



Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"

ROGÉRIO APARECIDO DE OLIVEIRA MUNHOZ

SISTEMA DE GESTÃO PARA AMBULATÓRIO MÉDICO OCUPACIONAL

Assis/SP
2015

ROGÉRIO APARECIDO DE OLIVEIRA MUNHOZ

SISTEMA DE GESTÃO PARA AMBULATÓRIO MÉDICO OCUPACIONAL

Orientador: Prof. Esp. Diomara Martins Reigato Barros

Área de concentração: Análise e Desenvolvimento de Sistemas

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, como requisito do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas.

FICHA CATALOGRÁFICA

MUNHOZ, Rogério Aparecido de Oliveira

Sistema de Gestão para Ambulatório Médico Ocupacional / Rogério Aparecido de Oliveira

Munhoz. Fundação Educacional do Município de Assis - Assis, 2015.

58 p.

Orientador: Prof. Esp. Diomara Martins Reigato Barros

Trabalho de Conclusão de Curso

Instituto Municipal de Ensino Superior de Assis – IMESA.

1. Ambulatório. 2. PHP.

CDD: 001.61
Biblioteca da FEMA

SISTEMA DE GESTÃO DE AMBULATÓRIO MÉDICO OCUPACIONAL

ROGÉRIO APARECIDO DE OLIVEIRA MUNHOZ

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, como requisito do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, analisado pela seguinte comissão examinadora:

Orientador: Prof. Esp. Diomara Martins Reigato Barros

Analisador: Prof. Dr. Luiz Ricardo Begosso

Assis/SP
2015

DEDICATÓRIA

Dedico esse trabalho à Deus, meus pais e minhas filhas, e em especial à minha noiva Renata que sempre esteve ao meu lado, me dando incentivo desde o início do curso; e por fim, a todos os amigos.

AGRADECIMENTOS

Em primeiro lugar a Deus, por ter me dado forças para me manter firme em todos os momentos de dificuldade, além da sabedoria necessária para a realização deste trabalho.

À meus pais José Roberto e Antonia Maria, por estarem ao meu lado sempre que necessário, e minhas filhas Lavínia e Lorena, por me alimentarem de esperança a cada dia de dificuldade.

Em especial à Renata Mazarim minha noiva e principal incentivadora, por me encorajar a iniciar e não me deixar desistir quando me faltava forças para continuar e também meus amigos pelo ajuda e companheirismo no decorrer do curso.

“Tudo o que um sonho precisa para ser realizado
é alguém que acredite que ele possa ser realizado.”

Roberto Shinyashiki

RESUMO

Este projeto tem como objetivo ser uma ferramenta de auxílio no gerenciamento de uma ambulatório ocupacional, proporcionando aos usuários uma maior facilidade na execução de suas tarefas diárias e cumprimento de normas legais.

O sistema será responsável por armazenar de forma segura as informações referentes aos colaboradores da empresa usuária, possibilitando um melhor controle e acesso, além auxiliar seus gestores em tomadas de decisões e controle de gastos. Este sistema Web foi desenvolvido utilizando a tecnologia PHP, que é uma linguagem extremamente voltada ao desenvolvimento web, com base de dados em MySQL, proporcionando aos seus usuários rapidez e segurança no acesso aos dados.

Este projeto foi desenvolvido com base no framework Laravel, que utiliza o padrão MVC (Model View Controller) em suas aplicações, possibilitando que cada camada seja acessada e modificada individualmente sem ter que modificar outras partes do software, além da alto reuso de código e facilidade na manutenção do mesmo.

Palavras-chaves: PHP; Laravel; Ambulatório.

ABSTRACT

This project aims to be a tool to aid in managing an occupational clinic, providing users with greater ease in performing their daily tasks and compliance with legal standards.

The system is responsible for securely store information relating to reviewers user company, enabling better control and access, as well assist their managers in decision-making and cost control. This Web system was developed using PHP technology, which is an extremely focused on web development language with database in MySQL, providing its users fast and secure access to data.

This project was developed based on Laravel framework, which uses the MVC (Model View Controller) into their applications, allowing each layer to be accessed and modified individually without having to modify other parts of the software, as well as high reuse code and ease in maintenance.

Keywords: PHP; Laravel; Clinic.

LISTA DE ILUSTRAÇÕES

Figura 1 - Estrutura Analítica do Projeto	24
Figura 2 - Mapa Mental	29
Figura 3 - Caso de Uso: Geral.....	30
Figura 4 - Caso de Uso: Auxiliar de Enfermagem (Geral)	31
Figura 5 - Caso de Uso: Key User (Geral).....	32
Figura 6 - Caso de Uso: Médico (Geral).....	33
Figura 7 - Caso de Uso: Agendar Consulta.....	34
Figura 8 - Caso de Uso: Manter Atestados	36
Figura 9 - Caso de Uso: Manter Estoque	38
Figura 10 - Caso de Uso: Manter Vacinas	39
Figura 11 - Caso de Uso: Realizar Atendimento de Enfermagem.....	40
Figura 12 - Caso de Uso: Manter Cargos.....	41
Figura 13 - Caso de Uso: Manter Empresas	42
Figura 14 - Caso de Uso: Manter Funcionários.....	43
Figura 15 - Caso de Uso: Manter Medicamentos	44
Figura 16 - Caso de Uso: Manter Médicos.....	45
Figura 17 - Caso de Uso: Manter Relatórios	46
Figura 18 - Caso de Uso: Realizar Consulta Médica.....	48
Figura 19 - Diagrama de Classes.....	49
Figura 20 - Agendar Consulta	50
Figura 21 - Atendimento de Enfermagem	51
Figura 22 - Diagrama de Sequência: Manter Atestados.....	52
Figura 23 - Diagrama de Sequência: Manter Estoque	53
Figura 24 - Manter Vacinas	54
Figura 25 - Diagrama de Sequência: Emitir Receituário	55
Figura 26 - Diagrama ER	56

LISTA DE TABELAS

Tabela 1 - Lista de Eventos.....	27
Tabela 2 - Agendar Consulta.....	35
Tabela 3 - Manter Atestados	37
Tabela 4 - Manter Estoque	38
Tabela 5 - Manter Vacinas	39
Tabela 6 - Realizar Atendimento de Enfermagem.....	40
Tabela 7 - Manter Cargos.....	41
Tabela 8 - Manter Empresas.....	42
Tabela 9 - Manter Funcionários.....	43
Tabela 10 - Manter Medicamentos.....	44
Tabela 11 - Manter Médicos.....	45
Tabela 12 - Manter Relatórios	47
Tabela 13 - Realizar Consulta Médica	48

LISTA DE ABREVIATURAS E SIGLAS

API - Application Programming Interface;

CID - Classificação Internacional de Doenças;

HTML - HyperText Markup Language;

HTTP - Hypertext Transfer Protocol;

IDE - Integrated Development Environment;

MVC - Model View-Controller;

PHP - Hypertext Preprocessor

SGBD - Sistema Gerenciador de Banco de Dados;

SQL - Structured Query Language;

UML - Unified Modeling Language;

WBS - Work Breakdown Structure;

XML - Extensible Markup Language.

SUMÁRIO

1. INTRODUÇÃO	15
1.1. OBJETIVOS	15
1.2. JUSTIFICATIVAS.....	16
1.3. MOTIVAÇÕES	16
1.4. ESTRUTURA DO TRABALHO.....	16
2. METODOLOGIA DO DESENVOLVIMENTO	18
2.1. MÉTODO DE ANÁLISE	19
2.2. PHP.....	19
2.3. LARAVEL.....	20
2.4. APACHE	21
2.5. MYSQL.....	22
2.6. SUBLIME TEXT	22
3. ESTRUTURA ANALÍTICA DO PROJETO	23
4. LEVANTAMENTO E ANÁLISE DE REQUISITOS	25
4.1. DETALHAMENTOS DO PROBLEMA A SER RESOLVIDO.....	25
4.2. RESULTADOS ESPERADOS NA IMPLEMENTAÇÃO DO SOFTWARE.....	25
4.3. FORMA ADOTADA PARA LEVANTAMENTO DOS REQUISITOS	26
4.4. RESTRIÇÕES DE DESENVOLVIMENTO DO SOFTWARE.....	26
4.5. PROBLEMAS POTENCIAIS	26
4.6. PRIORIZAÇÃO DA IMPLANTAÇÃO DOS REQUISITOS	26
4.7. LISTA DE EVENTOS	27
5. DIAGRAMAS	28
5.1. MAPA MENTAL	28
5.2. DIAGRAMA DE CASO DE USO	29
5.2.1. Caso de Uso: Geral	30
5.2.2. Auxiliar de Enfermagem: Geral.....	31
5.2.3. Key User: Geral	32
5.2.4. Médico: Geral	33
5.3. ESPECIFICAÇÃO DOS CASOS DE USO	34
5.3.1. Agendar Consulta	34
5.3.2. Manter Atestados.....	36
5.3.3. Manter Estoque	38
5.3.4. Manter Vacinas.....	39
5.3.5. Realizar Atendimento de Enfermagem	40
5.3.6. Manter Cargos.....	41
5.3.7. Manter Empresas	42

5.3.8. Manter Funcionários	43
5.3.9. Manter Medicamentos	44
5.3.10. Manter Médicos	45
5.3.11. Manter Relatórios	46
5.3.12. Realizar Consulta Médica	48
6. DIAGRAMA DE CLASSE	49
7. DIAGRAMA DE SEQUÊNCIA	50
7.1. Agendar Consulta	50
7.2. Atendimento de Enfermagem.....	51
7.3. Manter Atestados	52
7.4. Manter Estoque.....	53
7.5. Manter Vacinas	54
7.6. Emitir Receituário	55
8. DIAGRAMA ER	56
9. REFERÊNCIAS	57

1. INTRODUÇÃO

Atualmente a era da informação vem crescendo cada vez mais em ritmo acelerado e com esse fato todos os setores públicos e privados estão se atualizando.

Este projeto tem por objetivo criar um sistema capaz de gerenciar e documentar as informações geradas no ambiente de trabalho com referência à saúde ocupacional de colaboradores de uma empresa, independentemente de seu ramo de atuação, ou clínicas médicas terceirizadas.

1.1. OBJETIVOS

Este projeto tem como objetivo o desenvolvimento de um sistema capaz de informatizar as informações geradas no ambulatório médico em âmbito ocupacional, tais como observações médicas, ocorrências relativas à saúde de cada colaborador, visando uma melhor organização no atendimento da equipe de enfermagem e médica, facilitando o dia a dia de trabalho e evitando perda de informações que poderiam causar sérias complicações futuras a empresa envolvida.

Com a utilização deste software o usuário será capaz de armazenar informações referentes a observações encontradas no atendimento diário, registrar atestados médicos entregues por cada funcionário a fim de controlar o absenteísmo (falta do trabalho) e manipular relatórios para melhor controle, registrar a anamnese (exame físico), controlar a validade de vacinas, emitir receituário médico para medicação a ser administrada internamente, realizar controle da medicação entregue aos beneficiários, emitir documentos obrigatórios pela legislação. Para melhor gerenciamento deste software e controle de informações os usuários envolvidos poderão ser divididos em diferentes níveis de acessos, possibilitando assim restringir o acesso a determinadas informações, por exemplo, anotações médicas.

Com a utilização desta ferramenta, os funcionários envolvidos passarão a possuir além de um prontuário físico (papel) para cumprimento de leis, um prontuário on-line que possibilitará maior facilidade na consulta de informações e melhor controle da vida ocupacional enquanto funcionário da empresa usuária.

1.2. JUSTIFICATIVAS

A necessidade do desenvolvimento deste sistema se dá para atender as empresas que possuem ambulatório médico, e necessitem de uma ferramenta capaz de auxiliar nas tarefas operacionais, possibilitando maior agilidade e eficiência na absorção de dados gerados e maior controle sobre estes dados.

O sistema pode trazer contribuições significativas, tais como evitar perda de informações além de facilitar o acesso as mesmas, possibilitando também maior segurança em seu armazenamento.

Através deste sistema o detalhamento das informações poderá ser melhorado, assim como a transmissão destes dados aos gestores responsáveis da empresa usuária.

1.3. MOTIVAÇÕES

Facilitar o dia a dia de um ambulatório médico, auxiliando na gestão da saúde ocupacional de seus funcionários, a fim de atingir metas e obrigações legais, além de evitar perda de informações e melhor o gerenciamento das mesmas.

1.4. ESTRUTURA DO TRABALHO

Este projeto está dividido em capítulos que serão explicitados a seguir.

O primeiro capítulo apresenta a contextualização e a justificativa para o desenvolvimento deste projeto, juntamente aos objetivos pretendidos e motivações encontradas para o andamento do trabalho.

No segundo capítulo é realizado uma abordagem sobre os conceitos de fundamentação teórica referente às tecnologias utilizadas durante o desenvolvimento deste projeto.

O terceiro capítulo apresenta o processo EAP, que nada mais é do que o gerenciamento de entregas do trabalho, melhorando assim o controle e manutenção dos chamados pacotes de trabalho.

O quarto capítulo aborda as formas utilizadas para o levantamento de requisitos necessários para a criação deste projeto, assim como o nicho de mercado a qual este projeto visa a fazer parte, além dos principais problemas que podem vir a surgir, é apresentado às prioridades e necessidades para implantação do projeto, juntamente a listagem de eventos encontrados no levantamento de requisitos.

O quinto, sexto e sétimo capítulos apresentam as etapas de análise e especificações do sistema contemplando os principais requisitos levantados. Apresentando ainda Mapa Mental e os diagramas UML, Caso de Uso e suas narrativas, Classe e Sequencia.

O oitavo capítulo apresenta o cronograma atual do projeto, mostrando as etapas realizadas desde seu início até os dias atuais.

2. METODOLOGIA DO DESENVOLVIMENTO

Neste capítulo serão apresentadas as tecnologias e ferramentas utilizadas para o desenvolvimento do sistema AmbSystem.

AmbSystem é um sistema Web desenvolvido com a linguagem de programação PHP (Hypertext Preprocessor), que é uma linguagem de programação de ampla utilização, interpretada, que é especialmente interessante para desenvolvimento para a Web e pode ser mesclada dentro do código HTML. O ambiente de desenvolvimento é baseado no Sublime Text 2 que é uma poderosa IDE de desenvolvimento que conta com a maioria dos recursos disponíveis nas principais IDEs, suporte nativo a várias linguagens de programação.

A arquitetura do projeto faz uso do padrão MVC (Model, View, Controller), que se baseia na separação de dados (modelo), da interface com o usuário (view) e da lógica de negócio (controller), possibilitando que cada camada seja acessada e modificada individualmente sem ter que modificar outras partes do software, além da alta reusabilidade de código e facilidade na manutenção do mesmo. Para isso será utilizado o framework Laravel em sua versão 5.0. Laravel é um framework PHP para desenvolvimento web que utiliza a arquitetura MVC.

