

**LIGA DE ENSINO DO RIO GRANDE DO NORTE
FACULDADE NATALENSE PARA O DESENVOLVIMENTO DO RN
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO**

**IMPLANTAÇÃO DE UM SISTEMA DE
INFORMAÇÃO WEB PARA BANCO DE SANGUE**

Erikson Girley Andrade de Lucena

Monografia apresentada ao curso de Bacharelado em Sistemas de Informação da Faculdade Natalense para o Desenvolvimento do Rio Grande do Norte como requisito parcial para a obtenção do grau em Bacharelado em Sistemas de Informação.

Orientador:

Felipe Araújo Albuquerque, MSc.

NATAL/RN

2005

ERIKSON GIRLEY ANDRADE DE LUCENA

**FACULDADE NATALENSE PARA O DESENVOLVIMENTO
DO RIO GRANDE DO NORTE
CURSO DE SISTEMA DE INFORMAÇÃO**

**IMPLANTAÇÃO DE UM SISTEMA DE
INFORMAÇÃO WEB PARA BANCO DE SANGUE**

Erikson Girley Andrade de Lucena

**IMPLANTAÇÃO DE UM SISTEMA DE
INFORMAÇÃO WEB PARA BANCO DE SANGUE**

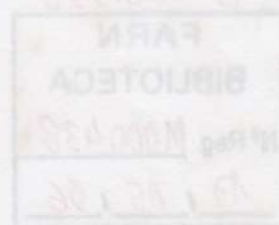
Monografia apresentada ao curso de Bacharelado em Sistemas de Informação da Faculdade Natalense para o Desenvolvimento do Rio Grande do Norte como requisito parcial para a obtenção do grau em Bacharel em Sistemas de Informação.

Orientador:

Fellipe Araújo Aleixo, MsC.

NATAL/RN

2005



Catalogação na Publicação – Biblioteca da FARN
Setor de Processos Técnicos

Lucena, Erikson Girley de
Implantação de um Sistema de Informação Web para Banco de Sangue / Erikson
Girley de Lucena. – Natal: FARN, 2005.
84 f.

Orientador: Fellipe Araújo Aleixo (M.Sc.)
Monografia (Bacharelado em Sistemas de Informação) – Faculdade Natalense
para o Desenvolvimento do Rio Grande do Norte.

1. Linguagem de Programação – Java – Monografia. 2. Internet – Serviço de
Saúde – Monografia. 3. Banco de Sangue – Sistema Web – Monografia. I. Título.

RN/FARN/BC

CDU 004.43

RB - 26.558

FARN
BIBLIOTECA
Nº Reg. M.000.438
17 / 05 / 06

M
004.43
L9681
2005

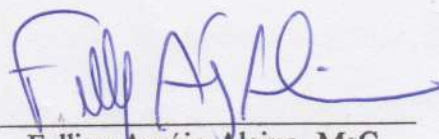
NATAL/RN
2005

**LIGA DE ENSINO DO RIO GRANDE DO NORTE
FACULDADE NATALENSE PARA O DESENVOLVIMENTO
DO RIO GRANDE DO NORTE
CURSO DE SISTEMA DE INFORMAÇÃO**

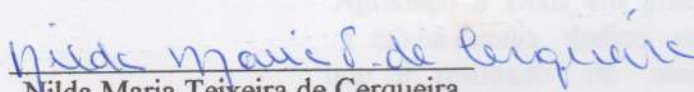
**IMPLANTAÇÃO DE UM SISTEMA DE
INFORMAÇÃO WEB PARA BANCO DE SANGUE**

Erikson Girley Andrade de Lucena

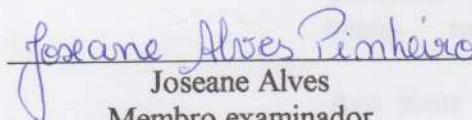
Monografia apresentada e aprovada em 17 de 12 de 2005 pela banca examinadora composta pelos seguintes membros.



Felipe Araújo Aleixo, MsC.
Orientador



Nilda Maria Teixeira de Cerqueira
Membro examinador



Joseane Alves
Membro examinador

Natal/RN, 17 de 12 de 2005.

AGRADECIMENTOS

Este trabalho foi construído com o apoio e incentivo de muitas pessoas e instituições. A elas venho agradecer com as mais sinceras palavras, esperando ter atingido os fins pretendidos neste empenho conjunto à atender as expectativas de todos que contribuíram para sua construção.

Assim, agradeço a todos os meus professores, pelos ensinamentos ministrados e pelo estímulo aos desafios profissionais e acadêmicos que deles recebi.

Aos meus professores revisores e orientadores, em especial o Prof. Felipe Alencar, que incondicionalmente me distinguiu com sua atenção e soube perceber os erros e acertos deste trabalho, fazendo-o sempre melhor.

A Faculdade Natalense para o Desenvolvimento do Rio Grande do Norte, particularmente nas pessoas de seus Diretores, Professores e Funcionários, pela presença e suporte que me ofereceram para a realização e conclusão desta oportunidade produtiva e valerosa que é a produção de conhecimento.

E a todos que, de alguma forma, contribuíram na elaboração deste trabalho.

Agradeço a Deus em primeiro lugar e a minha família, dedico este trabalho que é resultado de meu esforço e também do apoio que deles recebi em todos os projetos de minha vida. Em particular à minha esposa Eliana e a minha mãe Gezilda pelo apoio constante.

Aos meus amados filhos Erik e Evellyn.

AGRADECIMENTOS

Este trabalho foi construído com o apoio e incentivo de várias pessoas e instituições. A elas venho agradecer com as mais sinceras palavras, esperando ter atingido os fins pretendidos neste empenho conjunto e atendido as expectativas de todos que contribuíram para sua construção.

Assim, agradeço a todos os meus professores, pelos ensinamentos ministrados e pelo estímulo aos desafios profissionais e acadêmicos que deles recebi.

Aos meus professores revisores e orientadores, em especial o Prof. Fellipe Aleixo, que incondicionalmente me distinguiu com sua atenção e soube perceber os erros e acertos deste trabalho, fazendo-o sempre melhor.

À Faculdade Natalense para o Desenvolvimento do Rio Grande do Norte, particularmente nas pessoas de seus Diretores, Professores e Funcionários, pela presteza e suporte que me ofereceram para a realização e conclusão desta oportunidade produtiva e valorosa que é a produção de conhecimento.

E a todos que, de alguma forma, contribuíram na elaboração deste trabalho.

