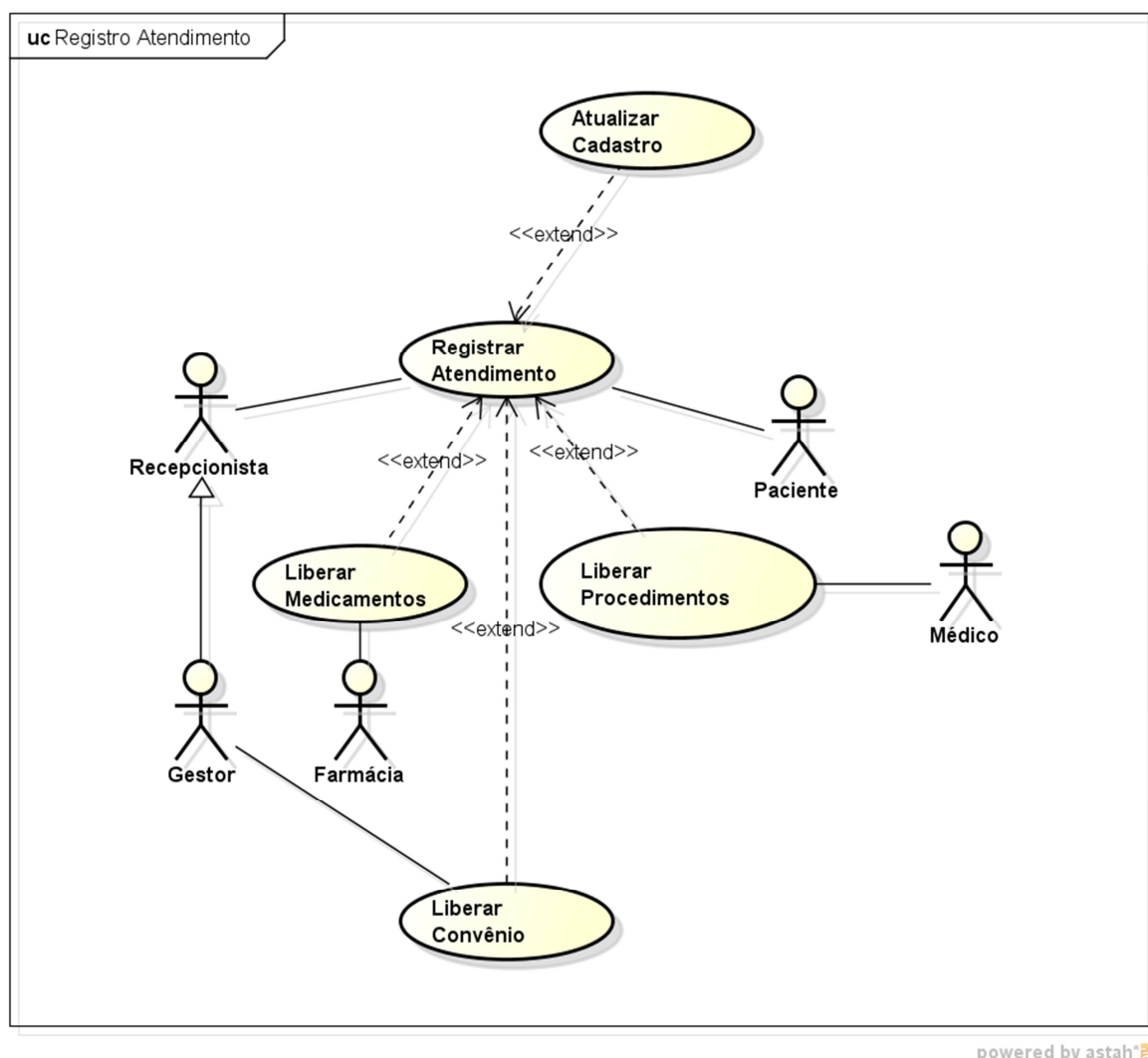


Figura 7 – Cenário Registro Atendimento.

Fonte: Resultado da pesquisa do autor.

8.1. Descrição do caso de uso Registrar Atendimento

Caso de uso: Registrar Atendimento.

- a. Descrição:** Este caso de uso descreve o processo de registro de atendimento com a chegada do paciente ao hospital para realização de consultas, cirurgias, procedimentos, entrega de medicamentos e internações no hospital.
- b. Ator:** Recepcionista, gestor, médico, paciente e farmácia.
- c. Pré-condições:** O Paciente precisa apresentar sua carteira do plano ou cartão do sus, caso o atendimento não seja particular.
- d. Pós-condições:** É necessário verificar se o atendimento foi concluído e realizar a alta ou encaminhamento do paciente para outro serviço.
- e. Fluxo ótimo:**
 - 1. O Recepcionista entra no sistema pelo setor de atendimento com suas devidas permissões.
 - 2. No menu de monitoração, o sistema disponibilizará uma tela específica de registro de atendimento.
 - 3. O recepcionista insere os dados no sistema.
 - 4. O sistema Verifica os dados.
 - A1:** Dados Incorretos.
 - A2:** Campos não preenchidos.
 - A3:** Convênio não autorizado.
 - A4:** Número de procedimentos ultrapassados no mês.
 - A5:** Número de entrega de medicamentos ultrapassados no mês.
 - 5. Sistema salva os dados.
- f. Fluxo Alternativo:**
 - A1:** Dados Incorretos.
 - 1. O sistema demonstra uma mensagem explicativa descrevendo o erro e o que fazer.
 - 2. O recepcionista toma as medidas necessárias para sanar o problema.

3. O sistema retorna para o passo 2(dois) do fluxo ótimo para retificar o impedimento apresentado.

A2: Campos não preenchidos.

1. O sistema demonstra uma mensagem explicativa descrevendo qual campo se faz necessário preencher.
2. O recepcionista verifica a informação.
3. O sistema retorna para o passo 2(dois) do fluxo ótimo para inserir o campo obrigatório.

A3: Convênio não autorizado.

1. O sistema demonstra uma mensagem informando que o convênio está suspenso.
2. O recepcionista informa ao gestor o problema.
3. O gestor analisa as informações dando ou não permissão ao funcionário prosseguir com o atendimento.
4. O sistema retorna para o passo 2(dois) do fluxo ótimo caso o gestor tenha retirado o impedimento ou o registro do atendimento é cancelado.

A4: Número de procedimentos ultrapassados no mês.

1. O sistema demonstra uma mensagem informando que o número de entrega de procedimentos realizados no mês desse paciente foi ultrapassado.
2. O recepcionista informa o problema ao gestor.
3. O gestor analisa as informações, e entra em contato com o médico.
4. O médico analisa as informações e dá ou não a liberação.
5. O sistema retorna para o passo 2(dois) do fluxo ótimo caso o médico tenha autorizado ao gestor retirar o impedimento, ou o atendimento é cancelado.

A5: Número de entrega de medicamentos ultrapassados no mês.

1. O sistema demonstra uma mensagem informando que o número de entrega de medicamentos no mês desse paciente foi ultrapassado.

2. O recepcionista informa o problema ao gestor.
3. O gestor analisa as informações, e entra em contato com a farmácia.
4. A farmácia analisa as informações e dá ou não a liberação.
5. O sistema retorna para o passo 2(dois) do fluxo ótimo caso a farmácia tenha autorizado ao gestor retirar o impedimento, ou o atendimento é cancelado.