O servidor de aplicação será o Apache Tomcat, através da ferramenta XAMPP que é um pacote com os principais servidores de código aberto, incluindo FTP, banco de dados MySQL com suporte à linguagem PHP.

O repositório de informações fica a cargo do banco de dados MySQL que é um sistema de gerenciamento de banco de dados (SGBD), que utiliza a linguagem SQL como interface e atualmente um dos bancos de dados mais populares no mundo, oferece também excelente desempenho, estabilidade e fácil manuseio.

2.1. MÉTODO DE ANÁLISE

O projeto será desenvolvido através de uma análise do problema apresentado pelo cliente, utilizando a metodologia orientada a objetos (POO – Programação Orientada a Objetos), pois se trata de uma abordagem de programação que serve de elo entre os problemas existentes e as soluções computacionais apresentadas no campo da programação da programação.

A POO apresenta-se como um paradigma de programação que permite aos programadores raciocinar e solucionar problemas em termos de objetos, os quais estão diretamente associados as entidades ou “coisas” reais. Como resultado desse mapeamento natural, utilizando a POO um programador pode concentrar nos objetos que compõem o sistema, em vez de entrar vislumbrar o sistema como um conjunto de procedimentos e dados. Os benefícios resultantes de empregar a POO coo abordagem de programação não se restringem a raciocinar e resolver problemas em termos de objetos ou entidades reais, mas implicam a reutilização de código (S. FILHO; ANTONIO MENDES, 2010).

2.2. PHP

O PHP como é conhecido hoje, é na verdade o sucessor para um produto chamado PHP/FI. Criado em 1994 por Rasmus Lerdof, a primeira encarnação do PHP foi um simples conjunto de binários Common Gateway Interface (CGI) escrito em linguagem de programação C. Originalmente usado para acompanhamento de visitas para seu currículo online, ele nomeou o conjunto de scripts de "Personal Home Page Tools" mais frequentemente referenciado como "PHP Tools." Ao longo do tempo, mais funcionalidades foram desejadas, e Rasmus reescreveu o PHP Tools, produzindo uma maior e rica implementação. Este novo modelo foi capaz de interações com Banco de Dados e mais, fornecendo uma estrutura no qual os usuários poderiam desenvolver simples e dinâmicas aplicações web, como um livros de visitas. Em Junho de 1995, Rasmus liberou o código fonte do PHP Tools para o público, o que permitiu que desenvolvedores usassem da forma como desejassem.

Isso permitiu - e encorajou - usuários a fornecerem correções para bugs no código, e em geral, aperfeiçoá-lo.

Em Setembro do mesmo ano, Rasmus expandiu o PHP e - por um breve período - mudou o nome PHP. Agora referindo-se a ferramenta como FI, abreviação para "Forms Interpreter", a nova implementação incluiu algumas funcionalidades básicas do PHP como bem conhecemos hoje. A linguagem foi desenvolvida para, deliberadamente, ser parecida com C, tornando-a fácil para ser adotada por desenvolvedores habituados com C, Perl e linguagens similares.

Atualmente o PHP encontra-se em sua versão 5 está lançada em Julho de 2004 após um longo desenvolvimento e vários pré-lançamentos. Principalmente impulsionado pelo seu core o Zend Engine 2.0 com um novo modelo de objeto e dezenas de outros novos recursos.

Alguns melhoramentos desta nova versão são a sua rapidez, pois agora primeiro compila-se e depois executa-se, quando antes se executava ao mesmo tempo que se interpretava o código. Outra qualidade é a sua independência do servidor web, criando versões de PHP nativas para mais plataformas, e também um API mais elaborado e com mais funções.

Neste último ano, o número de servidores que utilizam PHP aumentou, conseguindo situar-se perto dos 5 milhões de sites e 800.000 endereços IP, o que converteu PHP numa tecnologia popular. Isto deve-se, entre outras razões, ao fato de PHP ser o complemento ideal para que o Linux-Apache seja compatível com a programação do lado do servidor de sítios web (PHP; Manual PHP).

2.3. LARAVEL

Laravel é um framework de aplicações web open-source, criado por Taylor Otwell em Junho de 2011, sendo destinado ao desenvolvimento de aplicações web utilizando a arquitetura MVC, atualmente este framework encontra-se em sua versão 5.1, e é considerado como um dos mais populares framework PHP.

O Laravel fornece poderosas ferramentas nativas afim de facilitar tarefas comuns na maioria dos projetos web de pequeno e grande porte, tais como o Eloquent, um ORM que "repensa" completamente tudo o que tem a ver com base de dados. Além

da ferramenta Eloquent que auxilia na comunicação com o banco de dados, além de oferecer métodos capazes de substituir as chamadas SQL, sendo métodos simples e de fácil entendimento como o método find, save e delete.

Além disso o Laravel oferece a classe Middleware para melhor controle de acesso e autenticação de dados e uma classe chamada Request que auxilia o desenvolvedor na validação de dados e queries customizadas.

O Laravel também apresenta inovações PHP, para melhor manipulação de templates, rotas, migrations, classes RESTful e grande suporte através de sua documentação e comunidade (SILVA; SARA, 2013).

2.4. APACHE

Ao acessar qualquer site, há um servidor por trás daquele endereço responsável por disponibilizar as páginas e todos os demais recursos que você pode acessar. Assim, ao enviar um e-mail através de um formulário, coloca uma mensagem em um fórum de discussão, faz uma compra on-line, etc, um servidor Web (ou um conjunto de servidores) é responsável por processar todas essas informações. Sendo mais claro, um servidor Web é um computador que processa solicitações HTTP (Hyper TextTransfer Protocol), o protocolo padrão da Web.

Como servidor Web, o Apache é o mais conhecido e usado. Os motivos incluem sua excelente performance, segurança, compatibilidade com diversas plataformas e todos os seus recursos.

O servidor Apache (ou Apache Server) surgiu no National Center of Supercomputing Applications (NCSA) através do trabalho de Rob McCool, e teve sua primeira versão oficial (0.6.2) lançada em 1995.

O Apache Server é um software livre, o que significa que qualquer um pode estudar ou alterar seu código-fonte, além de poder utilizá-lo gratuitamente. É graças a essa característica que o software foi (e continua sendo) melhorado ao passar dos anos. Além de estar disponível para o Linux (e para outros sistemas operacionais baseados no Unix), o Apache também conta com versões para o Windows.

O servidor Apache é capaz de executar código em PHP, Perl, Shell Script e até em ASP e pode atuar como servidor FTP, HTTP, entre outros. Sua utilização mais

conhecida é a que combina o Apache com a linguagem PHP e o banco de dados MySQL (ALECRIM; EMERSON, 2006).

2.5. MYSQL

O MySQL é um sistema gerenciador de banco de dados relacional de código aberto usado na maioria das aplicações gratuitas para gerir suas bases de dados. O serviço utiliza a linguagem SQL (Structure Query Language – Linguagem de Consulta Estruturada), que é a linguagem mais popular para inserir, acessar e gerenciar o conteúdo armazenado num banco de dados.

O sistema foi desenvolvido pela empresa sueca MySQL AB e publicado, originalmente, em maio de 1995. Após, a empresa foi comprada pela Sun Microsystems e, em janeiro de 2010, integrou a transação bilionária da compra da Sun pela Oracle Corporation. Atualmente, a Oracle, embora tenha mantido a versão para a comunidade, tornou seu uso mais restrito e os desenvolvedores criaram, então, o projeto MariaDB para continuar desenvolvendo o código da versão 5.1 do MySQL, de forma totalmente aberta e gratuita (PISA; PEDRO, 2012).

2.6. SUBLIME TEXT

Um editor de texto é sem dúvida uma ferramenta essencial em qualquer sistema operativo. Além das funcionalidades básicas de edição de texto, este tipo de ferramentas integram hoje muitas funções direcionadas para a programação. Sublime Text é um editor de texto e código-fonte multiplataforma, escrito em linguagem C++. Este editor oferece recursos extraordinários e um desempenho simplesmente surpreendente.

Das muitas funcionalidades que oferece, destacam-se a: Possibilidade de usar múltiplas seleções para renomear facilmente variáveis, possibilidade de usar também múltiplas seleções em batch files, sugestão de comandos, suporte para expressões regulares. Sublime Text possui versões testes e também versões pagas (PINTO; PEDRO, 2013).

3. ESTRUTURA ANALÍTICA DO PROJETO (EAP)

A EAP é definida como o processo de subdivisão das entregas e do trabalho do projeto em componentes menores e mais facilmente gerenciáveis. O principal benefício desse processo é o fornecimento de uma visão estruturada do que deve ser entregue.

Segundo o PMBOK a EAP é uma decomposição hierárquica do escopo total do trabalho a ser executado pela equipe do projeto a fim de alcançar os objetivos do projeto e criar as entregas requeridas. A EAP organiza e define o escopo total do projeto e representa o trabalho específico na atual declaração do escopo do projeto aprovada.

O trabalho planejado é contido dentro dos componentes de nível mais baixo da EAP, que são chamados de pacotes de trabalho. Um pacote de trabalho pode ser usado para agrupar as atividades onde o trabalho é agendado, tem seu custo estimado, monitorado e controlado. No contexto da EAP, o trabalho se refere a produtos de trabalho ou entregas que são o resultado da atividade e não a atividade propriamente dita (PMI; PMBOK 5º ed.).

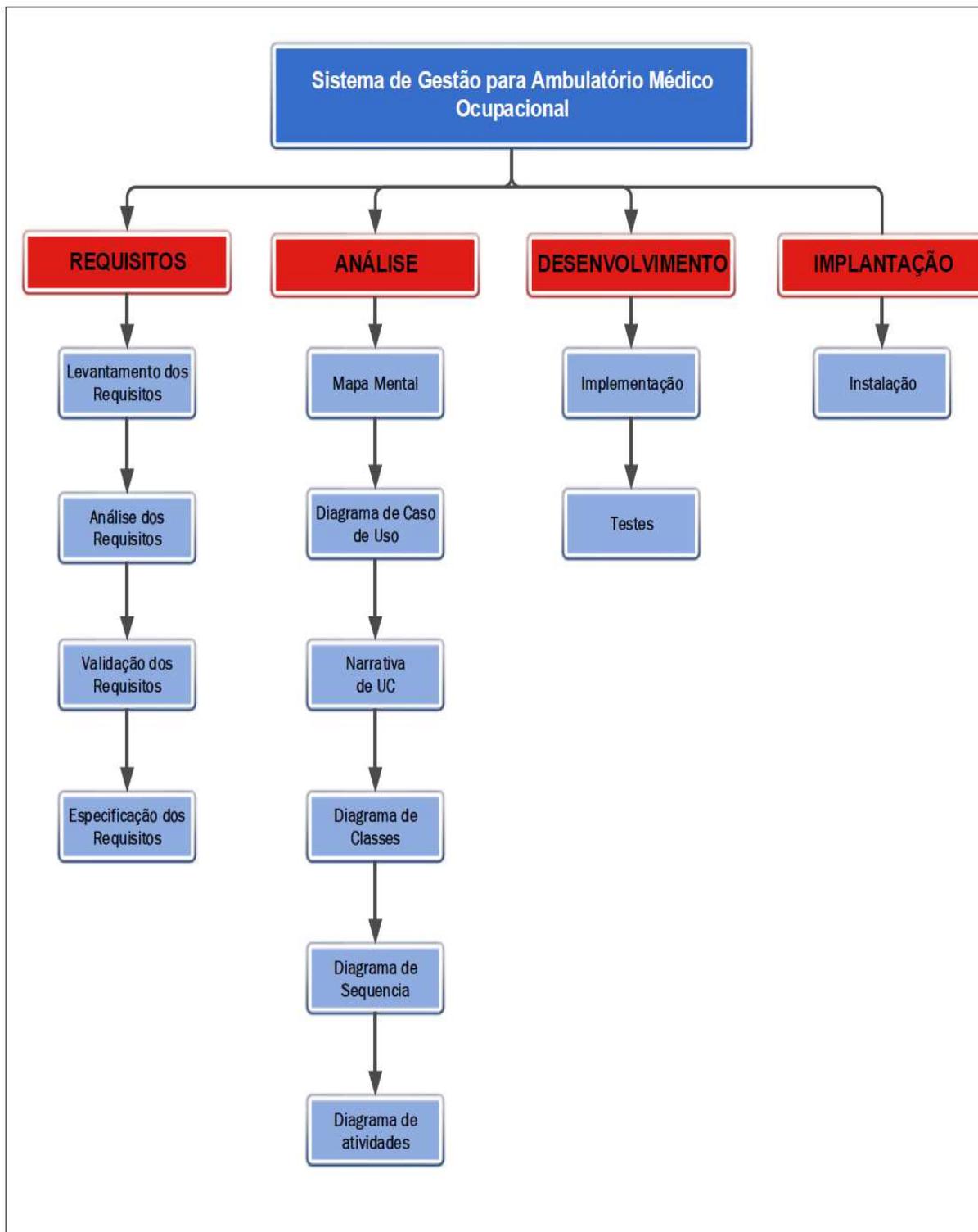


Figura 1 - Estrutura Analítica do Projeto

4. LEVANTAMENTO E ANÁLISE DE REQUISITOS

Para o desenvolvimento do sistema foi realizada entrevista padrão com envolvidos no ambiente ambulatorial, tais como Enfermeiras, Auxiliar de Enfermagem e Médico, considerando sempre a necessidade presente em médias e grandes empresas em administrar a vida ocupacional de seus funcionários e também a gestão interna de um ambulatório médico.

4.1. DETALHAMENTOS DO PROBLEMA A SER RESOLVIDO

Diversas empresas de médio e grande porte sofrem com a falta de um sistema estável e eficaz, para o controle da situação ocupacional de seus colaboradores, podendo assim causar o não cumprimento de normas legais e futuras ações judiciais.

Visto a necessidade do cumprimento das normas referentes à saúde do trabalhador e do trabalho, empresas encontram dificuldades no controle de validades de exames e na organização de observações encontradas ou relatadas no dia a dia dos trabalhadores, observações estas que podem fazer a diferença em defesas judiciais que podem vir a ocorrer.

Existe também um controle equivocado ou errado de informações geradas pelos envolvidos, tais como números de atestados médicos apresentados, quais doenças tem causado mais faltas no trabalho e diversas outras informações que podem auxiliar um gestor em tomadas de decisões a curto ou longo prazo.

4.2. RESULTADOS ESPERADOS NA IMPLEMENTAÇÃO DO SOFTWARE

Espera-se obter maior controle sobre todas as informações relativas ao que se trata da saúde dos trabalhadores envolvidos no ambiente de trabalho da empresa, além de uma melhor organização e fácil acesso as informações com velocidade e precisão. Melhor controle de custos derivados a materiais e amentos gastos no ambulatório da empresa.