Sumário

1.	INTRODUÇÃO.....	11
2.	REFERENCIAL TEÓRICO	
2.1.	BANCO DE SANGUE.....	12
2.2.	INTERNET.....	14
2.3.	ORIENTAÇÃO A OBJETOS.....	15
2.4.	JAVA.....	17
2.4.1.	Principais Características do Java	18
2.4.2.	Exemplo de Código em Java.....	20
2.4.3.	JSP (JavaServer Pages).....	20
2.4.3.1	exemplo de código em JSP.....	22
2.5.	HTML.....	23
2.5.1.	Exemplo de Código em HTML.....	23
2.6.	UML.....	24
2.7.	PostgreSQL.....	27
2.8.	Dreamweaver.....	28
3.	ANÁLISE E PROJETO DO SISTEMA WEB PARA BANCO DE SANGUE	
3.1.	DIAGRAMA GERAL DO SISTEMA.....	29
3.2.	DIAGRAMAS DE CASO DE USO	
3.2.1.	Diagrama de caso de uso: Agência.....	30
3.2.2.	Diagrama de Caso de Uso: Matriz.....	30
3.2.3.	Diagrama de Caso de Uso: Hospital.....	31
3.2.4.	Diagrama de Caso de Uso: Doador.....	31
3.2.5.	Diagrama de Caso de Uso: Gerente.....	32
3.3.	CASOS DE USO DE ALTO NÍVEL.....	32
3.3.1.	Descrição de Caso de Uso: Login.....	32
3.3.2.	Descrição de Caso de Uso: Cadastrar paciente.....	33
3.3.3.	Descrição de Caso de Uso: Cadastrar exames do paciente.....	33
3.3.4.	Descrição de Caso de Uso: Solicitar hemocomponente.....	33
3.3.5.	Descrição de Caso de Uso: Solicitar transferência de estoque.....	34
3.3.6.	Descrição de Caso de Uso: Transferir estoque.....	34
3.3.7.	Descrição de Caso de Uso: Consultar estoque.....	34
3.3.8.	Descrição de Caso de Uso: Cadastrar liberação de hemocomponentes.....	35
3.3.9.	Descrição de Caso de Uso: Agendar doação.....	35
3.3.10.	Descrição de Caso de Uso: Solicitar declaração de doação.....	35
3.4.	CASOS DE USO EXPANDIDO	
3.4.1.	Caso de uso expandido: Consultar estoque.....	36
3.4.2.	Caso de uso expandido: Cadastrar paciente.....	37
3.4.3.	Caso de uso expandido: Cadastrar exames do paciente.....	39

3.4.4.	Caso de uso expandido: Solicitar hemocomponente.....	40
3.4.5.	Caso de uso expandido: Liberação de hemocomponentes.....	42
3.4.6.	Caso de uso expandido: Solicitar transferência de estoque.....	44
3.4.7.	Caso de uso expandido: Transferir estoque.....	45
3.4.8.	Caso de uso expandido: Agendar doação.....	47
3.4.9.	Caso de uso expandido: Solicitar declaração de doação.....	49
3.5.	MODELOS DE DOMÍNIO	
3.5.1.	Modelo de Domínio do Sistema.....	51
3.6.	DIAGRAMAS DE SEQUÊNCIA DO SISTEMA	
3.6.1.	Diagrama de sequência - fase 1 – Caso de uso solicitar hemocomponente	52
3.6.1.1.	Diagrama de sequência - fase 2 – Caso de uso solicitar hemocomponente.....	52
3.6.2.	Diagrama de sequência - fase 1 – Caso de uso consultar estoque.....	53
3.6.2.1.	Diagrama de sequência - fase 2 – Caso de uso consultar estoque.....	53
3.6.3.	Diagrama de sequência - fase 1 – Caso de uso cadastro de paciente.....	54
3.6.3.1.	Diagrama de sequência - fase 2 – Caso de uso cadastro de paciente.....	54
3.6.4.	Diagrama de sequência - fase 1 – Caso de uso cadastro de exames do paciente.....	55
3.6.4.1.	Diagrama de sequência - fase 2 – Caso de uso cadastro de exames do paciente.....	55
3.6.5.	Diagrama de sequência - fase 1 – Caso de uso liberar hemocomponente.....	56
3.6.5.1.	Diagrama de sequência - fase 2 – Caso de uso Caso de uso liberar hemocomponente.....	56
3.6.6.	Diagrama de sequência - fase 1 – Caso de uso transferir estoque.....	57
3.6.6.1.	Diagrama de sequência - fase 2 – Caso de uso Caso de uso transferir estoque.....	57
3.6.7.	Diagrama de sequência - fase 1 – Caso de uso liberar solicitar transferência de estoque.....	58
3.6.7.1.	Diagrama de sequência - fase 2 – Caso de uso Caso de uso solicitar transferência de estoque.....	58
3.6.8.	Diagrama de sequência - fase 1 – Caso de uso agendar doação.....	59
3.6.8.1.	Diagrama de sequência - fase 2 – Caso de uso Caso de uso agendar doação.....	59

3.6.9. Diagrama de seqüência - fase 1 – Caso de uso liberar solicitar declaração de doação.....	60
3.6.9.1. Diagrama de seqüência - fase 2 – Caso de uso Caso de uso solicitar declaração de doação.....	60
3.7. TELAS DO SISTEMA.....	
3.7.1. Tela do index principal do sistema.....	61
3.7.2. Tela do módulo doador.....	62
3.7.3. Tela solicitação agendamento de doação.....	63
3.7.4. Tela de solicitação de declaração de doação.....	65
3.7.5. Tela de login do sistema.....	67
3.7.6. Tela do módulo index agência.....	69
3.7.7. Tela de solicitação de hemocomponentes.....	70
3.7.8. Tela de consulta de estoque.....	73
3.8. CÓDIGOS JSP.....	
3.8.1. Código JSP autentica login.....	76
3.8.2. Código JSP cadastro de paciente.....	77
3.9. CÓDIGOS JAVA.....	
3.9.1. Código Java pacienteBD.....	78
3.9.2. Código Java classe paciente.....	79
4. CONCLUSÃO.....	82
BIBLIOGRAFIA.....	83

Resumo

A Internet nos dias atuais proporciona cada vez mais uma expansão nos serviços prestados ao público, em geral, conectado a ela, entre estes serviços está o serviço de saúde. Desta forma o serviço de saúde estará mais preparado para enfrentar o futuro, como também oferecer serviços de qualidade e atingir seu objetivo principal que é o atendimento a pacientes que necessitam de tratamento hemoterápico. Este trabalho consistiu em avaliar e analisar as dificuldades dos serviços prestados por instituições de hemoterapia de Natal - RN. Desta avaliação identificou-se a necessidade de desenvolver um sistema que dê suporte as agências e hospitais vinculados ao serviço, interligando os seus componentes via Web. Ao final, conclui-se que é viável a utilização da internet neste serviço, melhorando a agilidade no serviço e oferecendo informações a seus usuários podendo obter vantagens no atendimento ao paciente e usuários do serviço.

Key words: Blood bank; Internet; Service of health

Palavra-Chave: Banco de Sangue; Internet; Serviço de saúde.

1. INTRODUÇÃO

Abstract

The Internet in the current days provides an expansion more and more in the services rendered the connected general public the same, among them: the services of health, this way, will be more prepared to face the future, as well as to offer quality services and to reach your main objective that is the attendance to patients that need transfusion treatment. This work consisted of to evaluate and to analyze the difficulties of the services institutions of transfusion treatment of Natal - RN. Of this evaluation the need was identified of developing a system that gives support the agencies and hospitals linked to the service, calling your components through Web. At the end, it is ended that is viable the use of the internet in this service, improving the agility in the service and offering information to your users could obtain advantages in the attendance to the patient and users of the service.

Key words: Blood bank; Internet; Service of health.