9. Casos de uso, Cenário Registrar Serviço de Apoio

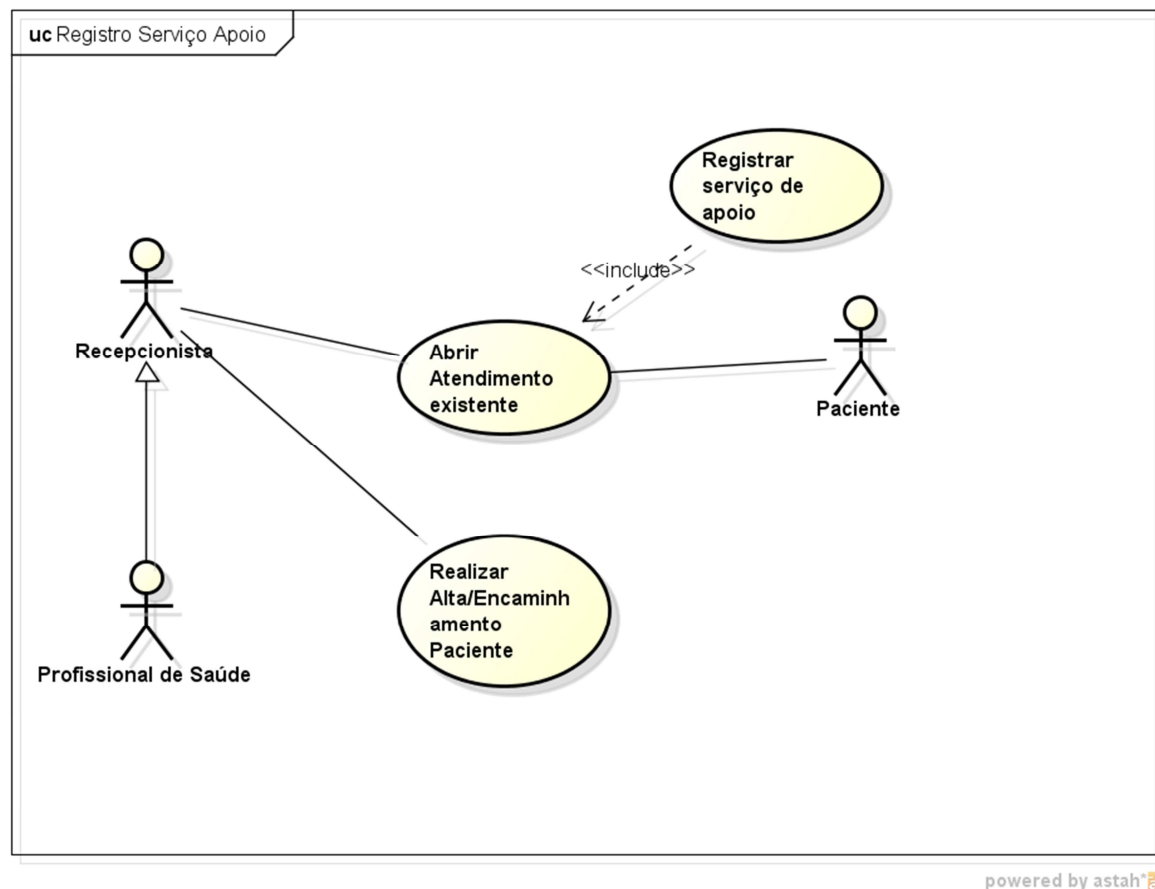


Figura 8 – Cenário Registro Serviço de Apoio.

Fonte: Resultado da pesquisa do autor.

9.1. Descrição Caso de uso Registrar Serviço de Apoio

Caso de uso: Registrar Serviço de Apoio.

- a. **Descrição:** Este caso de uso descreve o processo de registro de serviço de apoio a um atendimento existente, ou seja, um atendimento vinculado à outro.
- b. **Ator:** Profissional de saúde, recepcionista, Paciente;
- c. **Pré-condições:** O atendimento não pode ter tido alta, e o serviço tem que ter permissão de realização;

d. Pós-condições: O profissional de saúde ou recepcionista deve realizar a alta/encaminhamento do paciente;

e. Fluxo ótimo:

1. O Recepcionista ou profissional de saúde entra no sistema pelo setor de atendimento com suas devidas permissões;
2. Na tela de monitoração o sistema disponibilizará um botão para realizar esta ação que dá acesso a uma tela específica.
3. O recepcionista ou profissional de saúde insere os dados no sistema;
4. O sistema Verifica os dados;
A1: Dados Incorretos.
A2: Campos não preenchidos.
5. Sistema salva os dados;

f. Fluxo Alternativo:

A1: Dados Incorretos.

1. O sistema demonstra uma mensagem explicativa descrevendo o erro e o que fazer.
2. O recepcionista ou profissional de saúde toma as medidas necessárias para sanar o problema.
3. O sistema retorna para o passo 2(dois) do fluxo ótimo para retificar o impedimento apresentado.

A2: Campos não preenchidos.

1. O sistema demonstra uma mensagem explicativa descrevendo qual campo se faz necessário preencher.
2. O recepcionista ou profissional de saúde verifica a informação.
3. O sistema retorna para o passo 2(dois) do fluxo ótimo para inserir o campo obrigatório.

10. Casos de uso, Cenário Atender Paciente

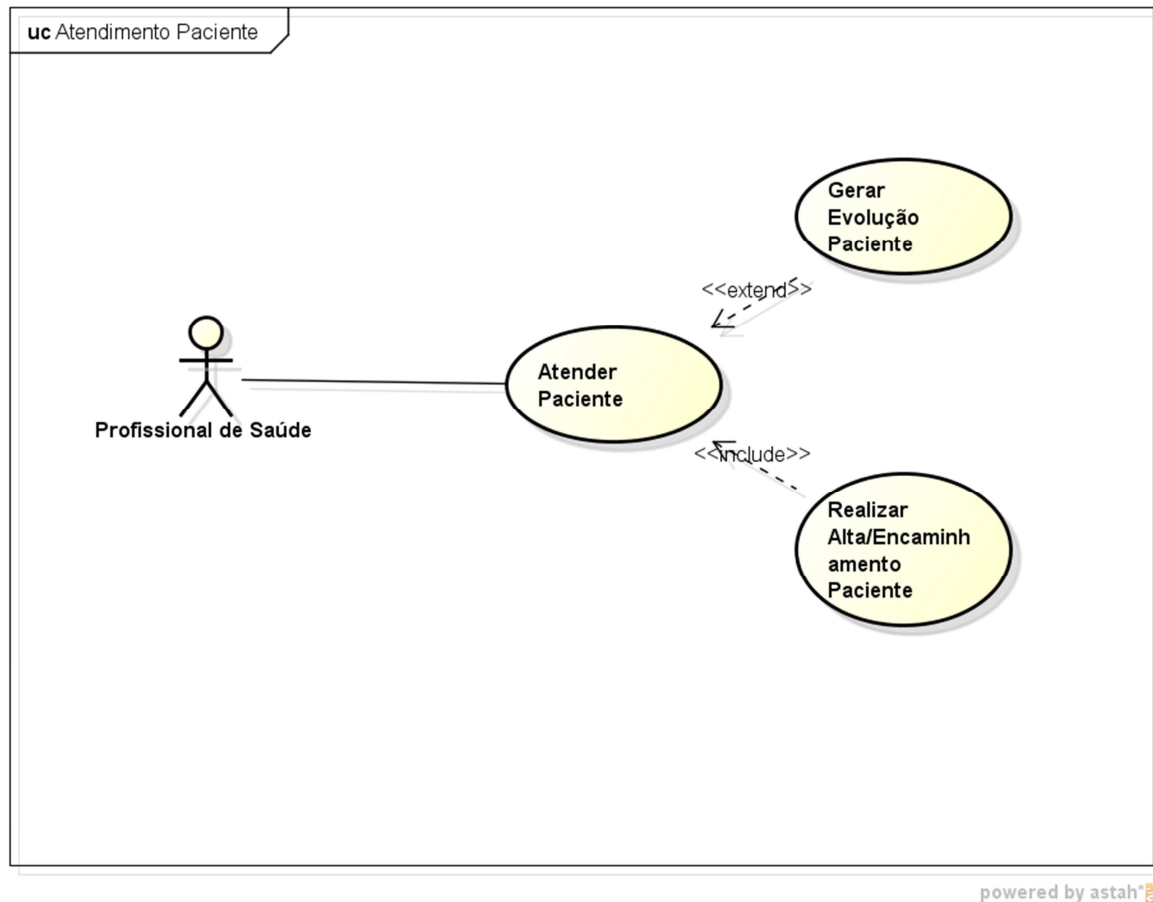


Figura 9 – Cenário Atendimento Paciente.

Fonte: Resultado da pesquisa do autor.

10.1. Descrição caso de uso Atender Paciente

Caso de uso: Atender Paciente.

- a. **Descrição:** Este caso de uso descreve o processo de atendimento ao paciente de um profissional de saúde.
- b. **Ator:** Profissional de saúde
- c. **Pré-condições:** O atendimento tem que ter sido registrado na recepção.
- d. **Pós-condições:** As observações inseridas no sistema de anamnese e evolução médica não podem ser alteradas ou excluídas.

e. Fluxo ótimo:

1. O profissional de saúde entra no sistema com suas devidas permissões de acesso.
2. O sistema disponibilizará uma tela específica.
3. O profissional de saúde insere os dados no sistema.
4. O sistema verifica os dados.

A1: Dados Incorretos.