4.3. FORMA ADOTADA PARA LEVANTAMENTO DOS REQUISITOS

Análise de como é realizado o atendimento médico e atendimento de enfermagem, como funciona o armazenamento de materiais e medicamentos, como é realizado a administração de informações e quais são as normas legais que devem ser cumpridas e quais documentos devem ser emitidos e qual o tempo devem ser mantidos para melhor entendimento dos problemas e resolução dos mesmos.

4.4. RESTRIÇÕES DE DESENVOLVIMENTO DO SOFTWARE

Para execução do sistema, será necessário um microcomputador com acesso a internet, navegador Google Chrome ou Firefox.

4.5. PROBLEMAS POTENCIAIS

- Controle de estoque;
- Controle de informações;
- Emitir relatórios diversos;
- Emissão de documentos.

4.6. PRIORIZAÇÃO DA IMPLANTAÇÃO DOS REQUISITOS

- Controle de funcionários;
- Observações médicas;
- Gestão de atestados;
- Emissão de relatórios.

4.7. LISTA DE EVENTOS

Nº	ATOR	DESCRIÇÃO	CASO DE USO
1	Aux. de Enfermagem	Realiza lançamento de atestados	Manter Atestados
2	Aux. de Enfermagem	Realiza a conferencia de estoque	Manter Estoque
3	Aux. de Enfermagem	Cadastra ou altera datas de vacinas	Manter Vacinas
4	Aux. de Enfermagem	Realiza atendimento de enfermagem	Realizar Atendimento de Enfermagem
5	Key User	Cadastra uma agenda médica	Manter Agenda
6	Key User	Cadastra os cargos da empresa	Manter Cargos
7	Key User	Cadastra as informações de uma empresa	Manter Empresas
8	Key User	Cadastra ou altera um funcionário	Manter Funcionários
9	Key User	Cadastra medicamentos	Manter Medicamentos
10	Key User	Cadastra ou altera um médico	Manter Médico
11	Key User	Realiza emissão de relatórios	Manter Relatórios
12	Médico	Realiza consulta médica	Realizar Consulta Médica

Tabela 1 - Lista de Eventos

5. DIAGRAMAS

Este capítulo apresenta as análises e especificações do sistema, com os principais requisitos levantados para o desenvolvimento da proposta de trabalho. Também apresenta o mapa mental dos principais diagramas UML (Casos de Uso, classe e sequência).

5.1. MAPA MENTAL

Mapa Mental é uma ferramenta que permite a memorização, organização e representação da informação com o propósito de facilitar os processos de aprendizagem, administração e planejamento organizacional, assim como, a tomada de decisão.

A técnica dos Mapas Mentais foi desenvolvida pelo britânico Tony Buzan com o objeto de fortalecer as conexões sinápticas que têm lugar entre os neurônios do córtex cerebral e que fazem praticamente todas as atividades intelectuais do ser humano. A construção do Mapa Mental foi realizada utilizando o software FreeMind.

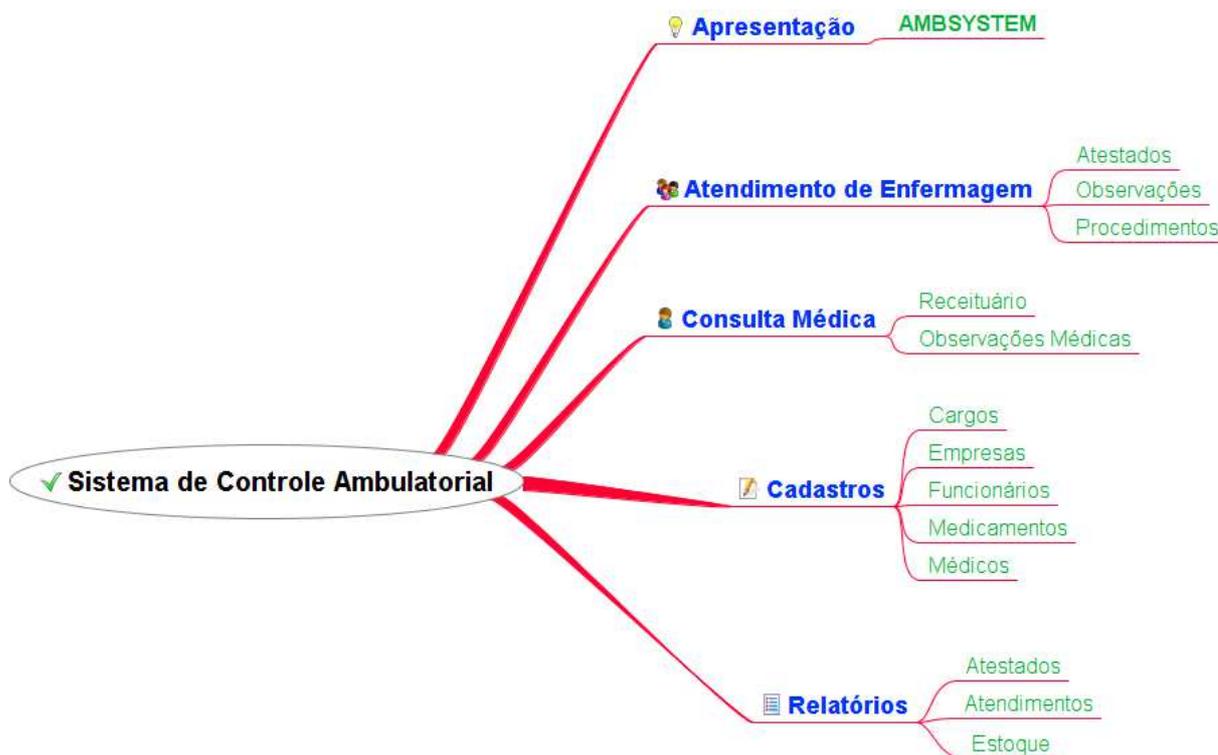


Figura 2 - Mapa Mental

5.2. DIAGRAMA DE CASO DE USO

Os casos de uso descrevem as principais funcionalidades do sistema e interação dessas funcionalidades com os usuários. Casos de uso retratam de forma geral as ações do usuário no sistema.

5.2.1. Caso de Uso: Geral

A figura abaixo ilustra as iterações do Auxiliar de Enfermagem com o sistema e suas ações.

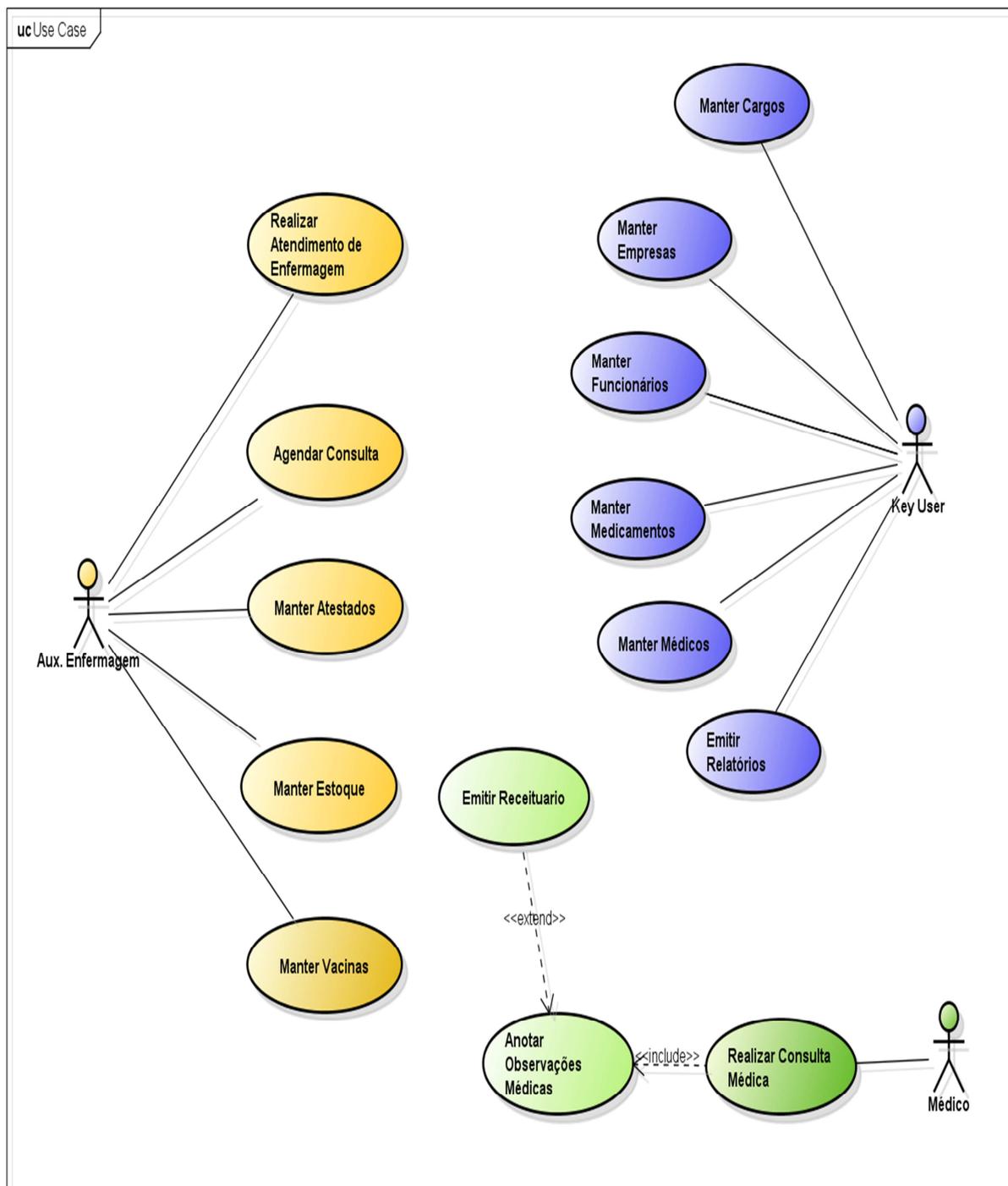
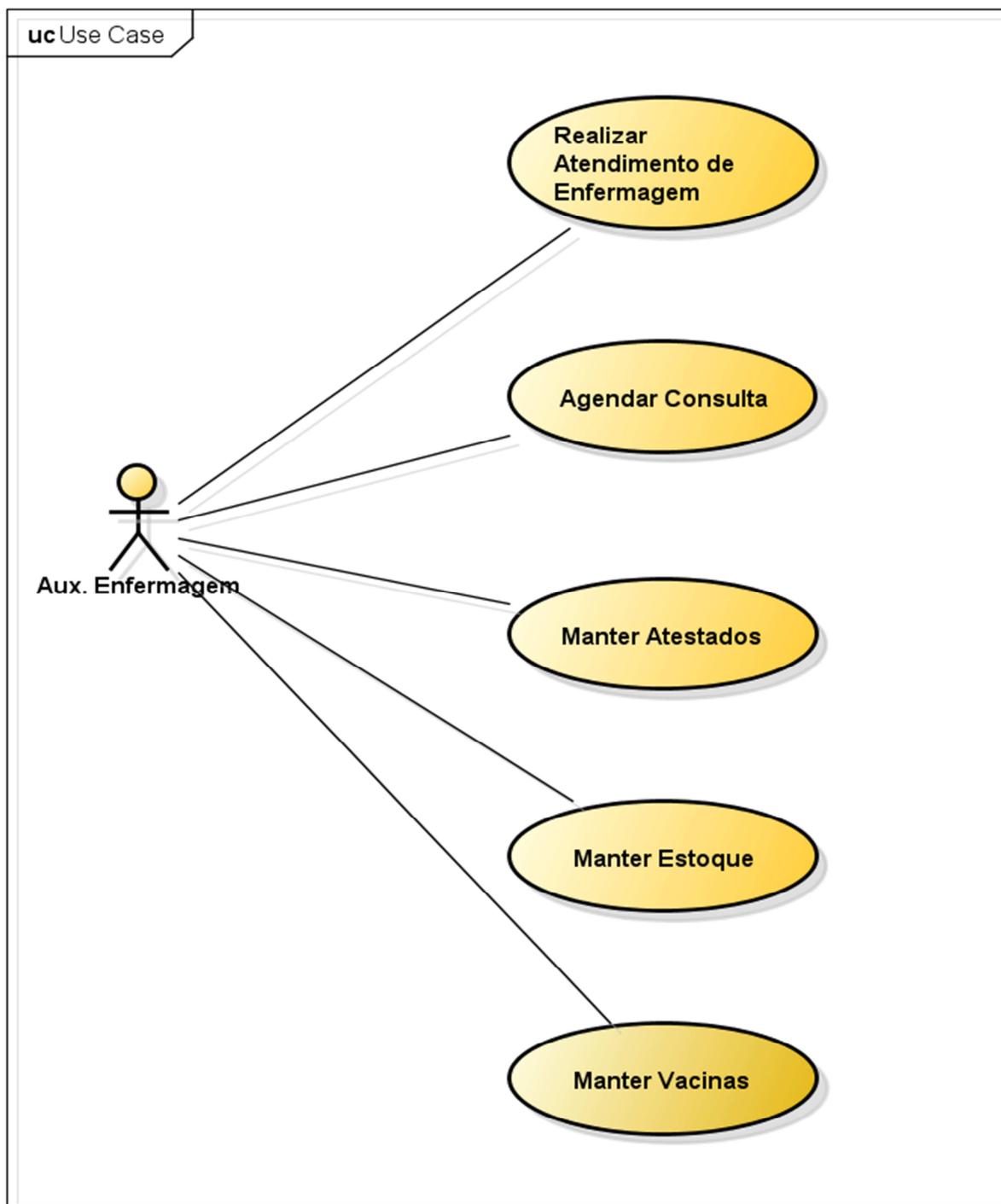


Figura 3 - Caso de Uso: Geral

5.2.2. Auxiliar de Enfermagem: Geral

A figura 4 ilustra o caso de uso Auxiliar de Enfermagem Geral do sistema AmbSystem.