1. INTRODUÇÃO

Nos dias de hoje, em virtude da globalização, a informação é considerado um fator primordial para o desenvolvimento do ser humano. Desde os tempos mais remotos o homem tem criado e aprimorado os símbolos e os significados, bem como seus sistemas para que as informações sejam transmitidas e compreendidas com a máxima eficiência e eficácia. Para tal, se faz necessário que sejam utilizados meios e processos de comunicação capazes de suprir as necessidades com base na quantidade e na qualidade.

Tem-se observado nos serviços de saúde, um crescente avanço tecnológico com vistas à qualidade da assistência e racionalização de custos. Sendo a globalização principal causa deste avanço, Ela não deve ser entendida somente como um conceito meramente econômico, mais também como o resultado da revolução da tecnologia da informação. Neste contexto os serviços de saúde, sejam públicos ou privados, necessitam cada vez mais contribuir para o desenvolvimento dos serviços prestados a sociedade, sendo necessário para tal, uma verdadeira "Revolução Organizacional" objetivando a melhoria na qualidade e no desenvolvimento de seu sistema funcional. O Banco de Sangue, como um serviço de saúde, têm a necessidade de participar desse desenvolvimento, principalmente no que diz respeito à informação, pois o sistema de informação, automatizado ou não, é vital para este serviço que tem no processo de comunicação o seu componente principal.

A utilização do computador no trabalho reduz o tempo gasto na veiculação de informações, de material e de pessoal envolvido. Aumenta também a eficácia e a fidelidade no desempenho das tarefas que necessitem de redação manual, tornando os dados mais legíveis e sistematizados, levando a uma contribuição para a melhoria da organização no gerenciamento das ações e cumprimento de normas da instituição.

Neste projeto apresentamos meios para a informatização de um Banco de Sangue, utilizando tecnologias baseadas em Internet, linguagens de programação Java, JSP e HTML, além de conhecimentos específicos em Banco de Sangue. Informatização essa, com o objetivo de dar suporte a todos que interagem com os serviços do Banco de Sangue, interligando todos os processos que compõe este serviço e todas as instituições que dele faça parte.

2. REFERÊNCIAL TEÓRICO

Para melhor entendimento deste projeto foram definidos alguns assuntos de grande importância para o leitor.

Para implementação de um sistema de informação utilizando a Internet e um Banco de Sangue, se fez necessário pesquisar sobre o funcionamento de um Banco de Sangue em todas as suas áreas, como também de Internet, Orientação a Objetos, UML, Java, JSP e HTML.

2.1. BANCO DE SANGUE

O Banco de Sangue é uma instituição sem fins lucrativos, não mercantil, responsável por capitar doadores, coletar, processar, distribuir e transfundir sangue e seus componentes e derivados. (NORMAS, 2002).

A história da Hematologia no Brasil na década de 80 era crítica, pois por anos ficou delegada ao esquecimento, sem gerenciamento e fiscalização do governo e sem perspectiva de melhoramento. Com o aparecimento da SIDA – Síndrome da Imuno Deficiência Adquirida, seguidas de outras doenças infecto contagiosas transmitidas pelo sangue como a Hepatite, Doença de Chagas, Sífilis entre outras, o assunto passou a chamar a atenção das autoridades governamentais. A maioria dos Bancos de Sangues apresentava graves irregularidades, tais como, a não realização de exames para detecção de tais doenças, triagem inadequada de doadores, condições de armazenamento inadequado do sangue, ausência de responsável técnico com formação médica na área, espaço físico inadequado e ausência de alvará de funcionamento, levando a interdição de vários serviços de Hemoterapia. (HAAF, 2005).

Hoje, no Brasil, os serviços hemoterápicos são regidos pelas normas técnica explicitadas na Resolução da Diretoria Colegiada número 343, de 13 de dezembro de 2002, editado pela Agência Nacional de Vigilância Sanitária. (LEGISLAÇÃO, 1999).

Com o desenvolvimento da medicina transfusional, ou hemoterapia (a especialidade médica que estuda o sangue, seus componentes e derivados com o objetivo de utilizá-los terapeuticamente, como também dos aspectos envolvidos na coleta, conservação e transfusão), incorporou-se novas metodologias, fundamentais para a prática moderna da medicina, entre as quais se inclui a aférese, a crio-biologia e os bancos de células derivadas de fontes diversas, levando a moralização deste serviço em todo o país, e a informática têm grande contribuição

neste desenvolvimento, tanto no que diz respeito a técnicas e equipamentos, como na parte dos registros. (HEMOTERAPIA, 2005).

A Agência Transfusional ou Banco de Sangue é o setor do hospital responsável pelo estoque de hemocomponentes e hemoderivados (concentrado de hemácias, concentrado de plaquetas, concentrado de Fator VIII, plasma fresco congelado, etc.). A aplicação destes hemocomponentes nos diversos pacientes conforme requisições e indicações médicas, investigação e acompanhamento de reações à transfusão de hemocomponentes, recuperação de sangue em cirurgias de grande volume transfusional, realização de exames imuno-hematológicos que possam contribuir para o diagnóstico e seguimento terapêutico de diversos pacientes e realização de programas e campanhas intra-hospitalares que facilitem o aumento da reposição e captação de doadores de sangue e, também, que possibilitem o uso mais racional de hemocomponentes nas diversas transfusões sangüíneas. (RDC Nº. 153, 2004).

A Agência fica funcionando ininterruptamente 24 horas por dia objetivando principalmente o atendimento de emergências em transfusão sangüíneas. Conta com médicos, enfermeiras e técnicos em hemoterapia, com cursos em outros Hemocentros e capacitados para prestar um serviço de qualidade. Além disso, estão equipadas com equipamentos (geladeiras, freezer, centrífugas, cell-wash, pipetas automáticas, etc.) de moderna tecnologia, que permitem a conservação dos hemocomponentes assegurando a sua qualidade e eficácia transfusional, além de outros equipamentos que permitem a realização dos diversos exames com segurança e exatidão. (RDC Nº. 153, 2004).

Atualmente a informática já é uma realidade nos Bancos de Sangue, embora ainda existam algumas dificuldades com relação à disponibilização de seus sistemas. Alguns serviços de hemoterapia no estado do RN disponibilizam de sistemas de informação informatizados, porém a muito custo, por se tratar de um serviço não mercantil, e a realização de um sistema interligando os seus serviços ainda não foi possível. Já a nível nacional existem vários sistemas privados interligando estes serviços, porém todos em redes privadas.

Para visualizar melhor o funcionamento técnico de um Banco de Sangue observe a figura 1.