A2: Campos não preenchidos.

5. Sistema salva os dados.

f. Fluxo Alternativo:

A1: Dados Incorretos.

1. O sistema demonstra uma mensagem explicativa descrevendo o erro e o que fazer.
2. O profissional de saúde toma as medidas necessárias para sanar o problema.
3. O sistema retorna para o passo 2(dois) do fluxo ótimo para retificar o impedimento apresentado.

A2: Campos não preenchidos.

1. O sistema demonstra uma mensagem explicativa descrevendo qual campo se faz necessário preencher.
2. O profissional de saúde verifica a informação.
3. O sistema retorna para o passo 2(dois) do fluxo ótimo para inserir o campo obrigatório.

11. Casos de uso, Cenário Monitorar Atendimento

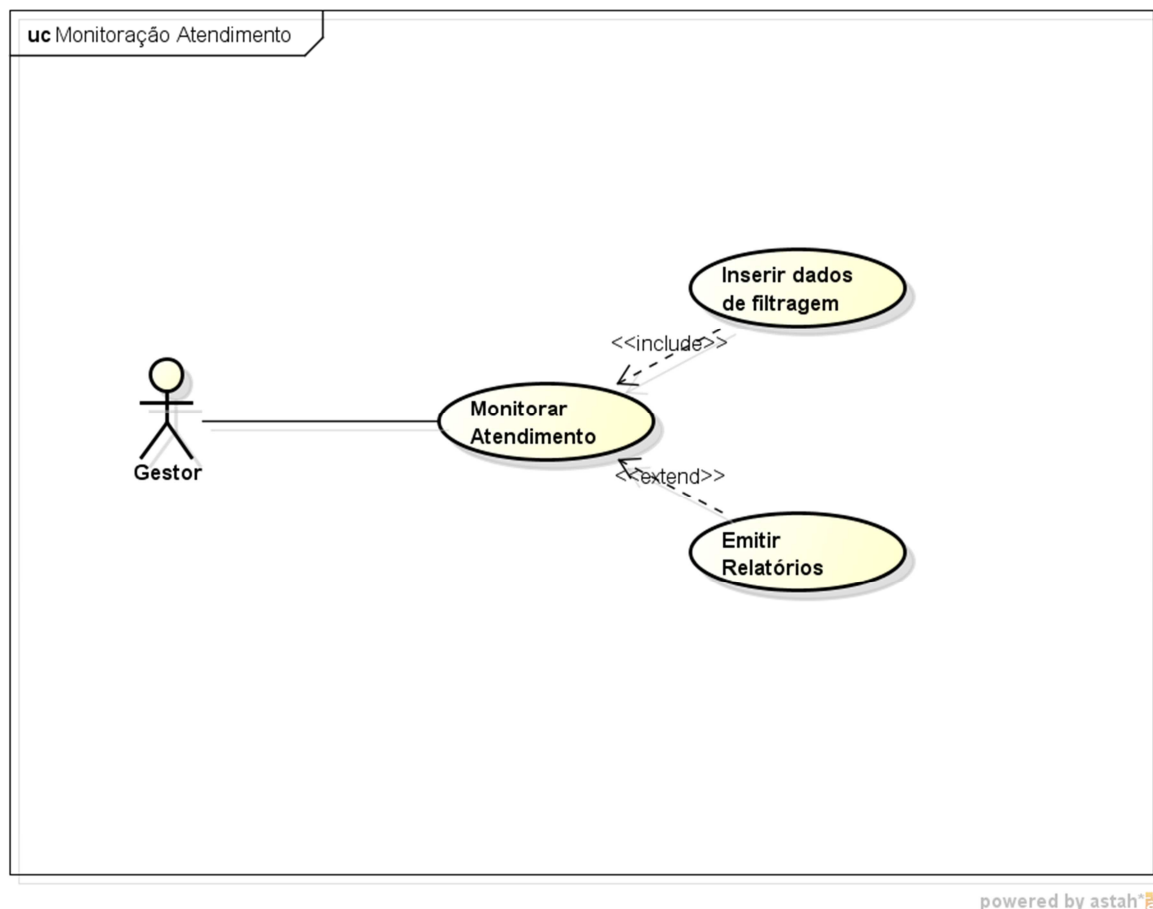


Figura 10 – Cenário de Monitoramento de Paciente.

Fonte: Resultado da pesquisa do autor.

11.1. Descrição caso de uso Atender Paciente

Caso de uso: Monitorar Atendimento.

- a. **Descrição:** Este caso de uso descreve o processo de monitoração dos atendimentos para gerenciamento do fluxo de entrada e saída de pacientes.
- b. **Ator:** Gestor.
- c. **Pré-condições:** Os filtros devem ser ajustados.
- d. **Pós-condições:** Emissões de relatórios.
- e. **Fluxo ótimo:**

1. O gestor entra no sistema com suas devidas permissões de acesso.
2. O sistema disponibilizará uma tela específica.
3. O gestor ajusta todos os filtros necessários para extração da informação.
4. O sistema verifica os dados.
A1: Dados Incorretos.
A2: Filtros não preenchidos.
5. Sistema emite consulta.

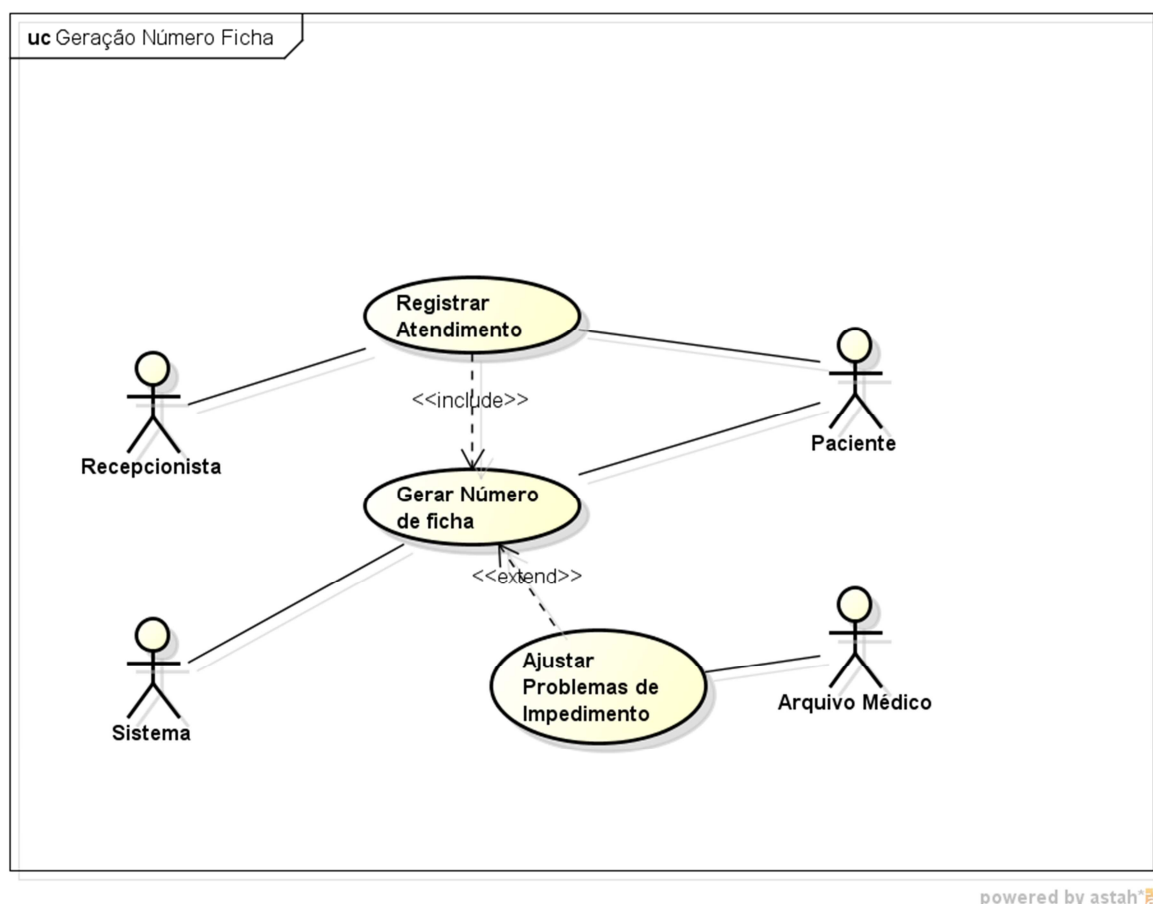
f. Fluxo Alternativo:

A1: Dados Incorretos.