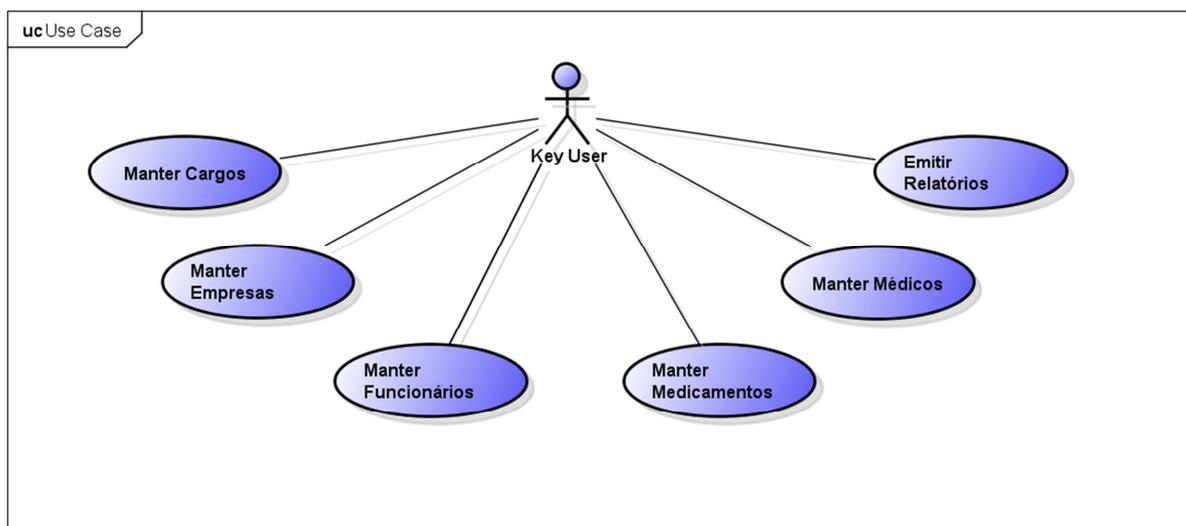


powered by Astah

Figura 4 - Caso de Uso: Auxiliar de Enfermagem (Geral)

5.2.3. Key User: Geral

A figura abaixo ilustra as iterações do key user com o sistema e as ações que deverão ser tomadas na implantação e decorrer do uso da aplicação.

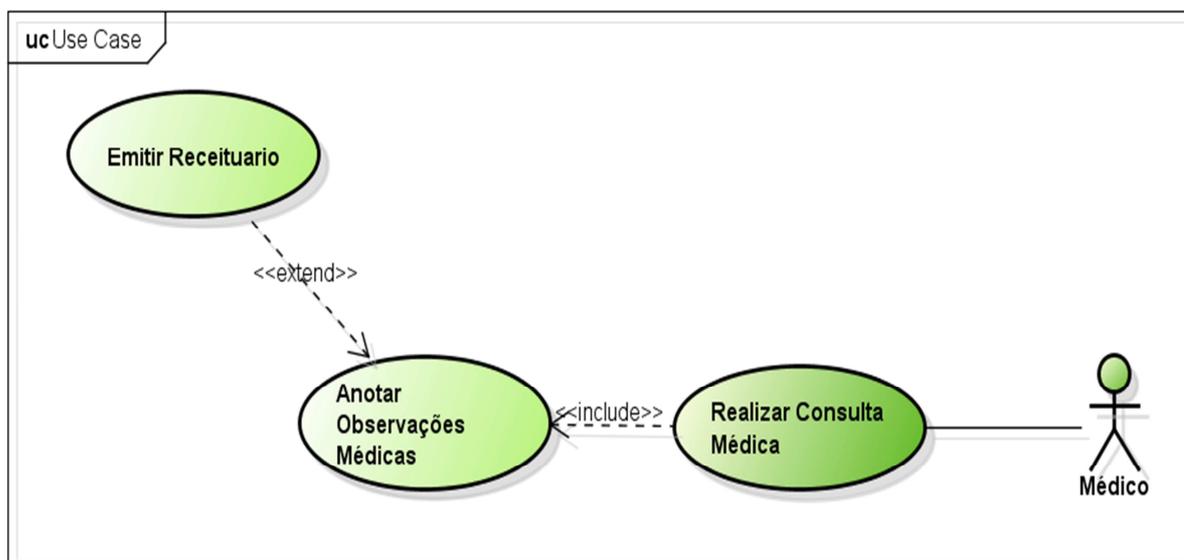


powered by Astah

Figura 5 - Caso de Uso: Key User (Geral)

5.2.4. Médico: Geral

A figura abaixo ilustra as iterações do Médico com o sistema e as ações que deverão ser tomadas na implantação e decorrer do uso da aplicação.



powered by Astah

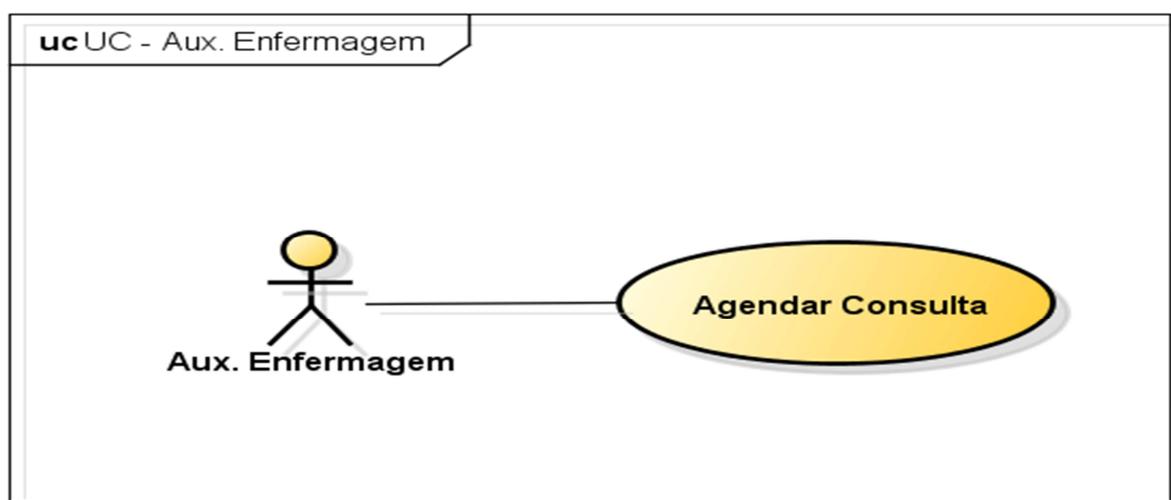
Figura 6 - Caso de Uso: Médico (Geral)

5.3. ESPECIFICAÇÃO DOS CASOS DE USO

A seguir são apresentadas as especificações dos casos de uso, identificados a partir do levantamento de requisitos, funcionalidades e relações existentes entre sistema e atores.

5.3.1. Agendar Consulta

A figura 7 ilustra o caso de uso Agendar Consulta.



powered by Astah

Figura 7 - Caso de Uso: Agendar Consulta

Finalidade/Objetivo	Realizar Agendamento de Consulta
Atores	Aux. de Enfermagem
Pré-Condição	Ter realizado o login no sistema
Fluxo Principal	<ul style="list-style-type: none"> a) O ator inicia o caso de uso selecionando a opção "Agenda Médica"; b) O sistema oferece a interface de agendamento médica; c) O Ator realiza no sistema a busca de um funcionário [A1]; d) O sistema exibe as informações do funcionário; e) O Ator seleciona o tipo de atendimento [A2]; f) O sistema exibe a descrição do tipo do atendimento; g) O Ator confirma a operação clicando em "salvar"; h) O sistema exibe a mensagem de confirmação; i) O UC é encerrado.
Fluxo Alternativo	A1 - Consultar Funcionário: <ul style="list-style-type: none"> a) O Ator informa o nome de um funcionário ou o numero de sua matricula;

	b) O sistema exibe uma listagem dos funcionários; c) O Ator seleciona um funcionário; d) Volta ao passo (d) do fluxo principal.
	A2 - Tipo de Atendimento: a) O Ator seleciona a opção “tipo de atendimento”; b) O sistema exibe uma listagem dos tipos de atendimentos; c) O Ator seleciona um tipo; d) Volta ao passo (f) do fluxo principal.

Tabela 2 - Agendar Consulta

5.3.2. Manter Atestados

A figura 8 ilustra o caso de uso Manter Atestados.

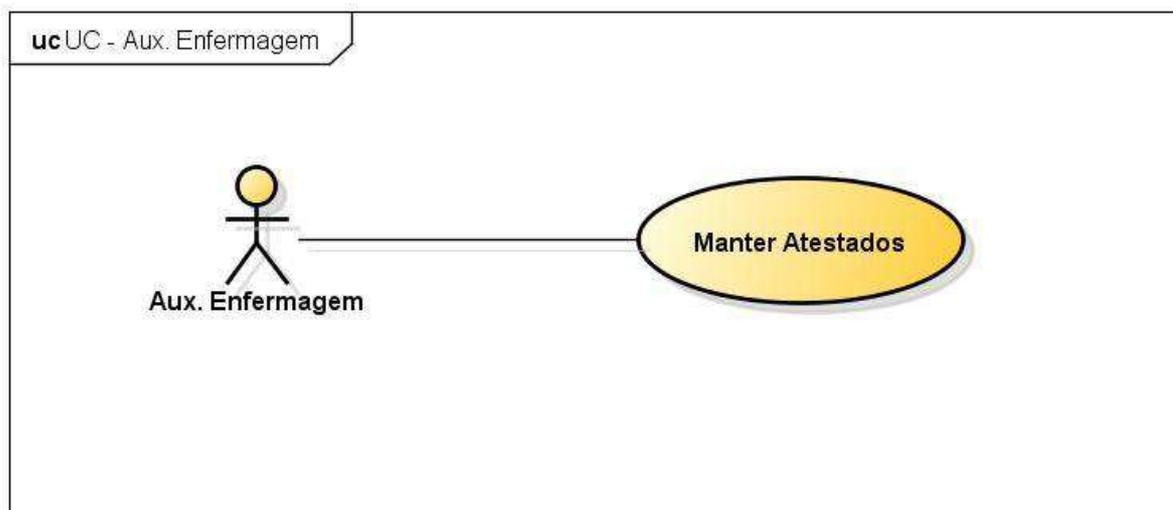


Figura 8 - Caso de Uso: Manter Atestados

Finalidade/Objetivo	Realizar Cadastramento de Atestados.
Atores	Aux. de Enfermagem
Pré-Condição	Ter realizado o login no sistema
Fluxo Principal	<ul style="list-style-type: none"> a) O ator inicia o caso de uso selecionando a opção "Atestado médico"; b) O sistema oferece a interface de Atestado médico; c) O Ator realiza no sistema a busca de um funcionário [A1]; d) O sistema exibe as informações do funcionário; e) O Ator inseri a data inicial e data fim do atestado; f) O sistema exibe o período de dias presentes no atestado; g) O Ator realiza no sistema a busca de um médico [A2], [E1]; h) O sistema exibe as informações sobre o médico; i) O Ator digita o CID existente no atestado; j) O sistema exibe as informações sobre o CID; k) O Ator seleciona a opção "salvar"; l) O sistema solicita confirmação sobre o evento; m) O Ator confirma a operação clicando em "sim"; n) O sistema exibe a mensagem de confirmação; o) O UC é encerrado.
Fluxo Alternativo	<p>A1 - Consultar Funcionário:</p> <ul style="list-style-type: none"> a) O Ator informa o nome de um funcionário ou o número de sua matrícula; b) O sistema exibe uma listagem dos funcionários; c) O Ator seleciona um funcionário;

	d) Volta ao passo (d) do fluxo principal.
	<p>A2 - Consultar Médico:</p> <p>a) O Ator seleciona a opção “médico”;</p> <p>b) O sistema exibe uma listagem dos médicos existentes;</p> <p>c) O Ator seleciona um médico;</p> <p>d) Volta ao passo (h) do fluxo principal.</p>
Fluxo de Exceção	<p>E1 - Médico não existe o cadastro:</p> <p>a) Ator realiza solicitação ao Key User responsável, que o médico seja cadastrado;</p> <p>b) Volta ao passo (h) do fluxo principal.</p>

Tabela 3 - Manter Atestados

5.3.3. Manter Estoque

A figura 9 ilustra o caso de uso Manter Estoque.

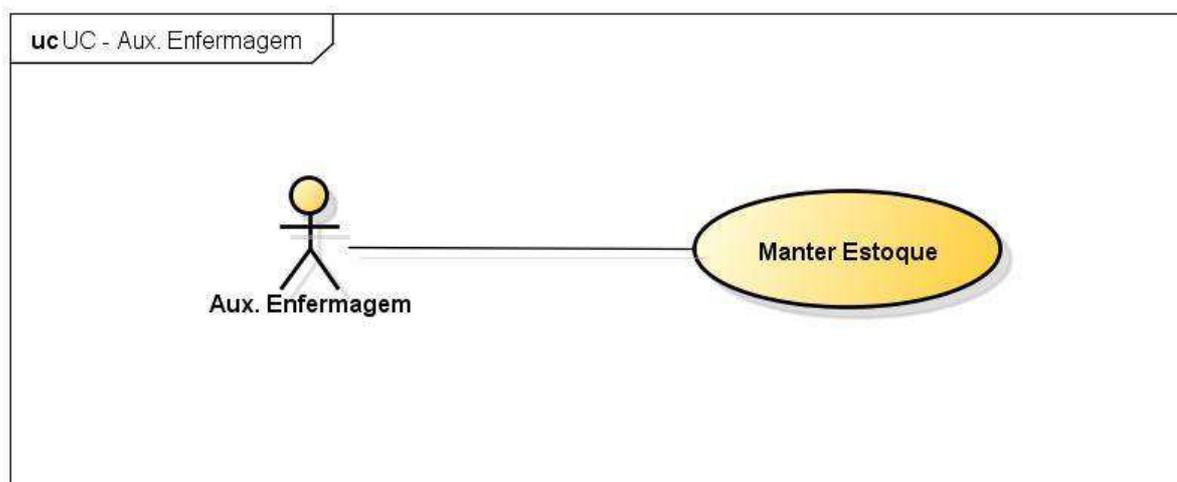


Figura 9 - Caso de Uso: Manter Estoque

Finalidade/Objetivo	Permitir ao Aux. de Enfermagem manter o estoque.
Atores	Aux. de Enfermagem
Pré-Condição	Ter realizado o login no sistema
Fluxo Principal	<ul style="list-style-type: none"> a) O ator inicia o caso de uso selecionando a opção "Controle de Estoque"; b) O sistema oferece a interface de Controle de Estoque; c) O Ator verifica a quantidade atual dos materiais existentes no sistema [A1] [A2]; d) O sistema exibe as informações da busca realizada; e) O UC é encerrado.
Fluxo Alternativo	<p>A1 - Consultar Material:</p> <ul style="list-style-type: none"> a) O Ator informa um material; b) O sistema exibe uma listagem dos materiais existentes em ordem alfabética; c) O Ator seleciona um material; d) Volta ao passo (d) do fluxo principal. <p>A2 - Alterar Estoque:</p> <ul style="list-style-type: none"> a) O Ator seleciona a opção "alterar"; b) O sistema exibe as informações sobre o material; c) O Ator altera as informações necessárias e confirma a alteração; d) O sistema confirma a alteração e exibe uma mensagem de confirmação;

Tabela 4 - Manter Estoque

5.3.4. Manter Vacinas

A figura 10 ilustra o caso de uso Manter Vacinas.