Fluxograma de Doação - O Percurso do Sangue Doador

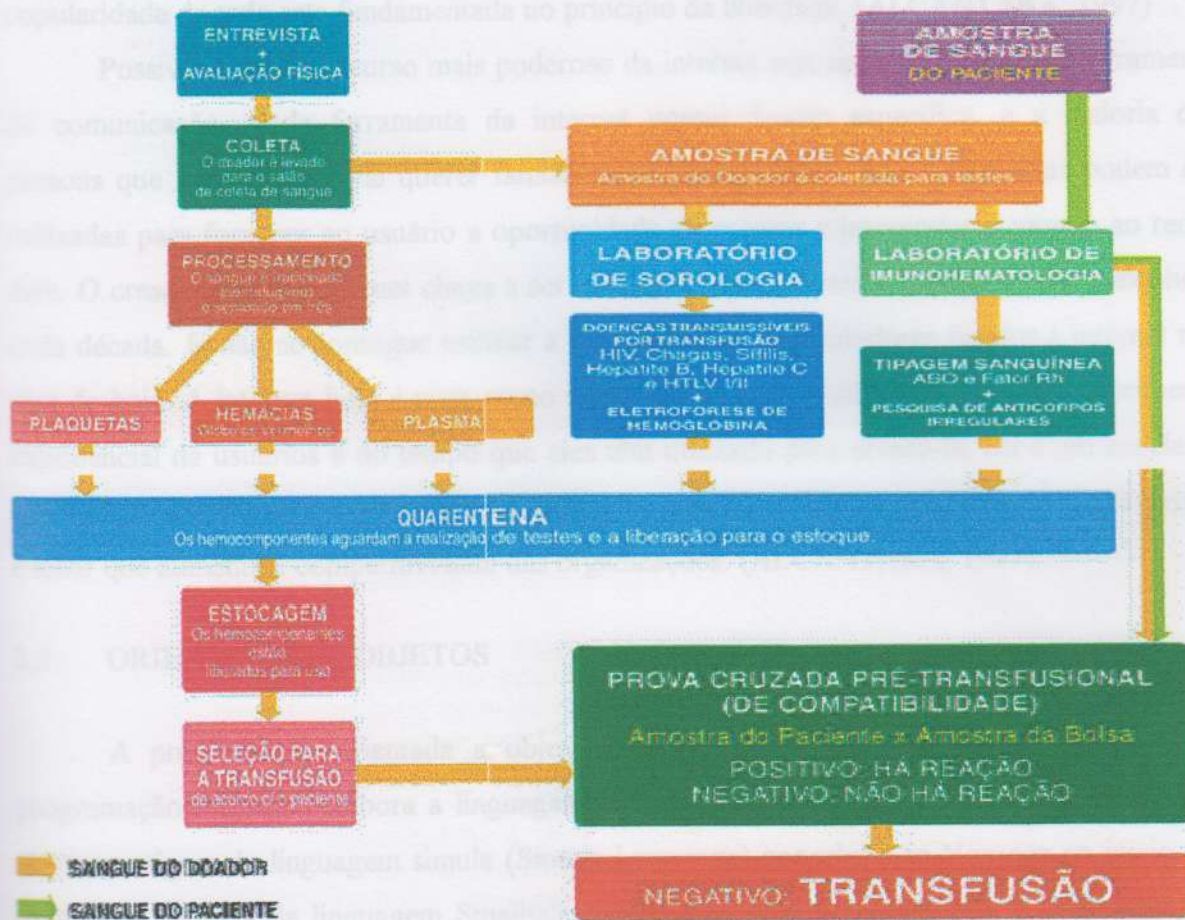


Figura 1. Esquema de funcionamento de um Banco de Sangue (EINSTEIN, 2004).

Iniciando pela triagem médica seguido pela coleta do sangue de um doador apto, segue-se o processamento do sangue, fazendo parte deste processamento, o fracionamento de seus componentes e derivados e análise do sangue. Nesta análise, o sangue é testado contra doenças infecto contagiosa e classifica-se o tipo sanguíneo. Após este período de quarentena, o sangue é estocado e liberado para transfusão após exames de compatibilidade positiva com o receptor (paciente).

2.2. INTERNET

A Internet é uma extensa rede de computadores interligados, mas independentes. Em menos de duas décadas transformou-se de uma rede altamente especializada de comunicações, utilizada principalmente para fins militares e acadêmicos, em um basar eletrônico de massa. (HEIDE, 2000).

Hoje a Internet é utilizada não só por cientistas, mas por várias outras pessoas. A popularidade da rede esta fundamentada no principio da liberdade. (ALCÂNTARA, 1997)

Possivelmente, o recurso mais poderoso da internet seja seu potencial como ferramenta de comunicação. Cada ferramenta da internet possui função específica, e a maioria das pessoas que a as explora vai querer familiarizar-se com todas. Tais ferramentas podem ser utilizadas para fornecer ao usuário a oportunidade de acessar e interpretar o mundo ao redor dele. O crescimento da Internet chega a ser assustador, praticamente duplicando de tamanho a cada década. Já não se consegue estimar a quantidade de computadores ligados à internet nos dias de hoje. A internet hoje é vista como uma rede de oportunidades e com um crescimento exponencial de usuários e do tempo que eles têm utilizado para acessá-la, ela é um excelente canal de comunicação que não pode ser ignorada, pois a possível interação com os internautas é fator que aumenta a competitividade das organizações. (ALCÂNTARA, 1997).

2.3. ORIENTAÇÃO A OBJETOS

A programação orientada a objeto (OO) pode ser considerada uma extensão da programação modular, embora a linguagem OO tenha sido concebida no início dos anos 70 ela teve origem da linguagem simula (Simula Language) concebida na Noruega no início dos anos 60 e seguida da linguagem Smalltalk, criada pela Xerox, considerada a linguagem que popularizou e incentivou o emprego da OOP. (LEITE, 2005)

Definindo de forma simples, um projeto orientado a objetos é uma técnica de desenvolvimento de sistemas que se concentra nos dados (ou objetos), e nas interfaces com esse objeto. Para fazer uma analogia com uma construção, se ela fosse "orientada a objeto", primeiro se concentraria no objeto a ser construído, depois que meios usaria para isso. Se fosse uma construção não-orientada a objetos ela se preocuparia primeiro com os meios e ferramentas a serem utilizados. (LEITE, 2005)

Segundo Leite (2005),

Uma definição para objeto seria a de um "ente" ativo dotado de certas características que o tornam "inteligente", a ponto de tomar certas decisões quando devidamente solicitado. Outra definição mais formal para objeto poderia ser: *uma unidade dinâmica, composta por um estado interno privativo (estrutura de dados) e um comportamento (conjunto de operações)*. E neste caso, segundo PRICE (1997), um objeto em particular é como um processador com memória própria e independente de outros objetos. Em termos de implementação, objeto é um bloco de dados privados envolvidos por código, de maneira que o acesso a ele só pode ser feito sob condições especiais. Todo o comportamento desse "ente" encapsulado é

descrito através de rotinas que manipulam seus dados, sendo que o seu estado corrente está em seus próprios dados; em outras palavras, cada objeto tem suas próprias características, moldadas a partir de uma matriz. Formalmente, para ser considerada uma linguagem OO, esta precisa implementar quatro conceitos básicos: abstração, encapsulamento, herança e polimorfismo.

De acordo com Leite (2005) os fortes da Orientação a Objeto são:

Proteção de dados: assegurar a confiabilidade e capacidade de modificação de um sistema pela redução de interdependências entre seus componentes. O estado de um módulo deve estar contido em vários locais visíveis somente do escopo do módulo, e os dados só devem ser manipulados por um conjunto localizado de procedimentos.