1. O sistema demonstra uma mensagem explicativa descrevendo o erro e o que fazer.
2. O gestor toma as medidas necessárias para sanar o problema;
3. O sistema retorna para o passo 2(dois) do fluxo ótimo para retificar o impedimento apresentado.

A2: Filtros não preenchidos.

1. O sistema demonstra uma mensagem explicativa descrevendo qual filtro se faz necessário preencher.
2. O gestor verifica a informação.
3. O sistema retorna para o passo 2(dois) do fluxo ótimo para inserir o os dados necessários para o filtro.

12. Casos de uso, Cenário Gerar Número de Ficha**Figura 11 – Cenário Geração Número Ficha.****Fonte: Resultado da pesquisa do autor.**

12.1. Descrição caso de uso Gerar Número de Ficha

Caso de uso: Gerar Número Ficha.

- a. **Descrição:** Este caso de uso descreve a geração de um número de ficha para o paciente para arquivamento físico de todas as entradas posteriores do paciente.
- b. **Ator:** Sistema, recepcionista, arquivo médico.
- c. **Pré-condições:** O paciente não ter número de ficha.
- d. **Pós-condições:** Criar um prontuário físico para o paciente.
- e. **Fluxo ótimo:**
 - 1. O recepcionista entra no sistema com suas devidas permissões de acesso.
 - 2. O sistema disponibilizará uma tela específica.
 - 3. O recepcionista insere os dados no sistema.
 - 4. O sistema gera um número automático de ficha para o paciente.
 - 5. O sistema verifica os dados.
 - A1:** Dados Incorretos.
 - A2:** Campos não preenchidos.
 - A3:** Número de ficha não pode ser gerado para o paciente.
 - 6. Sistema salva os dados.
- f. **Fluxo Alternativo:**
 - A1:** Dados Incorretos.
 - 1. O sistema demonstra uma mensagem explicativa descrevendo o erro e o que fazer.
 - 2. O recepcionista toma as medidas necessárias para sanar o problema.
 - 3. O sistema retorna para o passo 2(dois) do fluxo ótimo para retificar o impedimento apresentado.
 - A2:** Campos não preenchidos.
 - 1. O sistema demonstra uma mensagem explicativa descrevendo qual campo se faz necessário preencher.
 - 2. O recepcionista verifica a informação.

3. O sistema retorna para o passo 2(dois) do fluxo ótimo para inserir o campo obrigatório.

A3: Número de ficha não pode ser gerado para o paciente.

1. O sistema demonstra uma mensagem explicativa descrevendo o problema apresentado.
2. O recepcionista apresenta o problema para o arquivo médico.
3. O arquivo médico verifica os dados e sana o problema.
4. O sistema retorna para o passo 2(dois) do fluxo ótimo para geração do número de ficha do paciente.

13. Casos de uso, Cenário Protocolo Arquivo Paciente

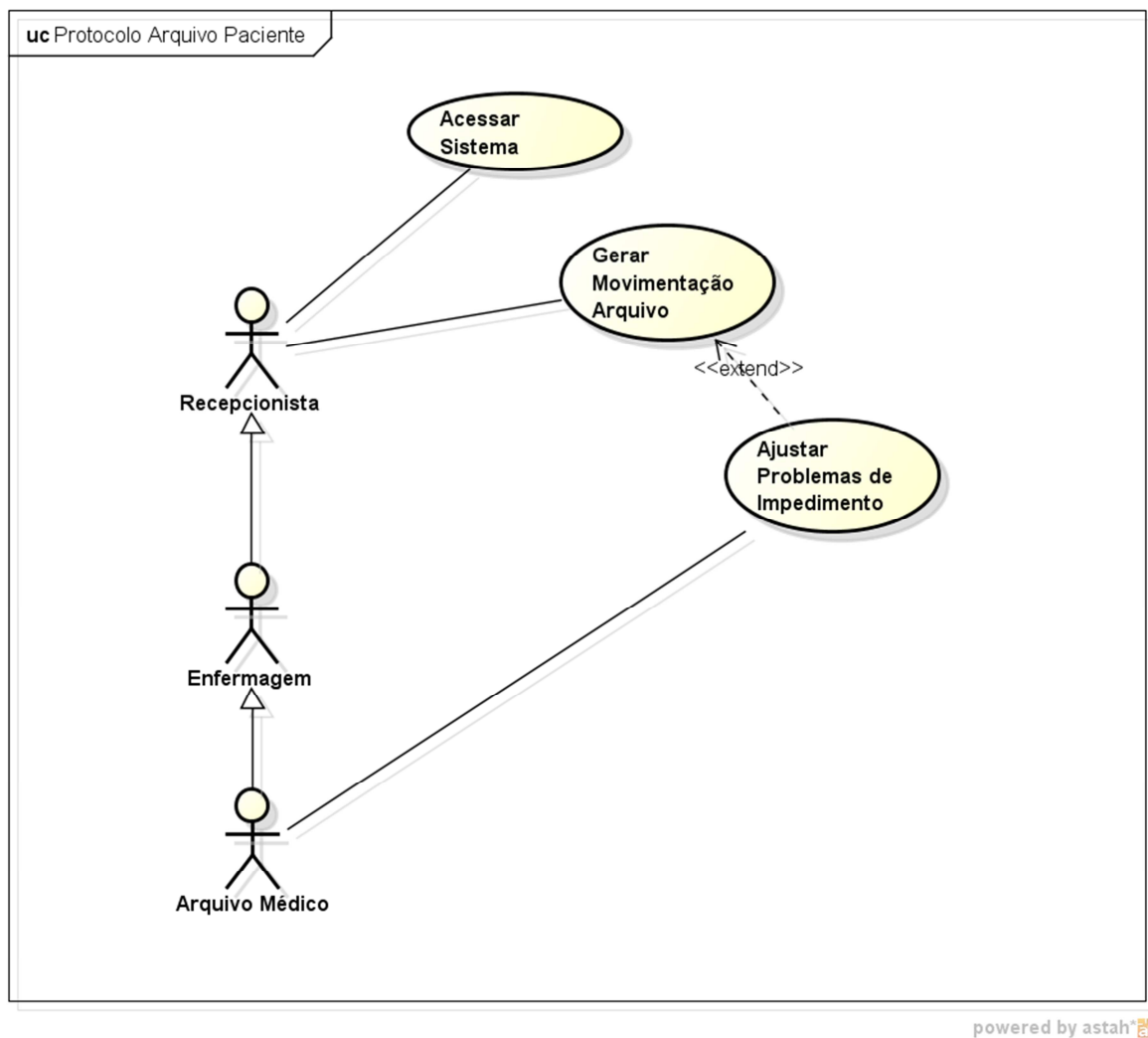


Figura 12 – Cenário Protocolo Arquivo Paciente.

Fonte: Resultado da pesquisa do autor.

13.1. Descrição caso de uso Protocolar Arquivo Paciente.

Caso de uso: Protocolar Arquivo Paciente.

- a. **Descrição:** Este caso de uso descreve o processo de logística do prontuário físico do paciente.
- b. **Ator:** Recepcionista, enfermagem, arquivo médico.
- c. **Pré-condições:** O arquivo tem que está fisicamente no setor.
- d. **Pós-condições:** O arquivo físico deve ser enviado para o setor desejado.
- e. **Fluxo ótimo:**
 - 1. O recepcionista ou enfermagem entra no sistema com suas devidas permissões de acesso.
 - 2. O sistema disponibilizará uma tela específica.
 - 3. O recepcionista ou enfermagem solicita ao setor onde se encontra o arquivo físico.
 - 4. O sistema verifica os dados.
 - A1:** Dados Incorretos.
 - A2:** Campos não preenchidos.
 - A3:** Filtros não preenchidos.
 - A4:** Arquivo não pode ser movimentado.
 - 5. Sistema salva os dados.
- f. **Fluxo Alternativo:**
 - A1:** Dados Incorretos.
 - 1. O sistema demonstra uma mensagem explicativa descrevendo o erro e o que fazer.
 - 2. O recepcionista ou enfermagem toma as medidas necessárias para sanar o problema.
 - 3. O sistema retorna para o passo 2 do fluxo ótimo para retificar o impedimento apresentado.
 - A2:** Campos não preenchidos.
 - 1. O sistema demonstra uma mensagem explicativa descrevendo qual campo se faz necessário preencher.