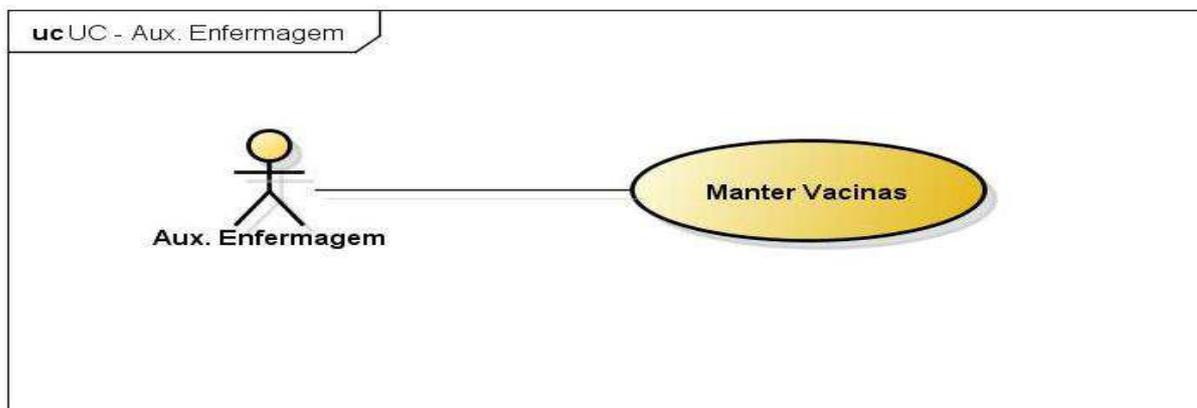


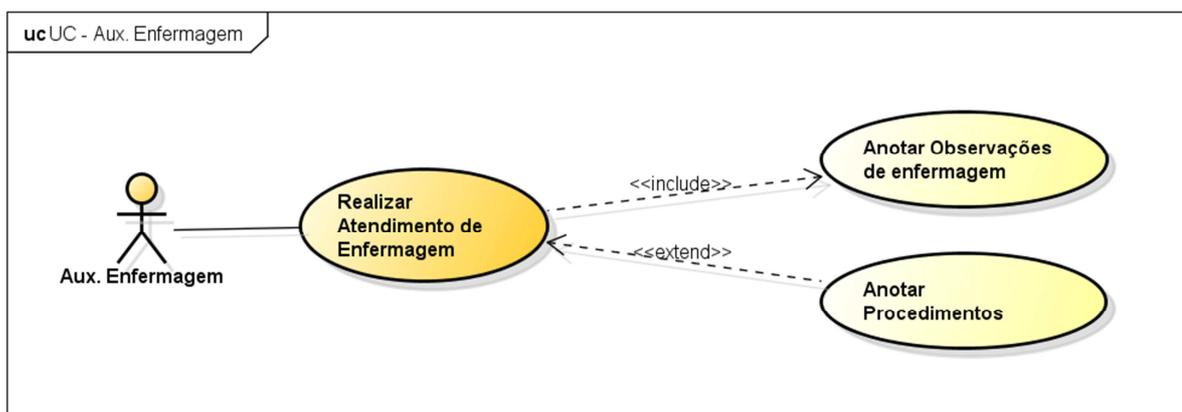
Figura 10 - Caso de Uso: Manter Vacinas

Finalidade/Objetivo	Realizar cadastramento de vacinas
Atores	Aux. de Enfermagem
Pré-Condição	Ter realizado o login no sistema
Fluxo Principal	<ul style="list-style-type: none"> a) O ator inicia o caso de uso selecionando a opção “Controle de Vacinas”; b) O sistema oferece a interface de Controle de Estoque; c) O Ator realiza no sistema a busca de um funcionário [A1]; d) O sistema exibe as informações do funcionário; e) O Ator seleciona o tipo de vacina [A2]; f) O sistema exibe a descrição do tipo do vacinas; g) O Ator insere as informações referentes a data da vacina; h) O sistema exibe as informações inseridas; i) O Ator confirma a operação clicando em “salvar”; j) O sistema exibe a mensagem de confirmação; l) O UC é encerrado.
Fluxo Alternativo	A1 - Consultar Funcionário: <ul style="list-style-type: none"> a) O Ator informa o nome de um funcionário ou o número de sua matrícula; b) O sistema exibe uma listagem dos funcionários; c) O Ator seleciona um funcionário; d) Volta ao passo (d) do fluxo principal.
	A2 - Tipo de Vacina: <ul style="list-style-type: none"> a) O Ator seleciona a opção “tipo de vacina”; b) O sistema exibe uma listagem dos tipos de vacinas; c) O Ator seleciona um tipo; d) Volta ao passo (f) do fluxo principal.

Tabela 5 - Manter Vacinas

5.3.5. Realizar Atendimento de Enfermagem

A figura 11 ilustra o caso de uso Realizar Atendimento de Enfermagem.



powered by Astah

Figura 11 - Caso de Uso: Realizar Atendimento de Enfermagem

Finalidade/Objetivo	Realizar Atendimento de Enfermagem.
Atores	Aux. de Enfermagem
Pré-Condição	Ter realizado o login no sistema
Fluxo Principal	<ul style="list-style-type: none"> a) O ator inicia o caso de uso selecionando a opção “Atendimento de enfermagem”; b) O sistema oferece a interface de atendimento de enfermagem; c) O Ator realiza no sistema a busca de um funcionário [A1]; d) O sistema exibe as informações do funcionário; e) O Ator digita as observações referentes ao atendimento [A2]; f) O sistema solicita confirmação sobre o evento; g) O Ator confirma a operação clicando em “sim”; h) O sistema exibe a mensagem de confirmação; i) O UC é encerrado.
Fluxo Alternativo	A1 - Consultar Funcionário: <ul style="list-style-type: none"> a) O Ator informa o nome de um funcionário ou o numero de sua matricula; b) O sistema exibe uma listagem dos funcionários; c) O Ator seleciona um funcionário; d) Volta ao passo (d) do fluxo principal.
	A2 - Procedimentos: <ul style="list-style-type: none"> a) O Ator seleciona a opção “procedimentos”; b) O sistema exibe uma listagem dos procedimentos; c) O Ator seleciona um procedimento; d) Volta ao passo (f) do fluxo principal.

Tabela 6 - Realizar Atendimento de Enfermagem

5.3.6. Manter Cargos

A figura 12 ilustra o caso de uso Manter Cargos.

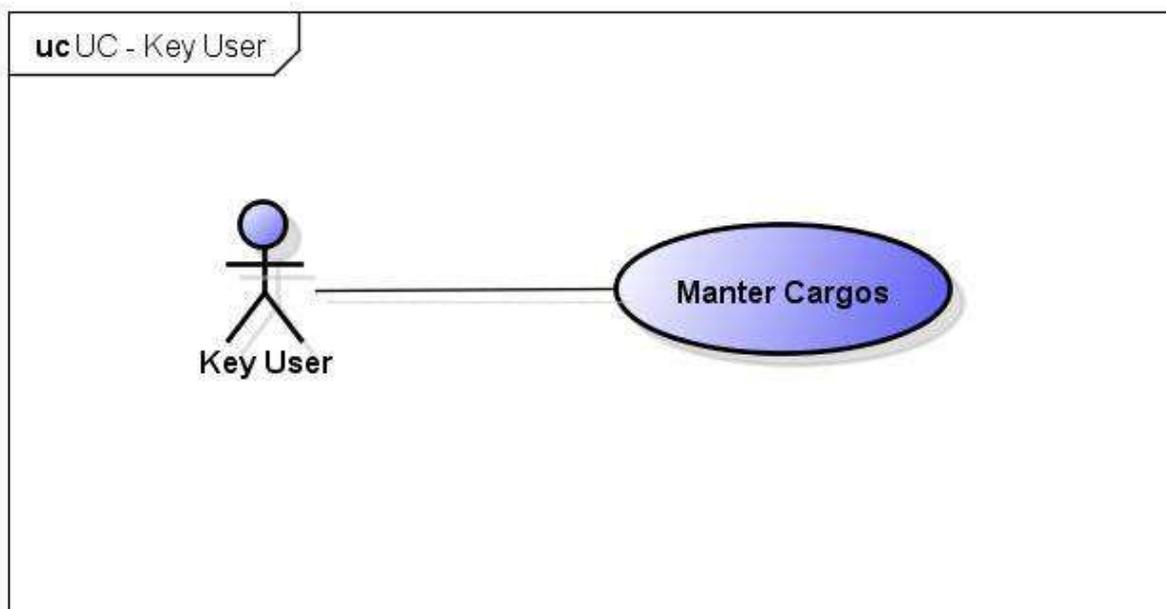


Figura 12 - Caso de Uso: Manter Cargos

Finalidade/Objetivo	Manter Cargos
Atores	Key User
Pré-Condição	Ter realizado o login no sistema
Fluxo Principal	<ul style="list-style-type: none"> a) O ator inicia o caso de uso selecionando a opção "Cargos"; b) O sistema oferece a interface de Cargos; c) O Ator realiza no sistema a busca de um cargo [A1]; d) O sistema exibe as informações do cargo; e) O UC é encerrado.
Fluxo Alternativo	<p>A1 - Consultar Cargos:</p> <ul style="list-style-type: none"> a) O Ator informa o nome de um cargo ou o código; b) O sistema exibe uma listagem de cargos; c) O Ator seleciona um cargo; d) Volta ao passo (d) do fluxo principal.

Tabela 7 - Manter Cargos

5.3.7. Manter Empresas

A figura 13 ilustra o caso de uso Manter Empresas.

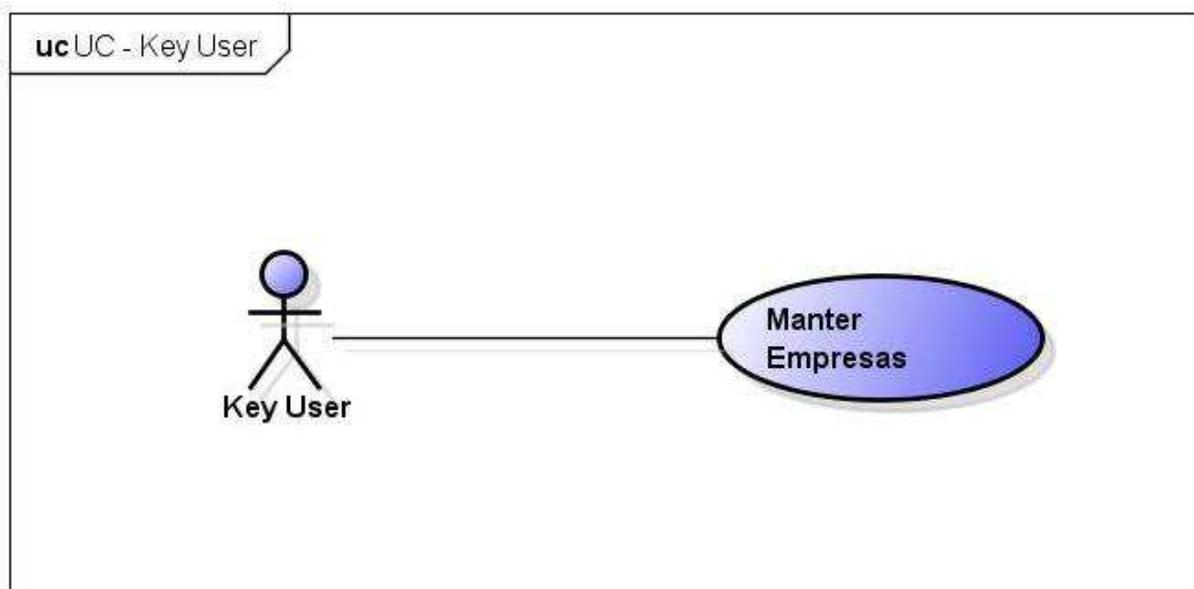


Figura 13 - Caso de Uso: Manter Empresas

Finalidade/Objetivo	Manter Empresas
Atores	Key User
Pré-Condição	Ter realizado o login no sistema
Fluxo Principal	<ul style="list-style-type: none"> a) O ator inicia o caso de uso selecionando a opção “Empresas”; b) O sistema oferece a interface de Empresas; c) O Ator realiza no sistema a busca de uma empresa [A1]; d) O sistema exibe as informações da empresa; e) O UC é encerrado.
Fluxo Alternativo	<p>A1 - Consultar Empresa:</p> <ul style="list-style-type: none"> a) O Ator informa o nome de uma empresas ou seu código; b) O sistema exibe uma listagem de empresas; c) O Ator seleciona uma empresa; d) Volta ao passo (d) do fluxo principal.

Tabela 8 - Manter Empresas

5.3.8. Manter Funcionários

A figura 14 ilustra o caso de uso Manter Funcionários.

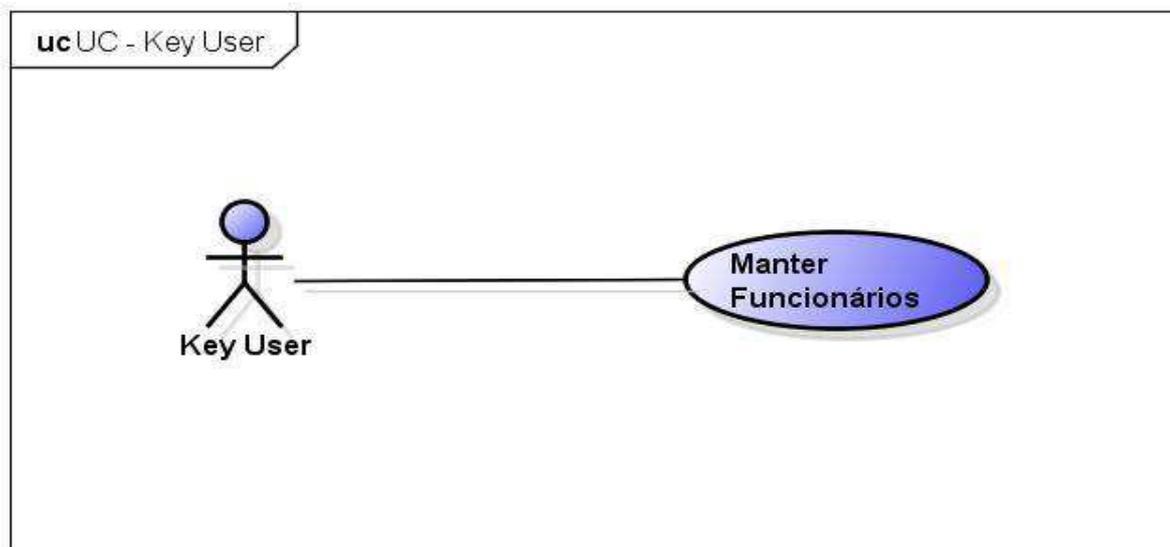


Figura 14 - Caso de Uso: Manter Funcionários

Finalidade/Objetivo	Manter Funcionários
Atores	Key User
Pré-Condição	Ter realizado o login no sistema
Fluxo Principal	<ul style="list-style-type: none"> a) O ator inicia o caso de uso selecionando a opção "Funcionários"; b) O sistema oferece a interface de Funcionários; c) O Ator realiza no sistema a busca de um funcionário [A1]; d) O sistema exibe as informações do funcionário; e) O UC é encerrado.
Fluxo Alternativo	<p>A1 - Consultar Funcionário:</p> <ul style="list-style-type: none"> a) O Ator informa o nome de um funcionário ou o número de sua matrícula; b) O sistema exibe uma listagem dos funcionários; c) O Ator seleciona um funcionário; d) Volta ao passo (d) do fluxo principal.