Abstração de dados: Mecanismo que utiliza a proteção de dados. O desenvolvedor define um tipo abstrato de dados consistindo de uma representação interna acrescida de um conjunto de procedimentos usados para acessar e manipular esses dados. Tanto a proteção quanto a abstração de dados aumentam a confiabilidade e facilitam a separação de especificações de procedimentos e representação.

Coesão Dinâmica: é indispensável utilizar um determinado tipo de código para outros tipos de dados. Processo em que o módulo chamador recebe o endereço da rotina associando uma mensagem a objetos com o método para executar essa mensagem.

Herança: Permite a criação de classes que são especializações de outras classes. A classe superior na hierarquia é denominada superclasse, e a inferior, subclasse. Uma subclasse herda variáveis instanciadas, variáveis de classe e métodos apropriados a objetos mais especializados. A subclasse pode ocultar ou fornecer um comportamento adicional às operações da superclasse.

Basicamente, a Orientação a Objetos utiliza os mesmos princípios da engenharia de *hardware* que projeta novos equipamentos usando os mesmos componentes básicos como transistores, resistores, fusíveis, diodos, chips, etc. Os "objetos" já existentes são utilizados para produzir novos "objetos", tornando essa metodologia mais poderosa que as metodologias tradicionais de desenvolvimento de sistemas. (CONSTANTINE, 2000)

Se considerarmos a Orientação ao Objeto como um novo paradigma de desenho de software, deve considerar, também, uma nova maneira de pensar, porque apesar de a escrita do código continuar sendo procedural, alguns conceitos mudam radicalmente: a estruturação e o modelo computacional. Fundamentalmente o que se deseja com esta metodologia são basicamente duas características: reutilização de código e modularidade de escrita; e nisto a OOP é imbatível quando comparada com as metodologias antigas. (CONSTANTINE, 2000)

Em termos de modelo computacional podemos dizer que enquanto as metodologias tradicionais utilizam o conceito de um processador, uma memória e dispositivos de I/O para processar, armazenar e exibir as informações, a OOP emprega um conceito mais real, mais concreto, que é o de Objeto. (CONSTANTINE, 2000)

2.4. JAVA *Principais Características do Java*

Java é uma linguagem de programação orientada a objetos, desenvolvida pela *Sun Microsystems*. Inicialmente elaborada para ser a linguagem-base de projetos de software para produtos eletrônicos, Java teve seu grande *boom* em 1995, devido ao sucesso mundial da *World Wide Web*. (NIEMAYER, 2000)

Como sabemos, a Internet é uma gigantesca "rede" (conceitualmente falando) que liga "milhares" de computadores entre si. É uma rede heterogênea, já que diversos tipos de computadores estão ligados a ela. Todos esses computadores utilizam-se do protocolo *TCP/IP* para comunicarem-se. Como eles são distintos, eles precisam de uma linguagem que não esteja necessariamente amarrada a uma plataforma de hardware/software específica. (NIEMAYER, 2000)

Os programas Java podem ser transmitidos como bytecodes e podem rodar em qualquer computador sem necessitar uma nova recompilação. Característica esta de independência de plataforma. (NIEMAYER, 2000)

Há dois tipos básicos de programas Java: *Aplicativos*, que são programas como outros quaisquer, e *Applets*, programas especialmente confeccionados para executarem dentro de uma página HTML. Ao abrir uma página HTML que tenha inserido em si uma applet Java, esta é automaticamente descarregada para seu computador e executada. (NIEMAYER, 2000)

Dáí vem à pergunta: Há riscos de um vírus infectar o seu computador cliente? Isso não ocorre, pois devido a uma série de limitações que os projetistas de Java impuseram as applets, por razões de segurança puramente, de forma que nenhuma applet Java seja capaz de "roubar" informações ou danificar dados do computador. A razão de toda essa segurança reside no fato de que programas Java são compilados em bytecodes que são verificados. Instruções bytecodes são muito similares a outros conjuntos de instruções projetados para plataformas específicas, com a diferença de que bytecodes são conferidos através de informações adicionais que eles carregam em si informando a legitimidade ou não do arquivo. (NIEMAYER, 2000)

2.4.1. Principais Características do Java

Java é uma linguagem de alto nível, com sintaxe extremamente similar à do C++, e com diversas características herdadas de outras linguagens, como *Smalltalk* e *Modula-3*. É antes de tudo uma linguagem simples, independente de arquitetura, robusta, segura, extensível, bem estruturada, distribuída, multithreaded e com garbage collection. (DEITEL, 2000)

Simplicidade: Java é muito parecida com C++, mas muito mais simples. Java não possui sobrecarga de operadores, structs, unions, aritmética de ponteiros, herança múltipla, diretivas de pré-processamento e a memória alocada dinamicamente é gerenciada pela própria linguagem, que usa algoritmos de garbage collection para desalocar regiões de memória que não estão mais em uso. (DEITEL, 2000)

Orientação a Objetos: Ao contrário de C++, que é uma linguagem híbrida, Java é uma linguagem orientada a objetos que segue a linha purista iniciada por *Smalltalk*. Com a exceção dos tipos básicos da linguagem (int, float, etc.), a maior parte dos elementos de um programa Java são objetos. (*A título de curiosidade, Smalltalk já é considerada puramente O.O. [orientada a objeto], pois absolutamente tudo em Smalltalk são objetos, não há tipos básicos.*). O código é organizado em classes, que podem estabelecer relacionamentos de herança simples entre si. Somente a herança simples é permitida em Java.. (DEITEL, 2000)

Processamento distribuído: Chamadas a funções de acesso remoto (sockets) e os protocolos Internet mais comuns (HTTP, FTP, Telnet, etc.) são suportadas em Java, de forma que a elaboração de aplicativos baseados em arquiteturas cliente-servidor é facilmente obtida. (DEITEL, 2000)

Multithreading: As maiores partes dos sistemas operacionais hoje no mercado dão suporte à multitarefa, um programa Java pode possuir mais de uma linha de execução (thread). Por exemplo, cálculos extensos, que em geral demandam muito tempo do processador, podem ser escritos em uma thread, e a parte de interface com o usuário, que depende mais dos periféricos de I/O que do processador, pode ser executada em outra thread. (DEITEL, 2000)

Exceção: Todo programador em geral está bastante acostumado com o computador "travando" por causa de um erro em um programa., a simples tentativa de abertura de um arquivo inexistente pode obrigar ao programador a reiniciar o computador. Programas Java, contudo, não dão pau no computador, já que a máquina virtual Java faz uma verificação em tempo de execução quanto aos acessos de memória, abertura de arquivos e uma série de eventos que podem gerar um travamento em outras linguagens, mas que geram exceções em programas Java. Em geral, ao escrever programas Java utilizando-se de herança de classes predefinidas, força-se em geral ao programador escrever algumas rotinas de tratamento de exceção, um trabalho que, se de início pode parecer forçado, irá poupar o programador de bastante dor de cabeça no futuro. (DEITEL, 2000)