2. O recepcionista ou enfermagem verifica a informação.
3. O sistema retorna para o passo 2 do fluxo ótimo para inserir o campo obrigatório.

A3: Filtros não preenchidos.

1. O sistema demonstra uma mensagem explicativa descrevendo qual filtro se faz necessário preencher.
2. O recepcionista ou enfermagem verifica a informação.
3. O sistema retorna para o passo 2(dois) do fluxo ótimo para inserir os dados necessários para o filtro.

A4: Arquivo não pode ser movimentado.

1. O sistema demonstra uma mensagem explicativa descrevendo o problema referente ao caso.
2. O recepcionista ou enfermagem informa o problema apresentado ao arquivo médico.
3. O arquivo médico verifica os dados e retira o impedimento.
4. O sistema retorna para o passo 2(dois) do fluxo ótimo para realizar o envio ou recebimento do arquivo.

14. Casos de uso, Alterar Atendimento

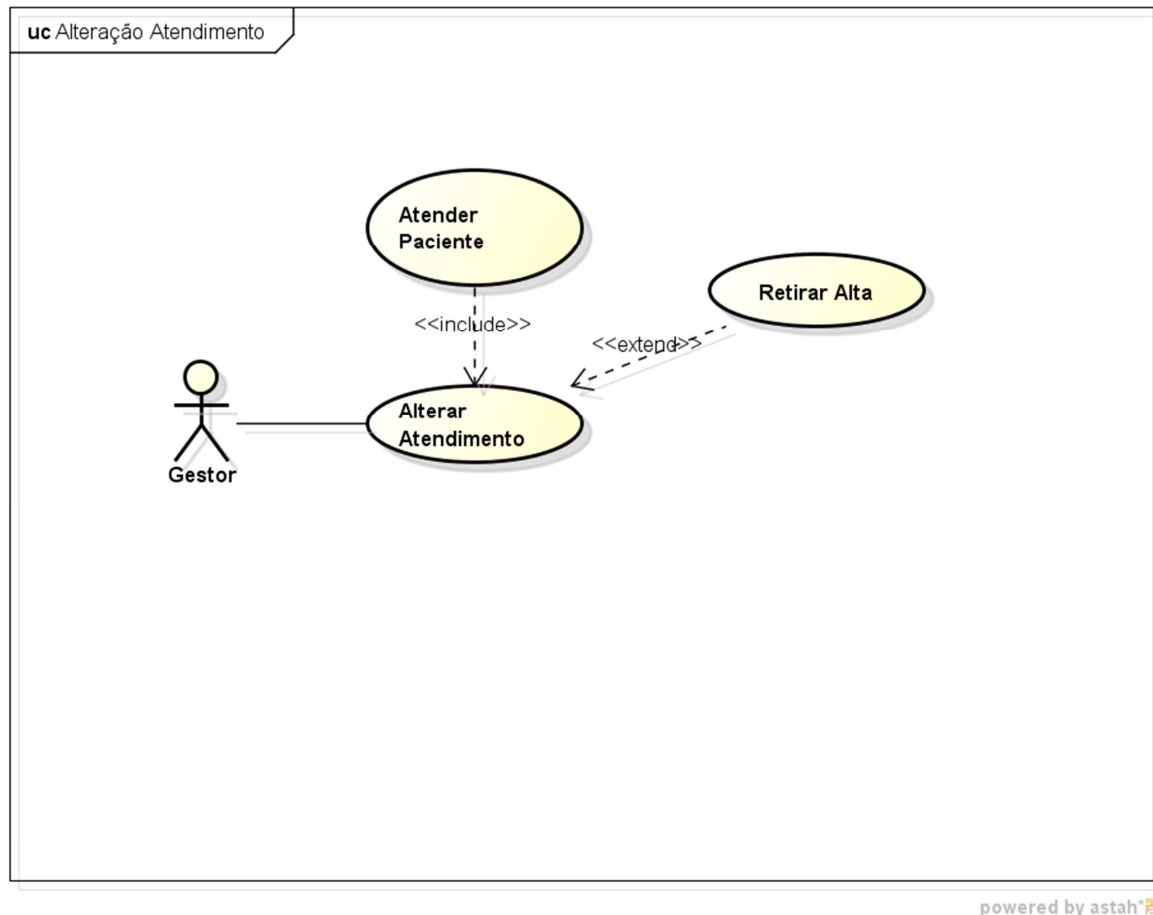


Figura 13 – Cenário Alteração Atendimento.

Fonte: Resultado da pesquisa do autor.

14.1. Descrição caso de uso Alterar Atendimento

Caso de uso: Alterar Atendimento.

- a. **Descrição:** Este caso de uso descreve o processo de alteração de atendimento do paciente.
- b. **Ator:** Gestor.
- c. **Pré-condições:** O atendimento tem que está registrado.
- d. **Pós-condições:** O gestor deve informar a recepção à retificação ocorrida.
- e. **Fluxo ótimo:**

1. O gestor entra no sistema com suas devidas permissões de acesso.
2. O sistema disponibilizará uma tela específica.
3. O gestor altera os dados.
4. O sistema verifica os dados.
A1: Dados Incorretos.
A2: Campos não preenchidos.
A3: Atendimento com alta.
5. Sistema salva os dados.

f. Fluxo Alternativo:

A1: Dados Incorretos.

1. O sistema demonstra uma mensagem explicativa descrevendo o erro e o que fazer.
2. O gestor toma as medidas necessárias para sanar o problema;
3. O sistema retorna para o passo 2(dois) do fluxo ótimo para inserir os dados corretamente.

A2: Campos não preenchidos.

1. O sistema demonstra uma mensagem explicativa descrevendo qual campo se faz necessário preencher.
2. O gestor verifica a informação.
3. O sistema retorna para o passo 2(dois) do fluxo ótimo para inserir o campo obrigatório.

A3: Atendimento com Alta

1. O sistema demonstra uma mensagem explicativa descrevendo que o atendimento está com alta.
2. O gestor verifica a informação.
3. O gestor retira à alta.
4. O sistema retorna para o passo 2(dois) do fluxo ótimo finalização da alteração desejada.