Tabela 9 - Manter Funcionários

5.3.9. Manter Medicamentos

A figura 20 ilustra o caso de uso Manter Medicamentos.

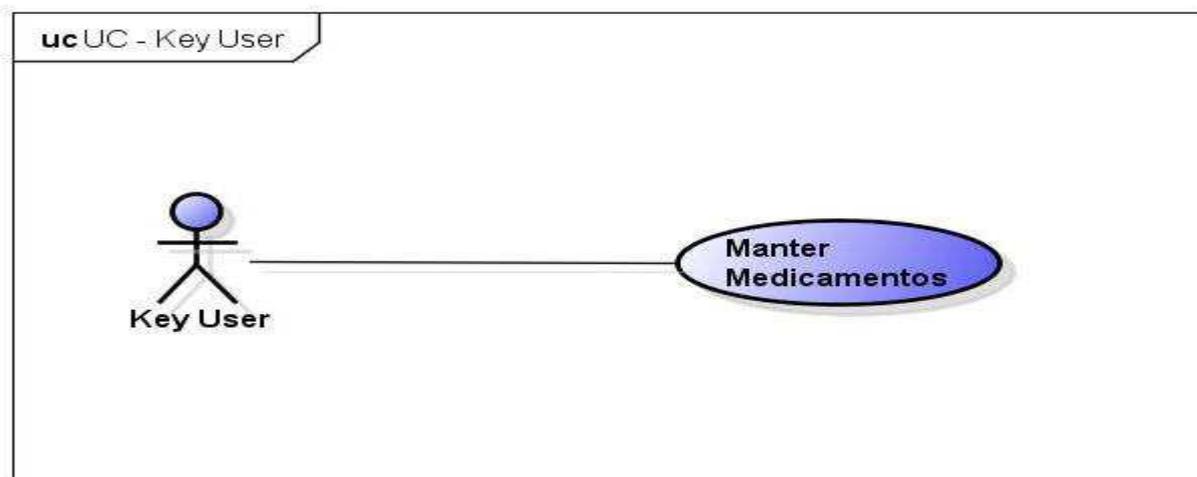


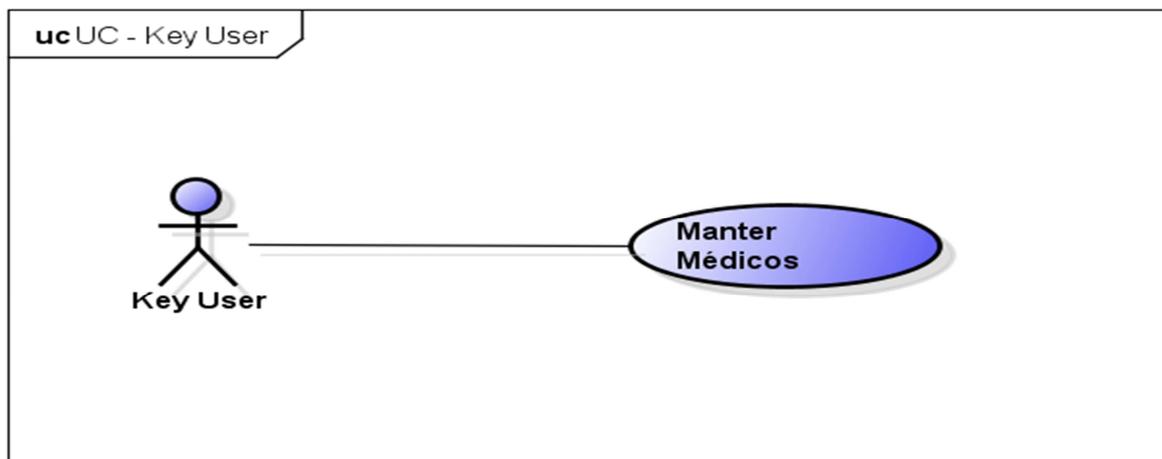
Figura 15 - Caso de Uso: Manter Medicamentos

Finalidade/Objetivo	Manter Medicamentos
Atores	Key User
Pré-Condição	Ter realizado o login no sistema
Fluxo Principal	<ul style="list-style-type: none"> a) O ator inicia o caso de uso selecionando a opção "Medicamentos"; b) O sistema oferece a interface de Medicamentos; c) O Ator realiza no sistema a busca de um medicamento [A1]; d) O sistema exibe as informações do medicamento; e) O Ator seleciona o medicamento [A2]; f) O UC é encerrado.
Fluxo Alternativo	<p>A1 - Consultar Medicamento:</p> <ul style="list-style-type: none"> a) O Ator informa o nome de um medicamento ou seu código; b) O sistema exibe uma listagem dos medicamentos [E1]; c) O Ator seleciona um medicamento; d) Volta ao passo (d) do fluxo principal.
Fluxo de Exceção	<p>E1 - Medicamento não existe no cadastro:</p> <ul style="list-style-type: none"> a) O Ator seleciona a opção novo medicamento; b) O sistema oferece a interface de novo medicamento; c) O Ator digita os dados referentes ao novo medicamento; d) O sistema exibe as informações do novo medicamento; e) O Ator confirma a operação clicando em "salvar" f) Volta ao passo (c) do fluxo alternativo.

Tabela 10 - Manter Medicamentos

5.3.10. Manter Médicos

A figura 16 ilustra o caso de uso Manter Médicos.



powered by Astah

Figura 16 - Caso de Uso: Manter Médicos

Finalidade/Objetivo	Manter Médicos
Atores	Key User
Pré-Condição	Ter realizado o login no sistema
Fluxo Principal	<ul style="list-style-type: none"> a) O ator inicia o caso de uso selecionando a opção "Profissionais"; b) O sistema oferece a interface de Profissionais; c) O Ator realiza no sistema a busca de um profissional [A1]; d) O sistema exibe as informações do profissional; e) O Ator seleciona o profissional; f) O UC é encerrado.
Fluxo Alternativo	<p>A1 - Consultar Profissional:</p> <ul style="list-style-type: none"> a) O Ator informa o nome de um profissional ou seu código; b) O sistema exibe uma listagem de profissionais [E1]; c) O Ator seleciona um profissional; d) Volta ao passo (d) do fluxo principal.
Fluxo de Exceção	<p>E1 - Medicamento não existe no cadastro:</p> <ul style="list-style-type: none"> a) O Ator seleciona a opção novo profissional; b) O sistema oferece a interface de novo profissional; c) O Ator digita os dados referentes ao novo profissional; d) O sistema exibe as informações do novo profissional; e) O Ator confirma a operação clicando em "salvar" f) Volta ao passo (c) do fluxo alternativo.

Tabela 11 - Manter Médicos

5.3.11. Emitir Relatórios

A figura 17 ilustra o caso de uso Manter Relatórios.

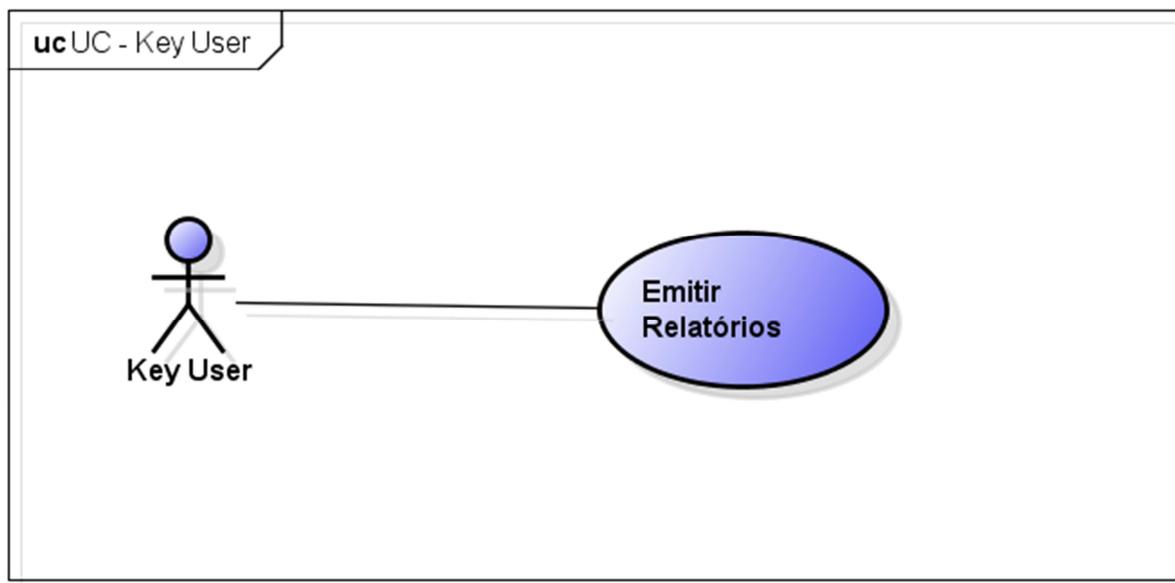


Figura 17 - Caso de Uso: Manter Relatórios

Finalidade/Objetivo	Realizar emissão de relatórios
Atores	Aux. de Enfermagem
Pré-Condição	Ter realizado o login no sistema
Fluxo Principal	<ul style="list-style-type: none"> a) O ator inicia o caso de uso selecionando a opção "Relatórios"; b) O sistema oferece a interface de emissão de relatórios; c) O Ator seleciona o tipo de relatório [A1]; d) O sistema exibe as informações do relatório; e) O UC é encerrado.
Fluxo Alternativo	<p>A1 - Atendimentos:</p> <ul style="list-style-type: none"> a) O Ator informa as opções de filtro; b) O sistema informa uma listagem conforme os filtros selecionados; c) O Ator realiza a impressão do relatório; d) Volta ao passo (d) do fluxo principal. <p>A2 - Atestados:</p> <ul style="list-style-type: none"> a) O Ator informa as opções de filtro; b) O sistema informa uma listagem conforme os filtros selecionados; c) O Ator realiza a impressão do relatório; d) Volta ao passo (d) do fluxo principal.

	A3 - Estoque: <ul style="list-style-type: none">a) O Ator informa as opções de filtro;b) O sistema informa uma listagem conforme os filtros selecionados;c) O Ator realiza a impressão do relatório;d) Volta ao passo (d) do fluxo principal.
	A4 - Funcionários: <ul style="list-style-type: none">a) O Ator informa as opções de filtro;b) O sistema informa uma listagem conforme os filtros selecionados;c) O Ator realiza a impressão do relatório;d) Volta ao passo (d) do fluxo principal.

Tabela 12 - Manter Relatórios

5.3.12. Realizar Consulta Médica

A figura 18 ilustra o caso de uso Realizar Consulta Médica.

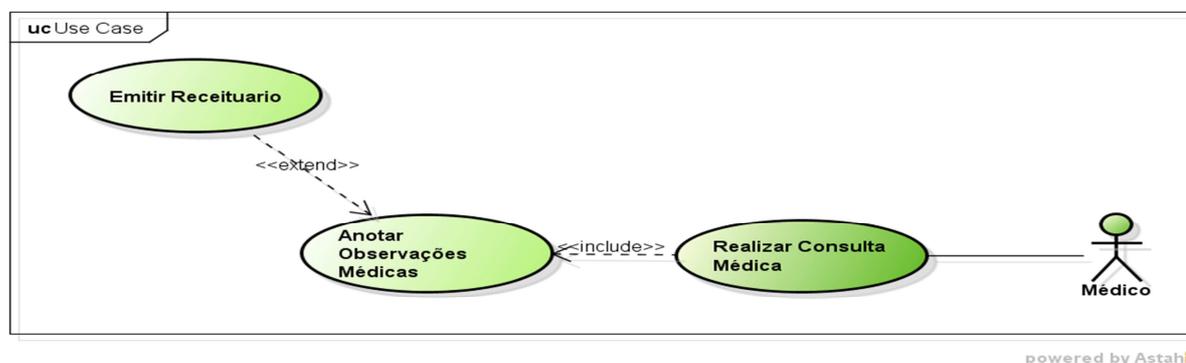


Figura 18 - Caso de Uso: Realizar Consulta Médica

Finalidade/Objetivo	Realizar Consulta médica
Atores	Médico
Pré-Condição	Ter realizado o login no sistema
Fluxo Principal	<ul style="list-style-type: none"> a) O ator inicia o caso de uso selecionando a opção “Consulta Médica”; b) O sistema oferece a interface de consulta médica; c) O Ator seleciona o funcionário já agendado; d) O sistema exibe as informações do funcionário; e) O Ator seleciona a opção “ficha médica”; f) O sistema oferece a interface de ficha médica; g) O Ator realiza a digitação das observações coletas durante o atendimento; h) O sistema exibe as informações digitadas; i) O Ator confirma a operação clicando em “salvar” [A1]; j) O sistema exibe a mensagem de confirmação; k) O UC é encerrado.
Fluxo Alternativo	A1 - Emitir receitaário: <ul style="list-style-type: none"> a) O Ator realiza no sistema a busca pelo medicamento [E1]; b) O sistema exibe uma listagem dos medicamentos existentes; c) O Ator seleciona um medicamento; d) Volta ao passo (j) do fluxo principal.
Fluxo de Exceção	E1 - Médico não existe o cadastro: <ul style="list-style-type: none"> a) Ator realiza solicitação ao Key User responsável, que o médico seja cadastrado; b) Volta ao passo (b) do fluxo alternativo.

Tabela 13 - Realizar Consulta Médica

6. DIAGRAMA DE CLASSE

É uma representação da estrutura que define todas as classes que o sistema necessita possuir e suas relações.