Garbage collector: Em Java, os programadores não necessitam preocupar-se com o gerenciamento de memória como em C++. Java, ao contrário, utiliza-se de um conceito já explorado por Smalltalk, que é o de *garbage collection* (coleta de lixo). Sua função é a de varrer a memória de tempos em tempos, liberando automaticamente os blocos que não estão sendo utilizados. Se por um lado isso pode deixar o aplicativo um pouco mais lento, por manter uma thread paralela que dura todo o tempo de execução do programa, evita problemas como referências perdidas e avisos de falta de memória quando sabe-se que há megas e megas disponíveis na máquina. (DEITEL, 2000)

Machine independent: Uma das características de Java que a tornou ideal para seu uso na elaboração de aplicativos distribuídos foi a sua independência de plataforma. Afinal, a Internet é uma grande "salada" de computadores de todos os tipos, dotados dos mais diversos sistemas operacionais e ambientes gráficos que se possa imaginar. Java consegue essa independência devido ao fato de que o compilador Java não gera instruções específicas a uma plataforma, mas sim um programa em um código intermediário, denominado bytecodes, que pode ser descrito como uma linguagem de máquina destinada a um processador virtual que não existe fisicamente. (DEITEL, 2000)

2.4.2. Exemplo de Código em Java

<pre>//criação de uma classe public class Automovel{ //atributos de uma classe String modelo, cor; Int ano; boolean estadoMotor false; //métodos de uma classe public void ligarMotor(){ estadoMotor = true; } public String getModelo(){ Return modelo; } Public void setAno (int newAno){ Ano = newAno; } }</pre>	<pre>//O programa executável public class Automovel { public static void main(String args[]){ // instanciando uma classe (criando objeto) Automóvel automovel = new Automovel(); // chamando método de um objeto automovel.ligarMotor(); // System.out.println("Motor ligado"); } }</pre>
--	---

Figura 2. Código Java.

2.4.3. JSP (JavaServer Pages)

JavaServer Pages (JSP) é uma tecnologia utilizada para desenvolvimento de Sites Web dinâmicos utilizando a linguagem orientada a objeto Java. Esta tecnologia fornece ao designer de HTML uma forma de acessar dados e lógicas armazenados em Java sem, necessariamente, dominar a linguagem Java. (TODD, 2003)

JSP (Java Server Pages) é uma tecnologia para desenvolvimento de aplicações WEB semelhante ao ASP (Active Server Pages), porém tem a vantagem da portabilidade de plataforma podendo ser executado em outros Sistemas Operacionais além dos da Microsoft. Ela permite que ao desenvolvedor de sites produza aplicações que permitam o acesso a banco de dados, o acesso a arquivos-texto, a captação de informações a partir de formulários, a captação de informações sobre o visitante e sobre o servidor, o uso de variáveis e loops entre outras coisas. (TODD, 2003)

Quem conhece servlets verá que o JSP não oferece nada que você não possa conseguir com os servlets puros, entretanto, oferece a vantagem de ser facilmente codificado, facilitando

assim a elaboração e manutenção de uma aplicação. Além disso, essa tecnologia permite separar a programação lógica (parte dinâmica) da programação visual (parte estática), facilitando o desenvolvimento de aplicações mais robustas, onde programador e designer podem trabalhar no mesmo projeto, mas de forma independente. Outra característica do JSP é produzir conteúdos dinâmicos que possam ser reutilizados. Quando uma página JSP é requisitada pelo cliente através de um Browser, esta página é executada pelo servidor, e a partir daí será gerada uma página HTML que será enviada de volta ao browser do cliente. (TODD, 2003)

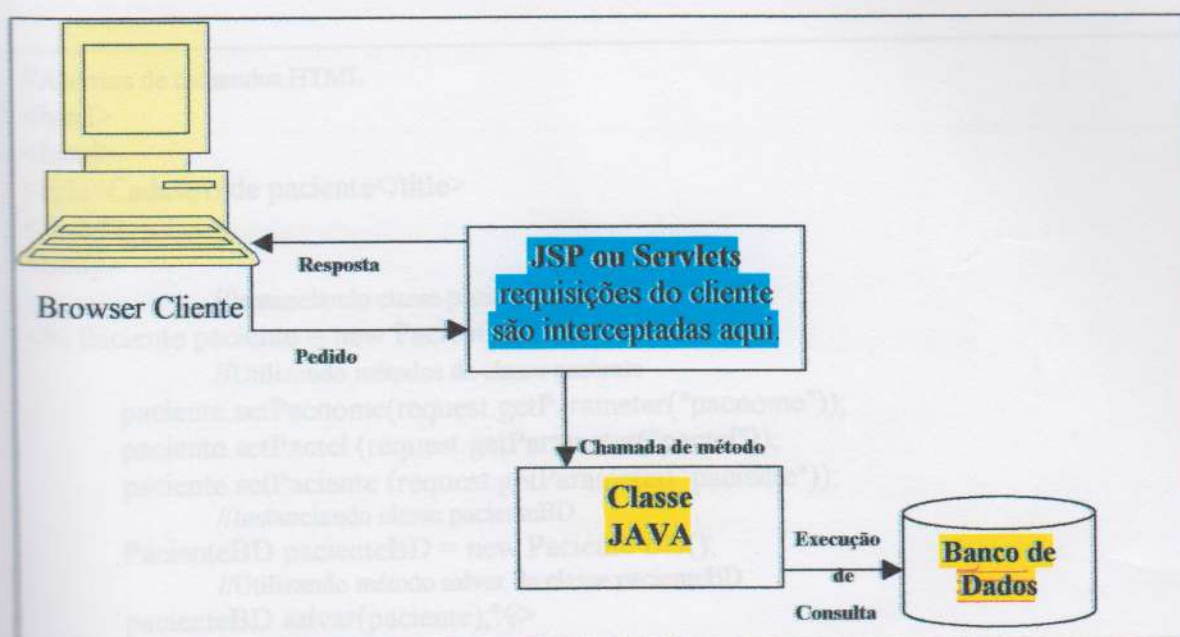


Figura 3. Esquema de funcionamento de uma JSP.

Quando o cliente faz a solicitação de um arquivo JSP, como vemos na figura 3, é enviado um pedido para a JSP engine, ela envia a solicitação de qualquer componente (podendo ser um Javabeans component, servlets ou enterprise Beans) especificado no arquivo.

O componente controla a requisição possibilitando a recuperação de arquivos em banco de dados ou outro dado armazenado, em seguida, passa o objeto response de volta para a JSP engine. A JSP engine e o WEB Server enviam a página JSP revisada de volta para o cliente, onde o usuário pode visualizar os resultados através do browser. O protocolo de comunicação usado entre o cliente e o servidor pode ser HTTP ou outro protocolo. (TODD, 2003).

Por definição, JSP usa Java como sua linguagem de scripts. Por esse motivo, o JSP é mais flexível e mais robusto do que outras plataformas baseadas simplesmente em Javascripts e Vbscripts (TODD, 2003).