15. Casos de uso, Gerar Alta/Encaminhamento Paciente

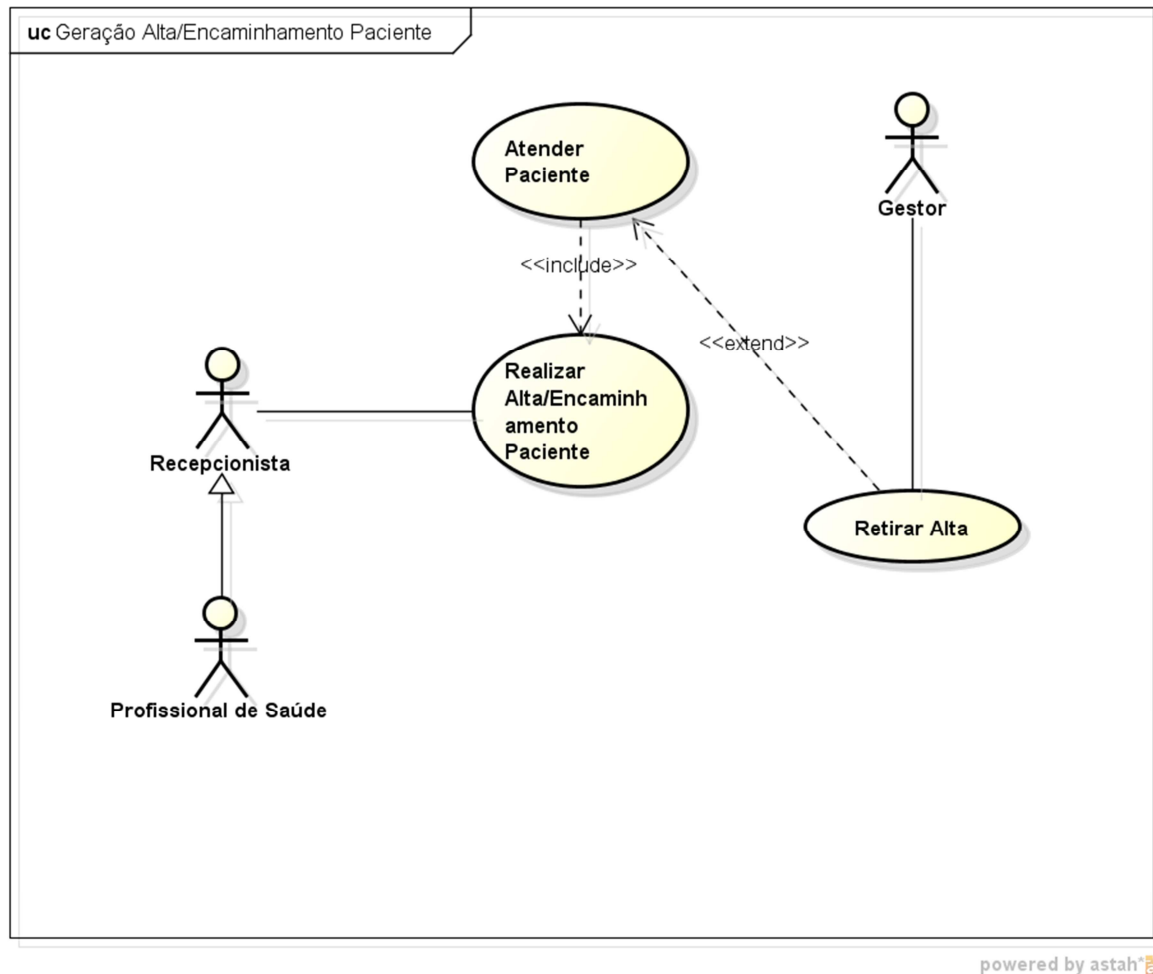


Figura 14 – Cenário Geração Alta/Encaminhamento Paciente.

Fonte: Resultado da Pesquisa do autor.

15.1. Descrição caso de uso Gerar Alta/Encaminhamento Paciente

Caso de uso: Gerar Alta/Encaminhamento Paciente.

- a. **Descrição:** Este caso de uso descreve o processo de alta hospitalar do paciente ou encaminhamento para outro serviço médico.
- b. **Ator:** Recepcionista, Profissional de saúde, gestor.
- c. **Pré-condições:** O atendimento tem que está em aberto.
- d. **Pós-condições:** Depois que o atendimento estiver com a alta/encaminhamento realizada, somente o gestor pode retira-la.

e. Fluxo ótimo:

1. O recepcionista ou profissional de saúde entra no sistema com suas devidas permissões de acesso.
2. O sistema disponibilizará uma tela específica.
3. O recepcionista ou profissional de saúde insere os dados.
4. O sistema verifica os dados.

A1: Dados Incorretos.

A2: Campos não preenchidos.

A3: Alta realizada incorretamente.

5. Sistema salva os dados.

f. Fluxo Alternativo:

A1: Dados Incorretos.

1. O sistema demonstra uma mensagem explicativa descrevendo o erro e o que fazer.
2. O recepcionista ou profissional de saúde toma as medidas necessárias para sanar o problema.
3. O sistema retorna para o passo 2(dois) do fluxo ótimo para inserir os dados corretamente.

A2: Campos não preenchidos.

1. O sistema demonstra uma mensagem explicativa descrevendo qual campo se faz necessário preencher.
2. O recepcionista ou profissional de saúde verifica a informação.
3. O sistema retorna para o passo 2(dois) do fluxo ótimo para inserir o campo obrigatório.

A3: Alta realizada incorretamente.

1. O recepcionista ou profissional de saúde informa o problema ao gestor.
2. O gestor verifica a informação.
3. O gestor retira a alta/encaminhamento do paciente.
4. O sistema retorna para o passo 2(dois) do fluxo ótimo para o recepcionista ou profissional de saúde inserir os dados corretamente.

16. Diagrama de Classes

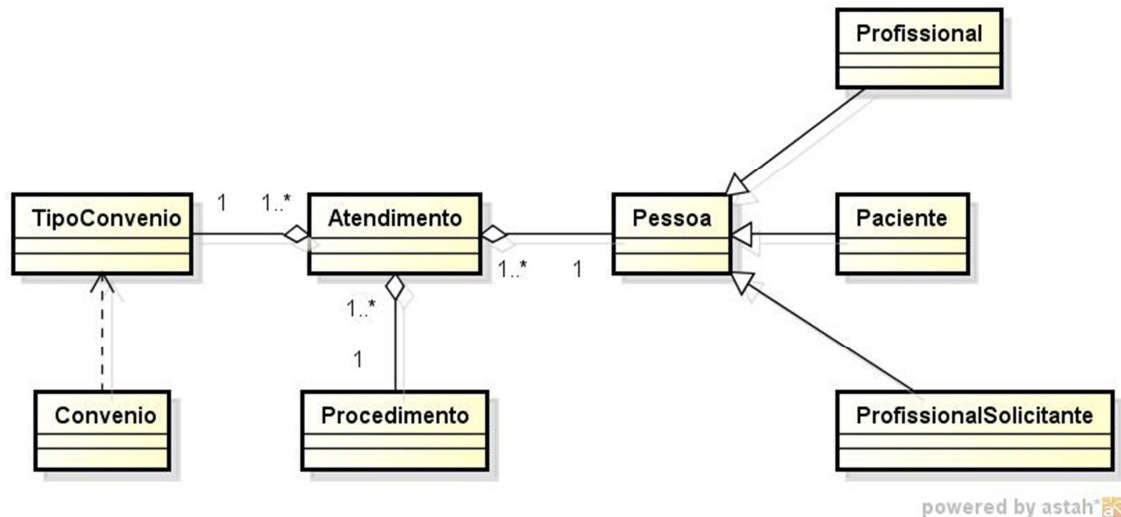


Figura 15 - Diagrama de Classes.

Fonte: Resultado da pesquisa do autor.

17. Processo de Atendimento

Para ocorrer o atendimento, alguns campos são necessários, como a especialidade, facilitando o filtro dos procedimentos e profissionais, o convênio, tipo convênio que especifica à categoria do plano de saúde do paciente, o profissional que seria o médico que executaria o serviço, o profissional solicitante que seja cadastrado no sistema o tipo da guia que tipifica o tipo de atendimento, se é um procedimento, uma consulta ou cirurgia. A tela de atender é a que fica responsável por todo processo explicado.

A tela de monitoração serve tanto para o médico ter base de quantos pacientes foram atendidos ou quantos já chegaram à recepção e estão esperando, quanto para a recepcionista realizar ações como à alta do paciente, impressão de ficha de referência entres outras funcionalidades.

18. Situação Atual

A equipe de desenvolvimento é composta por dois profissionais graduados em Sistemas de Informação. As funções executadas pelos profissionais são: análise de requisitos, análise de riscos, programação de aplicações, banco de dados, treinamento de novas funcionalidades e testes. Há ainda uma equipe de suporte composta por sete funcionários, sendo um gerente de T.I, um analista de infraestrutura, um gerente de redes, e quatro técnicos em suporte, estes distribuídos em dois na unidade Cekan, um no Hospital Luiz Antônio e um na unidade Policlínica. A equipe de suporte auxilia a equipe de desenvolvimento a implantar os módulos nas estações de trabalho dos setores.

18.1. Sistema Atual

O módulo atual onde é feito o atendimento do paciente dispõe de tais funcionalidades:

- Cadastros
 - o Paciente
 - o Gerar Número de Ficha
 - o Profissional Solicitante
 - o Clínica Externa
- Recepção
 - o Monitoração
 - o Alterar Atendimento
 - o Agenda
 - o Atender
 - o Serviço de Apoio
 - o Alta Por Lote
 - o Monitora Protocolo de Atendimento
 - o Recebimento de protocolo de atendimento
 - o Protocolo de arquivo
 - Monitoração

- Envio
 - Recebimento
- Relatórios
 - o Abertura de Prontuário
 - o Abertura de Prontuário por Médico
 - o Arquivos Abertos por Período
 - o Censo de transferências
 - o Comparativo entre meses
 - o Disponibilidade de leitos
 - o Lista de Atendimentos – Censo
 - o Lista de atendimentos – Caixa
 - o Lista de Altas
 - o Pacientes sem agendamento
 - o Produtividade Recepção
 - o Protocolo Atendimento
 - o Produção Médica
 - o Pendência cadastral
 - Paciente
- Informações do Sistema

18.2. Telas do Sistema Atual

Abaixo, uma demonstração de telas importantes para o processo de atendimento ao paciente do sistema atual utilizado na liga contra o câncer.