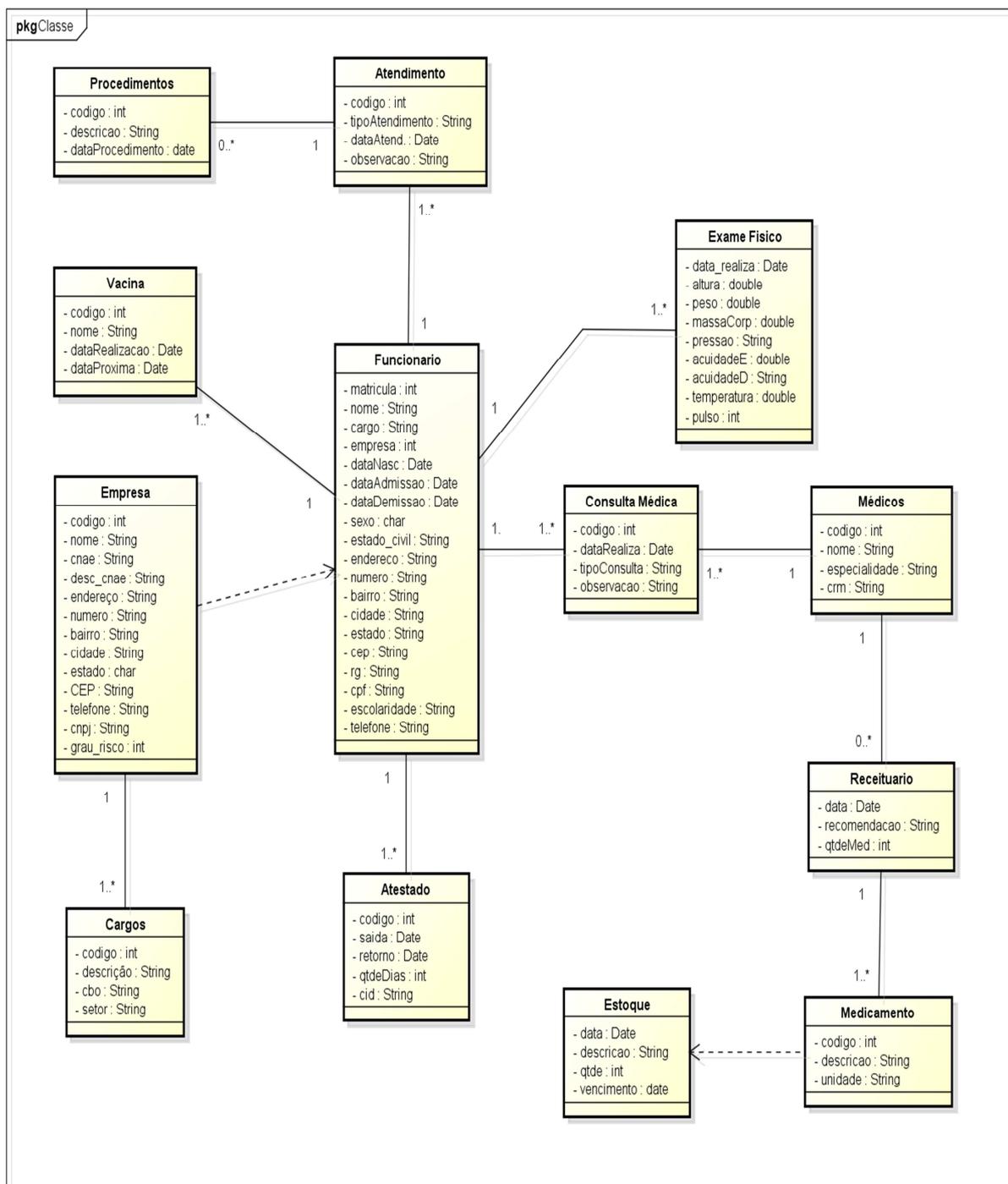
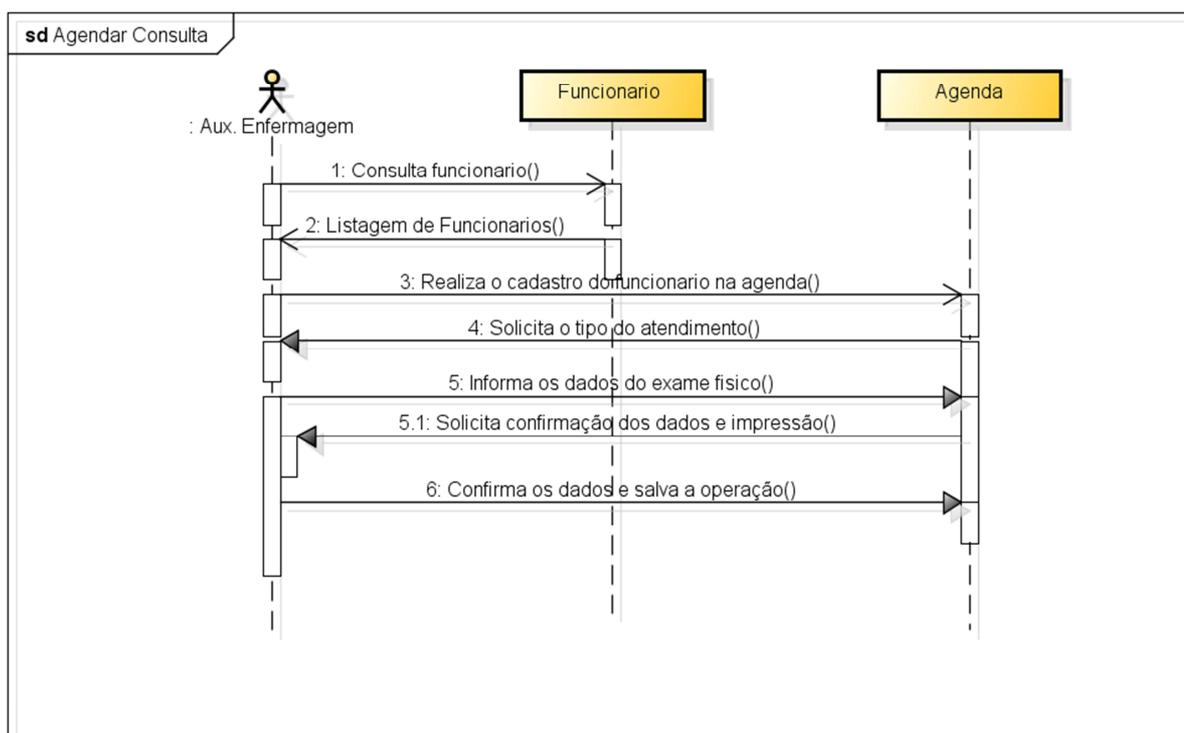


Figura 19 - Diagrama de Classes

7. DIAGRAMA DE SEQUÊNCIA

7.1. Agendar Consulta

A figura 20 ilustra o diagrama de sequência Agendar Consulta

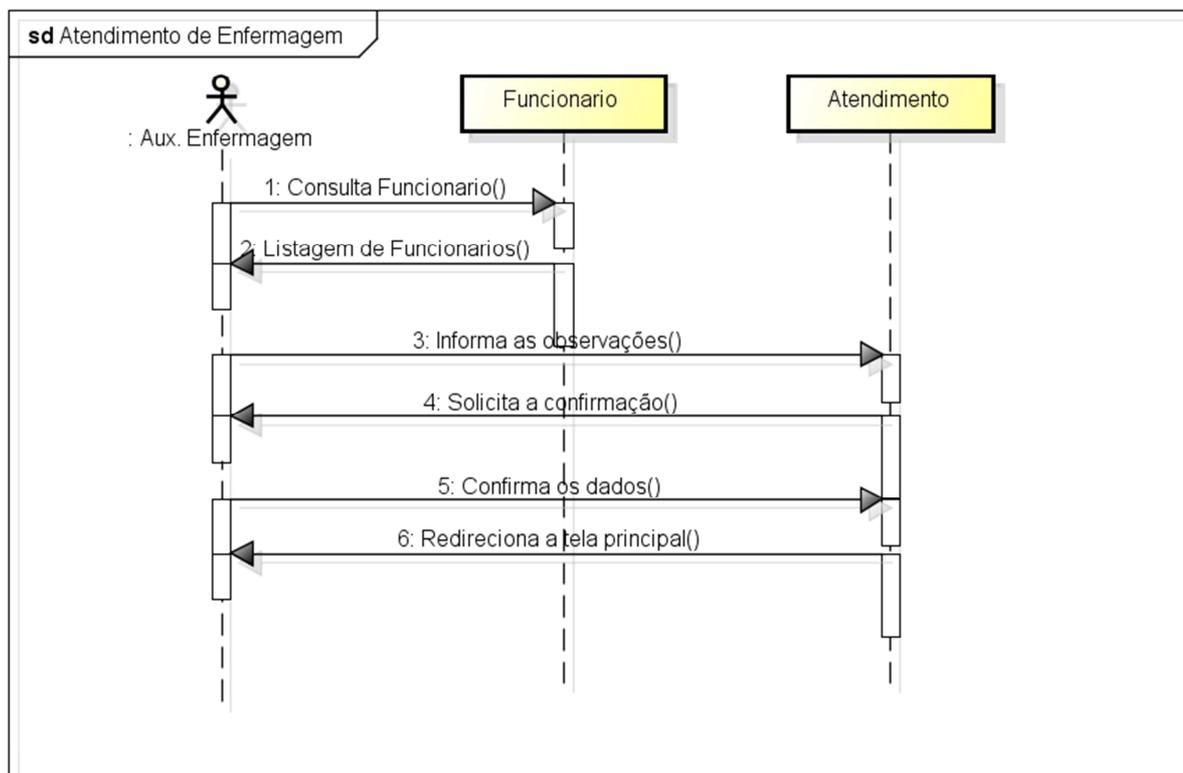


powered by Astah

Figura 20 - Agendar Consulta

7.2. Atendimento de Enfermagem

A figura 21 ilustra o diagrama de sequência Atendimento de Enfermagem



powered by Astah

Figura 21 - Atendimento de Enfermagem

7.3. Manter Atestados

A figura 26 ilustra o diagrama de seqüência Manter Atestados.

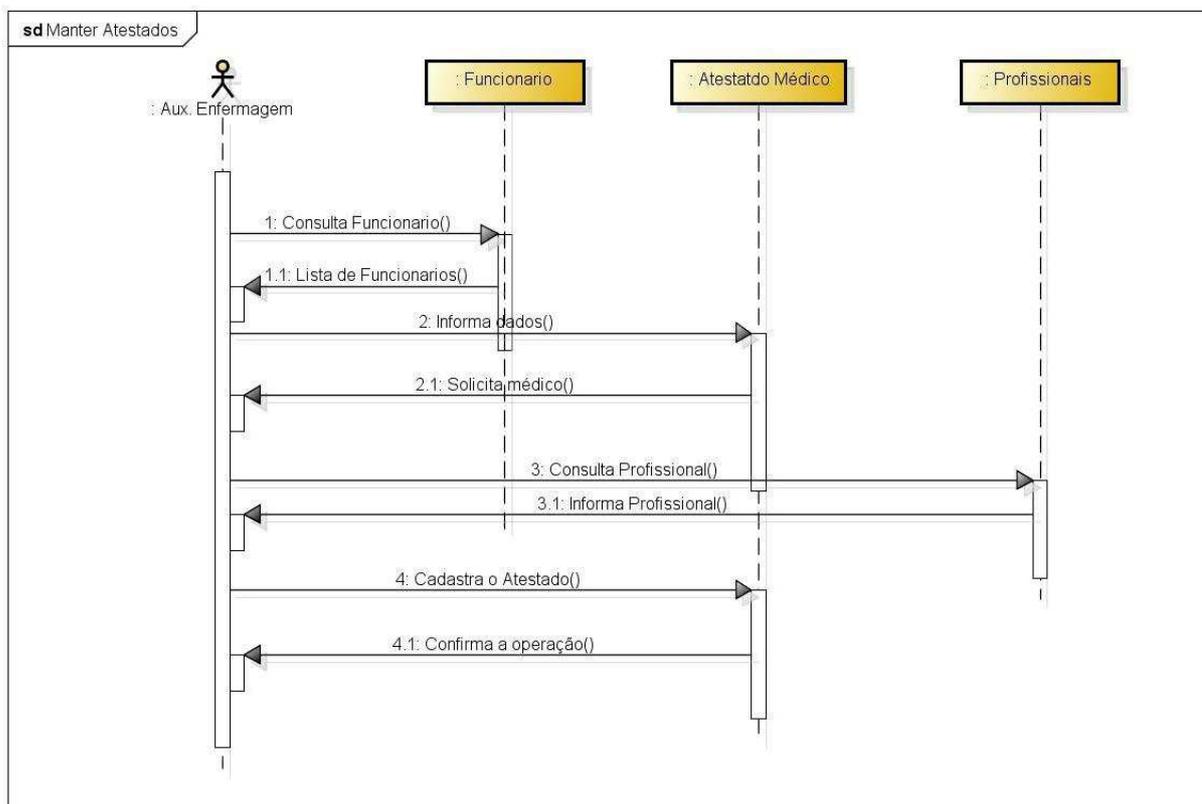
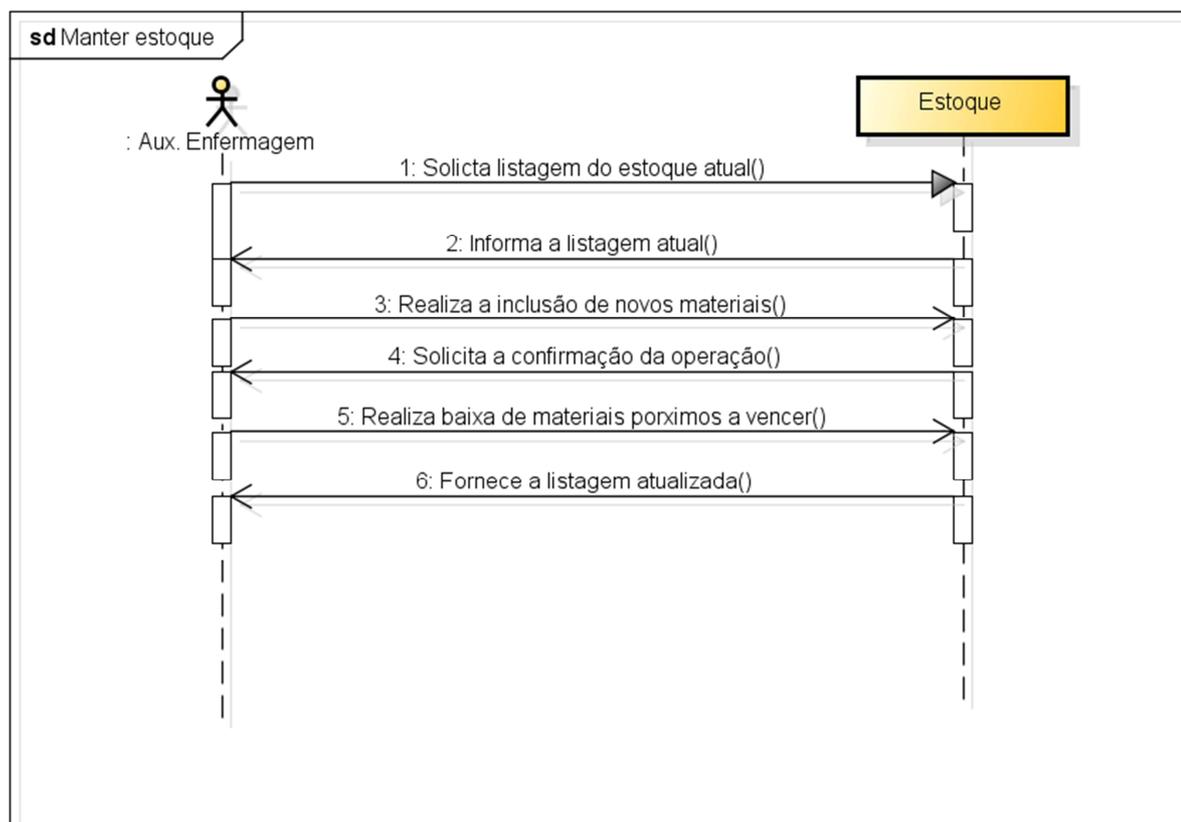


Figura 22 - Diagrama de Sequência: Manter Atestados

7.4. Manter Estoque

A figura 23 ilustra o diagrama de Manter Estoque.