Para tirar um bom proveito do JSP é preciso entender a linguagem Java. Isso porque, a parte lógica do JSP envolve Java Beans, Objetos JDBC, Enterprise Java Beans (EJB) entre outros componentes que interagem com a plataforma Java. (TODD, 2003)

2.4.3.1.Exemplo de código em JSP.

```
//Abertura de comandos HTML
<html>
<head>
<title>Cadastro de paciente</title>
</head>
<body>
    //Instanciando classe paciente através do JSP
    <% Paciente paciente = new Paciente ();
    //Utilizando métodos da classe paciente
    paciente.setPacnome(request.getParameter("pacnome"));
    paciente.setPactel (request.getParameter("pactel"));
    paciente.setPaciente (request.getParameter("paciente"));
    //Instanciando classe pacienteBD
    PacienteBD pacienteBD = new Paciente BD();
    //Utilizando método salvar da classe pacienteBD
    pacienteBD.salvar(paciente);%>
    //Imprimindo na tela
    !!! Cadastro efetuado com sucesso !!!<br><br><br><br>
<center>
    //Referenciando uma classe JSP
    <a href="cadastropaciente.jsp"><font color="orange"><center>Voltar</font></center>
</body>
</html>

//Fechamento de comandos HTML
```

Figura 4. Código JSP.

Observamos na figura 4 que o código JSP está inserido entre as tags do HTML através da qual podemos fazer referências a classes Java, instanciar objetos, trabalhar métodos de classes Java, entre outros.

2.5. HTML

HTML é a linguagem base para se trabalhar na WEB. A sigla HTML deriva da expressão inglesa HyperText Markup Language. Trata-se de uma linguagem de marcação utilizada para produzir páginas na Internet. Esses códigos podem ser interpretados pelos browsers para exibir as páginas da World Wide Web. Seu início se confunde com o aparecimento da internet por se tratar de uma linguagem de marcação todos os documentos hipertexto escritos na WEB são escritos nessa linguagem.

2.5.1. Exemplo de Código em HTML

```
<html>
<body>
  <form method="get">
    NOME:<input type="text" name="nome"><br>
    SENHA:<input type="password" name="cod"><br>
    SEXO:<input type="radio" name="sex" value="M">MASC &nbsp; &nbsp; &nbsp;
    <input type="radio" name="sex" value="F">FEM <BR>
    SO:<input type="checkbox" name="so" value="1">linux &nbsp; &nbsp; &nbsp;
    <input type="checkbox" name="so" value="2" checked>windows<br>
    <input type="submit" name="op" value="ENVIAR INFORMAÇÕES">
    <input type="reset" value="LIMPAR FORMULÁRIO">
  </form>
</body>
</html>
```

Figura 5. Código HTML.

Na figura 5 utilizamos a tag form que define o corpo do formulário, input que determina os campos do formulário atribuindo valores através dos métodos checkbox, que permite que mais de um item seja selecionado e radio que permite que apenas um item seja selecionado na lista.

Resultado do código na figura 6.

Formulário de registro de usuário:

NOME:

SENHA:

SEXO: ☒ MASC ☐ FEM

SO: ☒ linux ☐ windows

Figura 6. Código HTML.

2.6. UML

Segundo Larman (2000), UML é a abreviação de linguagem de modelagem unificada (Unified Modeling Language), uma notação para modelagem de sistemas usando o conceito orientado a objeto.

A UML tem origem na compilação das "melhores práticas de engenharia" que provaram ter sucesso na modelação de sistemas grandes e complexos. Sucedeu aos conceitos de Booch, OMT (Rumbaugh) e OOSE (Jacobson) fundindo-os numa única linguagem de modelagem comum e largamente utilizada. (LARMAN, 2000)

A criação da UML teve início em outubro de 1994, quando Rumbaugh se juntou a Booch com o objetivo de unificar os métodos Booch e OMT, decorrido um ano de trabalho, foi lançado, em outubro de 1995, o esboço da versão 0.8 do Método Unificado. Nesta mesma época, Jacobson se juntou ao projeto e o escopo do projeto da UML foi expandido para incorporar o método OOSE, nascendo então, em junho de 1996, a versão 0.9 da UML. (LARMAN, 2000)

Para desenvolver um sistema, é indispensável uma descrição do problema e seus requisitos, ou seja, qual o problema e o que o sistema deve fazer. (LARMAN, 2000)

Durante essa análise com UML, há uma ênfase na descoberta e na descrição dos objetos, ou conceitos, do domínio do problema, já durante o projeto em UML, A ênfase passa

a definição dos elementos lógicos de software os quais serão implementados em uma linguagem de programação. (LARMAN, 2000)

A UML utiliza de ferramentas como Casos de uso, Diagramas, Atores entre outros para padronizar a linguagem de entendimento entre o analista de sistema e o programador. (LARMAN, 2000)

Algumas ferramentas utilizadas na UML:

- **Atores:** São entidades externas que entram em contato com o sistema. Como vemos na figura 7, esses atores são representados por um desenho.

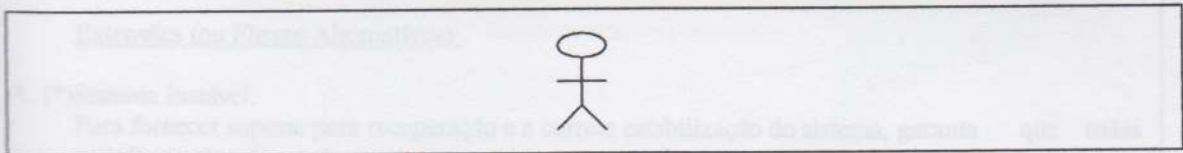


Figura 7. Ator.

- **Casos de uso:** Como vemos na figura 8, caso de uso é um documento narrativo que descreve seqüência de eventos de um ator (agente externo) que utiliza o sistema para completar um processo, podendo ser de alto nível ou expandido.
- Exemplo de caso de uso de alto nível.

Caso de uso: Cadastrar paciente

Atores: Técnico, Enfermeiro

Tipo: Primário

Descrição: O técnico recebe pedido de hemotransusão via Web confere dados do cadastro do paciente, caso não tenha cadastro, efetua registro no sistema.

Figura 8. Caso de uso alto nível.

- Exemplo na figura 9 de caso de uso expandido Cadastrar paciente:

Ator Principal: Técnico, Enfermeiro.

Interessados e Interesses:

- Técnico: deseja a opção de cadastrar o paciente com seus devidos dados, tais como: nome, data de nascimento, filiação, sexo, enfermagem, leito, diagnóstico e nº. de prontuário, endereço, telefone e hospital, dados esses que devem ser precisos e não podem ocorrer erros.

Pré-condições: O usuário deve ser identificado e autenticado para poder ter acesso ao sistema, limitado ao seu perfil de usuário.

Garantias de sucesso (Pós-condições): O cadastro do paciente é gravado com sucesso em BD.

Cenário de Sucesso Principal (ou Fluxo Básico):

1. O técnico deve iniciar o sistema através de login e senha.
2. O sistema identifica o usuário e libera o sistema para uso do mesmo conforme perfil do usuário.
3. O técnico ao receber solicitação via web de uma transfusão sanguínea, abre o sistema solicitação e confirma os dados do paciente, caso positivo, confirma o registro no sistema.
4. O sistema registra os dados do paciente envia informações ao sistema cadastro de paciente e confirma ao usuário "cadastro efetuado com sucesso".