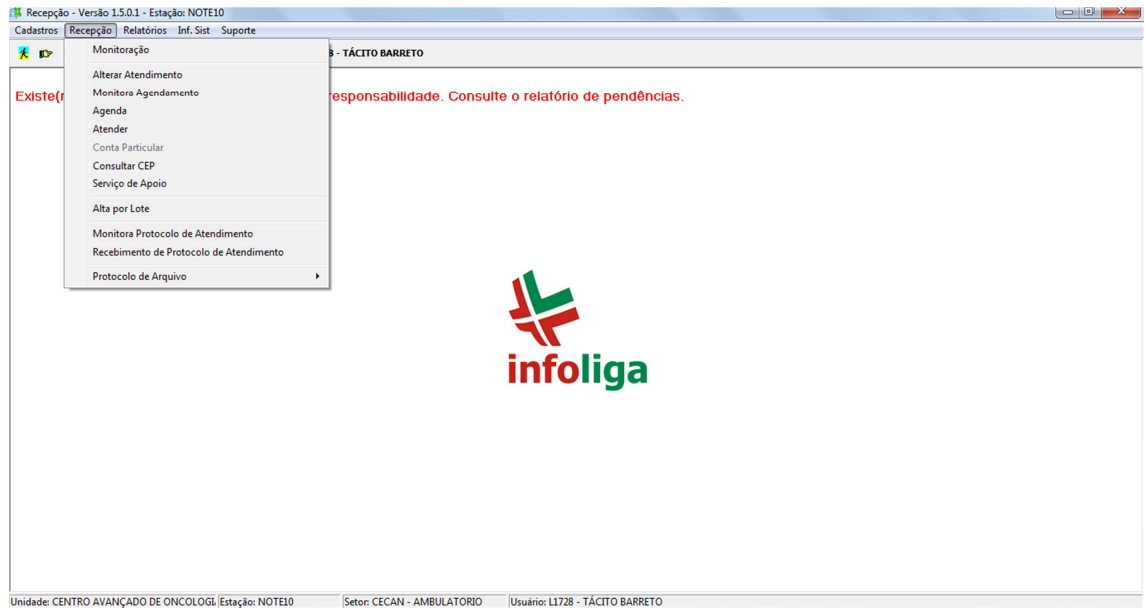


Figura 16 - Menu Responsável pelo processo de atendimento.

Fonte: Resultado da pesquisa do autor.

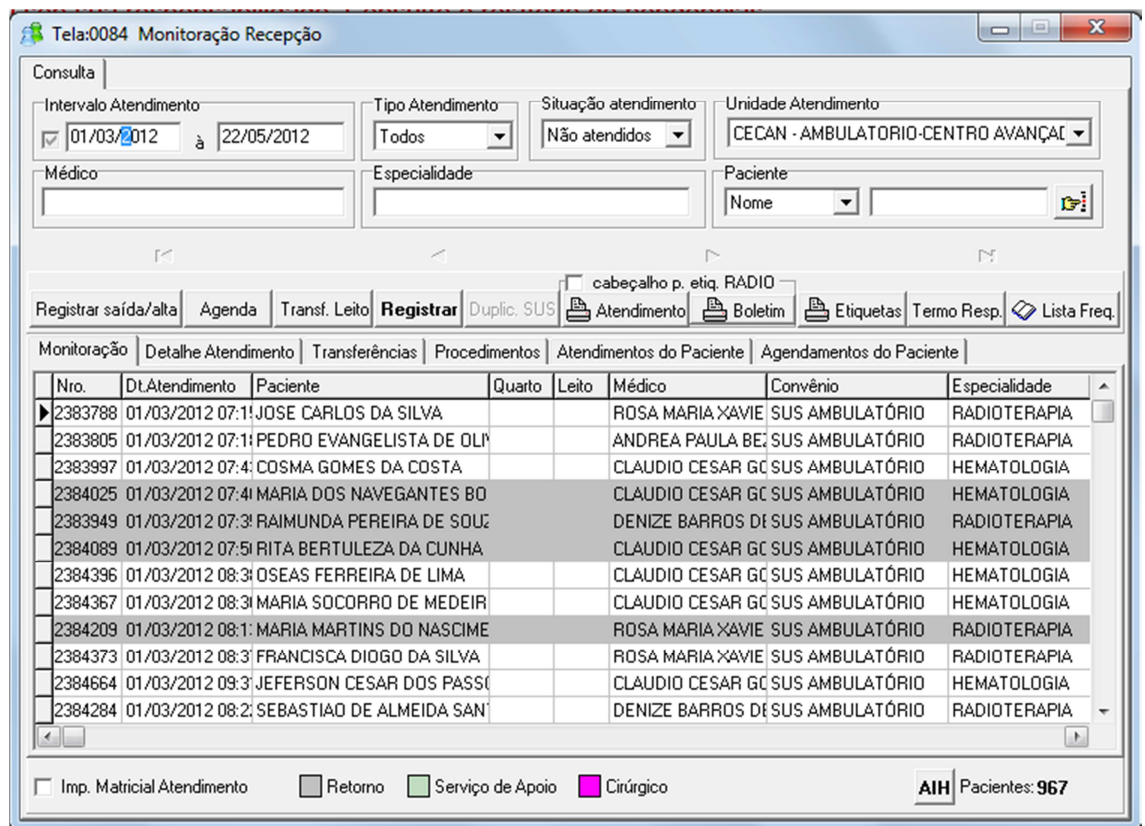


Figura 17 - Tela de Monitoração de Atendimento

Fonte: Resultado da pesquisa do autor.

Tela:0053 Atender

☐ Urgência
 ☒ Retorno
 ☐ 1ª Vez
 ☐ Autorizado SUS

Data Atendimento: 22/05/2012 10:46:03
 Unid. Atendimento: CECAN - AMBULATORIO-CENTRO AVANÇADO DE
 Tipo Clínica: MEDICA

Paciente:
 Responsável:
 Cidade Origem:

Informações Convênio
 Tipo Convênio:
 Nro. Carteira:
 Validade Carteira:

Nro Guia:
 Validade Guia:
 Tipo Guia:
 Senha Atendimento:

Unidade Internação:
 Quarto:
 Leito:
 Leitos:
 Leito (dias):

Especialidade Atendimento:
 Profissional:
 Peso (Kg.):
 Altura (m.):

Profissional Solicitante:
 CID:

Procedimentos Atendimento:
 < F3 > Incluir
 < F4 > Alterar
 < F5 > Gravar
 < F6 > Apagar

Procedimentos < F7 >
 Proced. Convênio < F8 >
 Procedimentos SUS < F9 >
☐ Proced.Principal

Qte.	Procedimento	Cód.Convênio	Proced. Convênio	Cód. SUS	Proced. SUS
▶					

Profissional Auxiliar:
 Especialidade:
 Profissional:
 Classificação:

OBS.:

Figura 18 - Tela de Atendimento.

Fonte: Resultado da pesquisa do autor

Tela:0091 Paciente

Consulta | Cadastro | Atendimentos | Agendamentos | Presc./Devol. | Moviment. Arquivo | Controle Médico

Nro. Paciente: 292272 | Nro. Arquivo: 12160.09 | Tipo Arquivo: FICHA | Nro. Pront.: | Nro. Ficha: | Etiquetas | Fichas | Imprimir Cadastro

Nome: TACITO BARRETO NETO | **Dt. Nasc.:** 04/01/1986 | **Idade:** 26 Anos, 4 Meses e 18 Dias | **Sexo:** M

CNS: | **Cidade Nascimento:** 240810 -NATAL - RN | **RG:** 2276320 | **Emissor:** SSP-RN | **Dt Emissão:** 05/01/2001 | **CPF:** 054.564.704-54

Nome Mãe: MARIA DE FATIMA FREIRE BAF | **Nome Pai:** FRANCISCO DE ASSIS BARRETO | **Estado Civil:** SOLTEIRO | **Cônjuge:**

Logradouro: RUA ANTONIO MADRUGA | **Nro.:** 1945 | **Complemento:** CASA 3 | **Bairro:** CAPIM MACIO

Cidade Endereço: 240810 -NATAL - RN | **CEP:** - | **Tel. Fixo:** 3642-0010 | **Tel. Trabalho:** | **Tel. Celular:** | **E-mail:**

Nacionalidade: Brasileiro | **Religião:** | **Cor Pele:** | **Raça:**

Escolaridade: | **Ocupação:**

Obs.:

Última Alteração de Cadastro: 27/04/2012 17:55:33

< F5 > Inserir | < F6 > Alterar | < F7 > Salvar | < ESC > Fechar tela

Figura 19 - Tela de cadastro do paciente

Fonte: Resultado da pesquisa do autor.