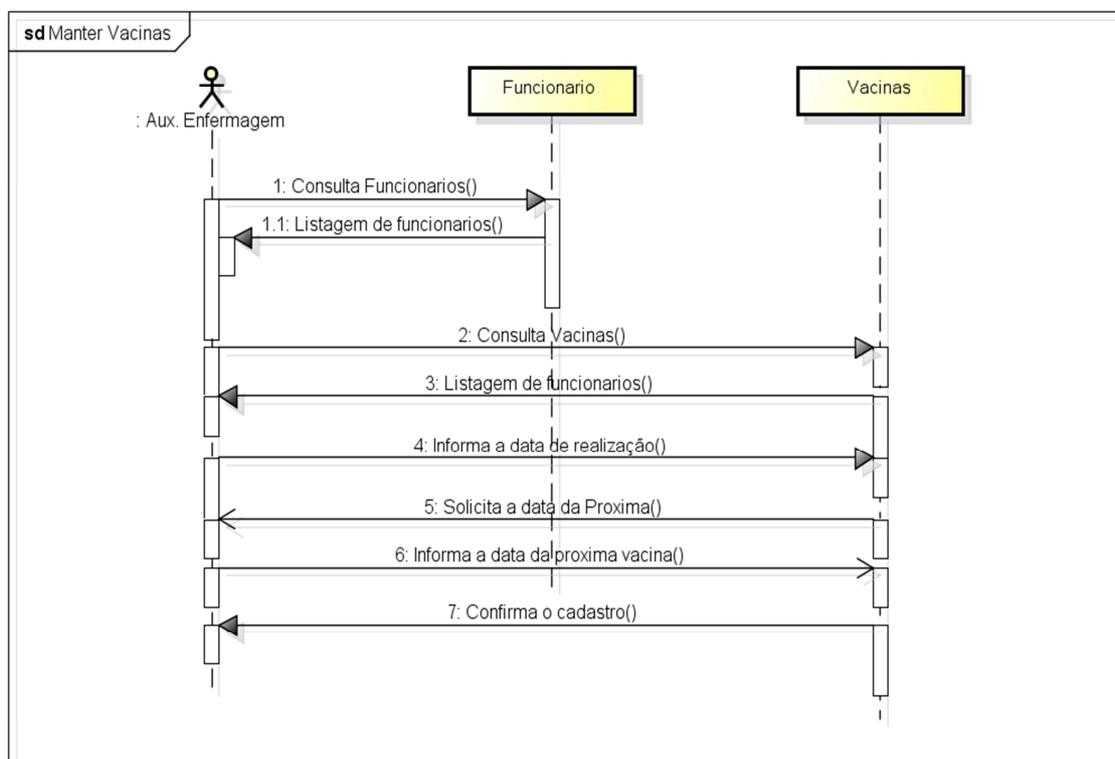


powered by Astah

Figura 23 - Diagrama de Sequência: Manter Estoque

7.5. Manter Vacinas

A figura 24 ilustra o diagrama de sequência Manter Vacinas.



powered by Astah

Figura 24 - Manter Vacinas

7.6. Emitir Receitaúrio

A figura 25 ilustra o diagrama de seqüência Emitir Receitaúrio.

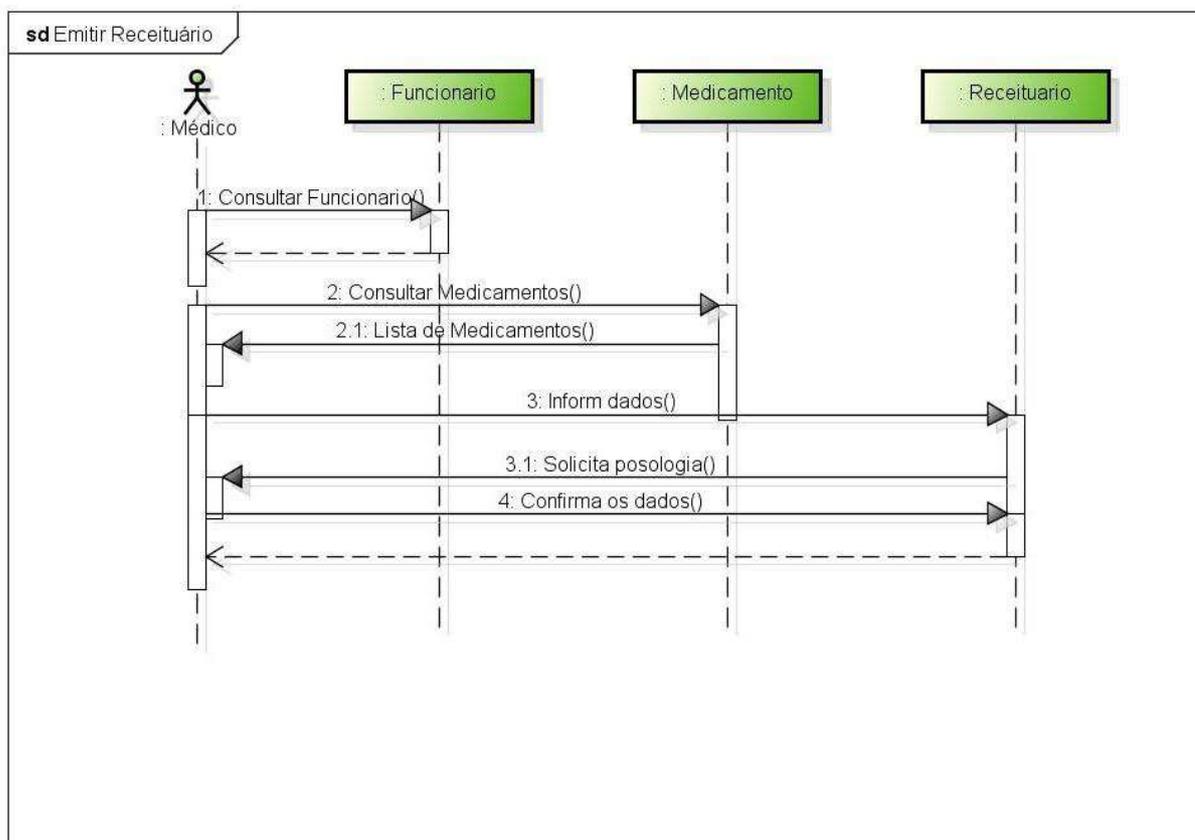


Figura 25 - Diagrama de Sequência: Emitir Receitaúrio

8. DIAGRAMA ER

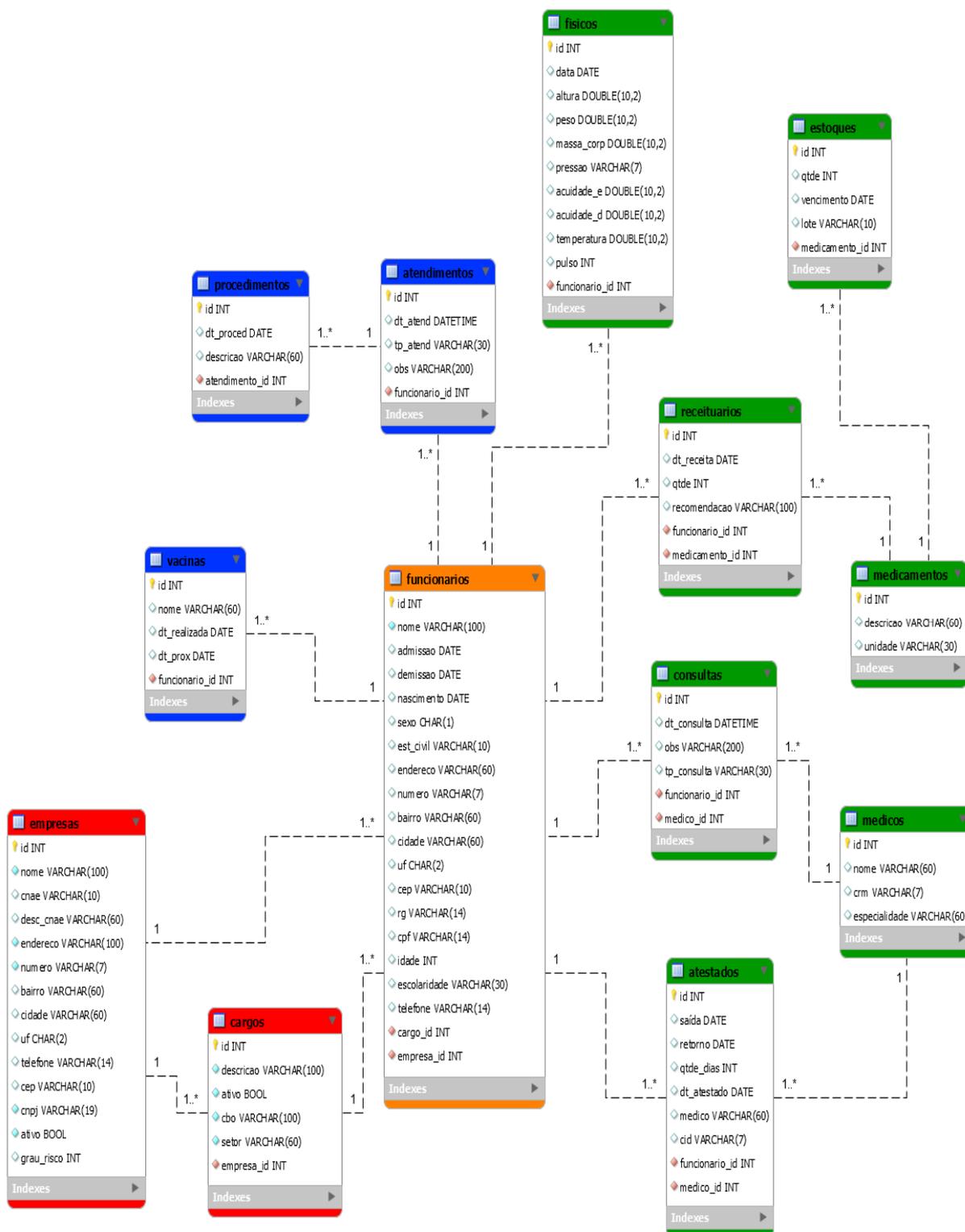


Figura 26 - Diagrama ER

9. CONCLUSÃO

Este trabalho de conclusão de curso foi desenvolvido com o intuito de oferecer agilidade e segurança no gerenciamento de informações geradas no dia a dia de trabalho de um ambulatório ocupacional, auxiliando em qualquer tipo de tomada de decisões, e apresentando relatórios para várias informações importantes sobre dados da empresa.

Através de pesquisas e conversas com pessoas que atuam no ambiente de um ambulatório foi possível identificar pontos que necessitavam serem desenvolvidos para que assim eu desenvolvesse um sistema no qual contém informações que satisfazem a necessidade de futuros usuários desse sistema.

A utilização dos diagramas da UML foram de extrema importância, pois eles proporcionaram um melhor entendimento dos relacionamentos e comportamentos dos atributos e métodos de cada classe representada no banco de dados, facilitando no desenvolver desta aplicação.

A implementação deste sistema contribuiu para um melhor aprendizado da tecnologia. PHP juntamente ao framework Laravel, além de ampliar meus conhecimentos em UML e manipulação de bancos de dados, gerando satisfação e êxito no desenvolvimento de aplicações Web.

10. REFERÊNCIAS

ALECRIM, Emerson. **Conhecendo o Servidor Apache (HTTP Server Project)**: Disponível em: <<http://www.infowester.com/servapach.php>>, Acesso em 08/08/2015.

GUERRA, Eduardo. **HTML5 e CSS3 Domine a Web do Futuro- Francisco Moraes**, 1º ed. São Paulo: Editora Casa do Código, 2014.

História do PHP: Disponível em:

<http://php.net/manual/pt_BR/history.php.php>, Acesso em: 08/08/2015.

J. BAPTISTELLA, Adriano. **Abordando a arquitetura MVC, e Design Patterns: Observer, Composite, Strategy**. Disponível em: <<http://www.linhadecodigo.com.br/artigo/2367/abordando-a-arquitetura-mvc-e-design-patterns-observer-composite-strategy.aspx#ixzz3TFNwmqgh>>. Acesso em: 26/02/2015.

NUNES, O'NEILL; Mauro, Henrique. **Fundamental de UML**. 2ª. ed. Lisboa: Editora FCA-EDITORIA DE INFORMÁTICA, 1998.

PINTO, Pedro. **Sublime Text – É este o melhor editor de texto do mundo?**: Disponível em: <<http://pplware.sapo.pt/software/sublime-text-e-este-o-melhor-editor-de-texto-do-mundo/>>, Acessado em: 08/08/2015.

PISA, Pedro. **O que é e como usar o MySQL?**: Disponível em: <<http://www.techtudo.com.br/artigos/noticia/2012/04/o-que-e-e-como-usar-o-mysql.html>>, Acessado em 08/08/2015.

PMI, Project Management Institute. Um Guia do Conhecimento em Gerenciamento de Projetos (**GUIA PMBOK**). 5ª ed. Filadélfia, 2013.

PRESSMAN, Roger S. **Engenharia de software**. 6ª ed. Porto Alegre: Bookman, 2006.

S. FILHO, Antonio Mendes. **Introdução à Programação Orientada a Objetos**, 1º ed. Rio de Janeiro: Editora Campus, 2010.

SILVA, Sara. **Você conhece o Laravel? Pois deveria...:** Disponível em: <<http://www.programabrasil.org/voce-conhece-o-laravel-pois-deveria/#.VcaFUiZViko>>, Acesso em: 08/08/2015.

TURINI, Rodrigo. **PHP e Laravel Crie aplicações web como um verdadeiro artesão**, 1º ed. São Paulo: Editora Casa do Código, 2015.