19. Proposta

O Módulo de recepção é um dos mais utilizados no hospital, é nele que é feito o cadastro do paciente, dado entrada do mesmo, impressão de relatórios, alta do paciente entre outras utilidades. A proposta é fazer um novo módulo chamado de Atendimento, onde seria voltado para entrada do paciente de forma mais prática e rápida em plataforma totalmente web.

Pelo fato do sistema atual ser desenvolvido em plataforma desktop utilizando IDE de programação Delphi, encontra-se dificuldade para seleção de novos profissionais para o projeto, já que o cenário atual de formação de profissionais em T.I é voltado para desenvolvimento web de linguagens orientadas a objetos.

Outra dificuldade encontrada está relacionada à comunicação entre as unidades, uma alocada no município de Caicó e a outra que é a base de dados

centralizada no Cekan, uma vez que a conexão é feita através de uma VPN entre ambas com velocidade limitada à 1 mb (um megabyte) , pois atualmente seria inviável o uso de uma conexão através de fibra ótica ou rádio por questões físicas de distância.

Outra questão, seria manter além da base centralizada na unidade Cekan ter uma base alocada no hospital de Caicó, para caso de perda de conexão, as informações sejam acessadas localmente de forma que quando a conexão volte, aconteça o sincronismo das informações tornando o infoliga um sistema de alta disponibilidade.

20. Escopo do Projeto

O escopo do projeto contém 10 cenários de casos de uso, com uma análise de tempo versus custo, existindo a possibilidade de aumento ou não do quadro de funcionários utilizando a metodologia SCRUM para mensurar os itens do escopo agregado de um tempo reservado para o plano de migração de software após término do projeto.

20.1. Tempo do Projeto versus Custo

O tempo do projeto será de acordo com as possibilidades de produção, tendo 2(duas) opções: manter-se com o quadro atual de funcionários aumentando o prazo de término ou contratando a princípio mais um funcionário com dedicação integral ao projeto diminuindo o tempo, porém aumentando o custo para instituição com folha de pagamento e encargos trabalhistas. Como a metodologia utilizada será o SCRUM, iremos estabelecer o desenvolvimento dos casos de uso em sprint's mensais podendo existir uma diferença entre as 2(duas) opções existentes. Abaixo, as possibilidades explicadas de forma detalhada.

20.1.1. Quadro atual de dois funcionários

Nessa opção, os 2(dois) funcionários existentes disponibilizariam de suas 44(quarenta e quatro) horas semanais de carga horária, 20(vinte) horas para pôr em prática o andamento do projeto proposto, ou seja, 4(quatro) horas por dia de trabalho. Cada Sprint terá 2(duas) semanas para haver maior iteratividade nos ciclos. Nesse caso o custo do projeto para a instituição seria a proporção de horas proporcional ao valor da hora trabalhada. Dessa maneira fica estabelecido o desenvolvimento dos casos de uso com a explicação abaixo:

1. Caso de uso Cadastrar Paciente:	3 sprint's;
2. Caso de uso Agendar Paciente:	4 sprint's;
3. Caso de uso Registrar Atendimento:	4 sprint's;
4. Caso de uso Registrar Serviço de Apoio:	1 sprint;
5. Caso de uso Atender Paciente:	2 sprint's;
6. Caso de uso Monitorar Atendimento:	2 sprint's;
7. Caso de uso Gerar Número de Ficha:	2 sprint's;
8. Caso de uso Protocolar Arquivo Paciente:	1 sprint;
9. Caso de uso Alterar Atendimento:	3 sprint's;
10. Caso de uso Gerar Alta/Encaminhamento Paciente:	2 sprint's;

Nesse modo, totalizando em um total de 24 sprint's, ou seja, 12 meses de duração.

20.1.2. Quadro futuro com três funcionários

Nessa opção, os 2(dois) funcionários existentes disponibilizariam de suas 44(quarenta e quatro) horas semanais de carga horária, 15(quinze) horas para pôr em prática o andamento do projeto proposto, ou seja, 3(três) horas por dia de trabalho e o novo funcionário, dedicaria tempo integral para o projeto. Cada Sprint terá 2(duas) semanas para haver maior iteratividade nos ciclos. Nesse caso o custo do projeto para a instituição seria a proporção de horas proporcional ao valor da hora trabalhada dos dois funcionários existentes, agregado ao salário do novo

funcionário como foi citado anteriormente. Dessa maneira fica estabelecido o desenvolvimento dos casos de uso com a quantidade de sprint's abaixo:

- | | |
|---|--------------|
| 1. Caso de uso Cadastrar Paciente: | 2 sprint's; |
| 2. Caso de uso Agendar Paciente: | 2 sprint's; |
| 3. Caso de uso Registrar Atendimento: | 2 sprint's; |
| 4. Caso de uso Registrar Serviço de Apoio: | 1 sprint; |
| 5. Caso de uso Atender Paciente: | 2 sprint's; |
| 6. Caso de uso Monitorar Atendimento: | 12 sprint's; |
| 7. Caso de uso Gerar Número de Ficha: | 1 sprint's; |
| 8. Caso de uso Protocolar Arquivo Paciente: | 1 sprint; |
| 9. Caso de uso Alterar Atendimento: | 2 sprint's; |
| 10. Caso de uso Gerar Alta/Encaminhamento Paciente: | 1 sprint; |

Nesse modo, totalizando em um total de 13 sprint's, ou seja, 6 meses e meio de duração.

20.2. Plano de Migração

Pelo fato do sistema já ser utilizado com o mesmo propósito, porém em uma plataforma desktop, estima-se que o plano de migração de software não haja risco. Como o banco de dados a ser utilizado será o Sql Server 2005, que é o existente na instituição, serão criadas as classes e o mapeamento objeto relacional utilizando o sistema em 2(duas) plataformas com a mesma base, até permanecer somente o sistema em plataforma web. Quanto à necessidade de treinamento e apresentação dos módulos, foi levantada à necessidade de 3(três) horas de apresentação para cada unidade hospitalar e um manual com os cenários e descrições dos casos de uso para às chefias de recepção.

21. Conclusão

O projeto proposto neste trabalho é a transição de um sistema em plataforma cliente servidor desenvolvido em Delphi para um sistema WEB, por isso, conjectura-se a máxima manutenção dos padrões atuais de interatividade com o usuário, facilitando o treinamento dos mesmos e aproveitando a prática atual dos colaboradores. Além dessa transição de tecnologia, a implantação de uma metodologia ágil como o SCRUM, facilitará no cumprimento do prazo descrito no cronograma fazendo com que a análise de custo do desenvolvimento do sistema esteja dentro da realidade provisionada. A conclusão desse projeto impactará em um novo conceito de desenvolvimento e soluções de software na Liga Norte-riograndense contra o Câncer, melhorando na qualidade de atendimento e tratamento de pessoas.

22. REFERÊNCIAS

DEITEL, H.M. **Java Como Programar 6ª edição**, são Paulo, Pearson Prentice Hall, 2008.

HOTEK, Mike; MACKIN, J. C. **Mcitp Self Paced Training Kit (Exam 70-443)**, Microsoft press, U.S., 2006.

JAVA, **MVC**. Disponível em: <<http://java.sun.com/blueprints/patterns/MVC-detailed.html>> Acesso em 02 dez. 2012.

JBOSS, **Hibernate – Annotations**. Disponível em: <http://docs.jboss.org/hibernate/annotations/3.5/reference/en/html_single/> Acesso em 20 dez 2012.

ORACLE, **JavaServer Faces**. Disponível em: <<http://www.oracle.com>> Acesso em 23 dez. 2011.

ORACLE, **JavaServer Faces**. Disponível em: <<http://www.oracle.com>> Acesso em 23 dez. 2011

ORACLE, Java SE Application Design With MVC. Disponível em: <<http://www.oracle.com/technetwork/articles/javase/mvc-136693.html#1>> Acesso em 07 nov. 2011.

SBIS, **Informática em Saúde**. Disponível em: <<http://www.sbis.org.br/indexframe.html>> Acesso em 10 dez. 2011.

SCRUMBRASIL, **SCRUM**. Disponível em: <<http://scrumbrasil.blogspot.com.br/2011/09/scrum-em-30-segundos.html>> Acesso em 15 dez. 2011.

TIEXPERT, **Introdução à Plataforma Java**. Disponível em: <<http://www.tiexpert.net/programacao/java/introducao.php>> Acesso em 10 jun. 2012.