Extensões (ou Fluxos Alternativos):

A. (*) Sistema instável:

Para fornecer suporte para recuperação e a correta estabilização do sistema, garanta que todas as informações de estado sensíveis nas transações e todos os eventos possam ser recuperados a partir de qualquer passo do cenário:

- i. O técnico reinicia o Sistema, registra-se e solicita a recuperação do estado anterior.
- ii. O Sistema restaura o estado anterior
- iii. O Sistema detecta anomalias que impedem a restauração:
 - a. O Sistema avisa ao técnico sobre o erro, registra o erro e, então, entra em um novo estado de estabilidade.
 - b. O técnico começa um novo registro.

B.(1) Autenticação de usuário inválida:

Sistema informa falha na autenticação e rejeita a entrada do usuário.

C.(3) Solicitação manual:

O técnico deve efetuar a solicitação no sistema e segue com os passos seguintes a solicitação via web.

D.(4) Paciente já cadastrado: o técnico confere os dados ou atualiza os dados do paciente.

Requisitos Especiais:

- IU - Interface com o usuário de fácil entendimento por monitor de 15" SVGA com boa visualização a distância de um metro.
- Impressões feitas por impressoras a jato de tinta.
- Respostas rápidas do sistema, principalmente nos passos 3 com aviso sonoro e visual.
- De alguma forma, queremos segurança na gravação e recuperação dos dados do sistema.

Lista de Variações Tecnológicas e de Dados:

Nenhuma.

Frequência de Ocorrência: Poderia ser quase contínuo.

Problemas em Aberto:

- Atualizações das leis vigentes dos serviços de hemoterapia.
- Na utilização do sistema por mais de um usuário no mesmo horário.

Figura 9. Caso de uso expandido.

- Diagrama de casos de uso: como observamos na figura 10, é uma descrição gráfica onde utilizamos atores, setas ou linhas, indicando o fluxo dos dados ou estímulos ao sistema.

Identificamos o ator pelo nome e o Banco de Dados por dois pontos seguidos pelo nome do mesmo sublinhado. (CONSTANTINE, 2000)

- Exemplo de diagrama de casos de uso:

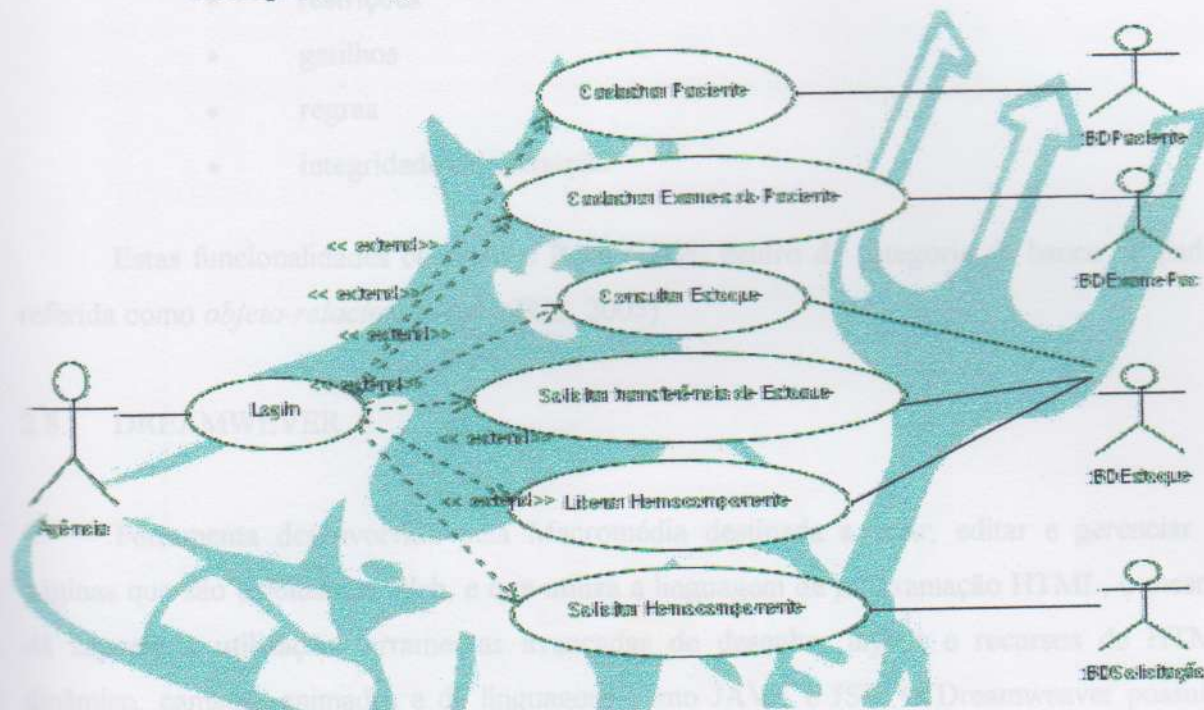


Figura 10. (Diagrama de caso de uso módulo Agência)

2.7. POSTGRESQL

O PostgreSQL é um sistema de gerenciamento de banco de dados Objeto-Relacional (SGBDOR) baseado no POSTGRES, versão 4.2, desenvolvido no Departamento de Ciências da Computação da Califórnia em Berkeley. Possui código fonte aberto e fornece suporte às linguagens SQL92/SQL99 além de trabalhar com conceitos como herança, tipo de dados, funções, restrições, gatilhos, entre outros que são cotidianos de quem trabalha com linguagem orientada a objetos. (BIAZUS, 2005)

O PostgreSQL foi pioneiro em muitos conceitos objetos-relacionais que agora estão se tornando disponíveis em alguns bancos de dados comerciais oferecendo recursos adicionais pela incorporação dos conceitos mostrados abaixo, tornando possível aos usuários estenderem o sistema facilmente:

- herança
- tipos de dado
- funções

Outras funcionalidades fornecem poder e flexibilidades adicionais:

- restrições
- gatilhos
- regras
- integridade da transação

Estas funcionalidades colocam o PostgreSQL dentro da categoria de banco de dados referida como *objeto-relacional*. (BIAZUS, 2005)

2.8. DREAMWEAVER

Ferramenta desenvolvida pela Macromédia destinada a criar, editar e gerenciar as páginas que são exibidas na Web, e que utiliza a linguagem de programação HTML, o mesmo dá suporte a utilização ferramentas avançadas de desenho, layout e recursos de HTML dinâmico, camadas animadas e de linguagens como JAVA e JSP. O Dreamweaver possui o aspecto de facilitar a criação de aplicativos que retorna informações de uma base de dados. (ARROYO, 2005)

Figura 11. Diagrama de caso de uso

3. ANÁLISE E PROJETO DO SISTEMA WEB PARA BANCO DE SANGUE

O protótipo do sistema proposto foi composto pelos módulos usuários Hospital, Agência, Doador, Gerente e Matriz. Optou-se pela divisão por módulos pela complexidade do sistema e para oferecer perfis personalizados a cada usuário do sistema.

O sistema oferece ao usuário todo o controle sobre os hemocomponentes e registros ligados ao mesmo, sendo possível ser acessado por qualquer usuário devidamente cadastrado em qualquer terminal que possua uma conexão com Internet através de login e senha.

3.1. DIAGRAMA GERAL DO SISTEMA

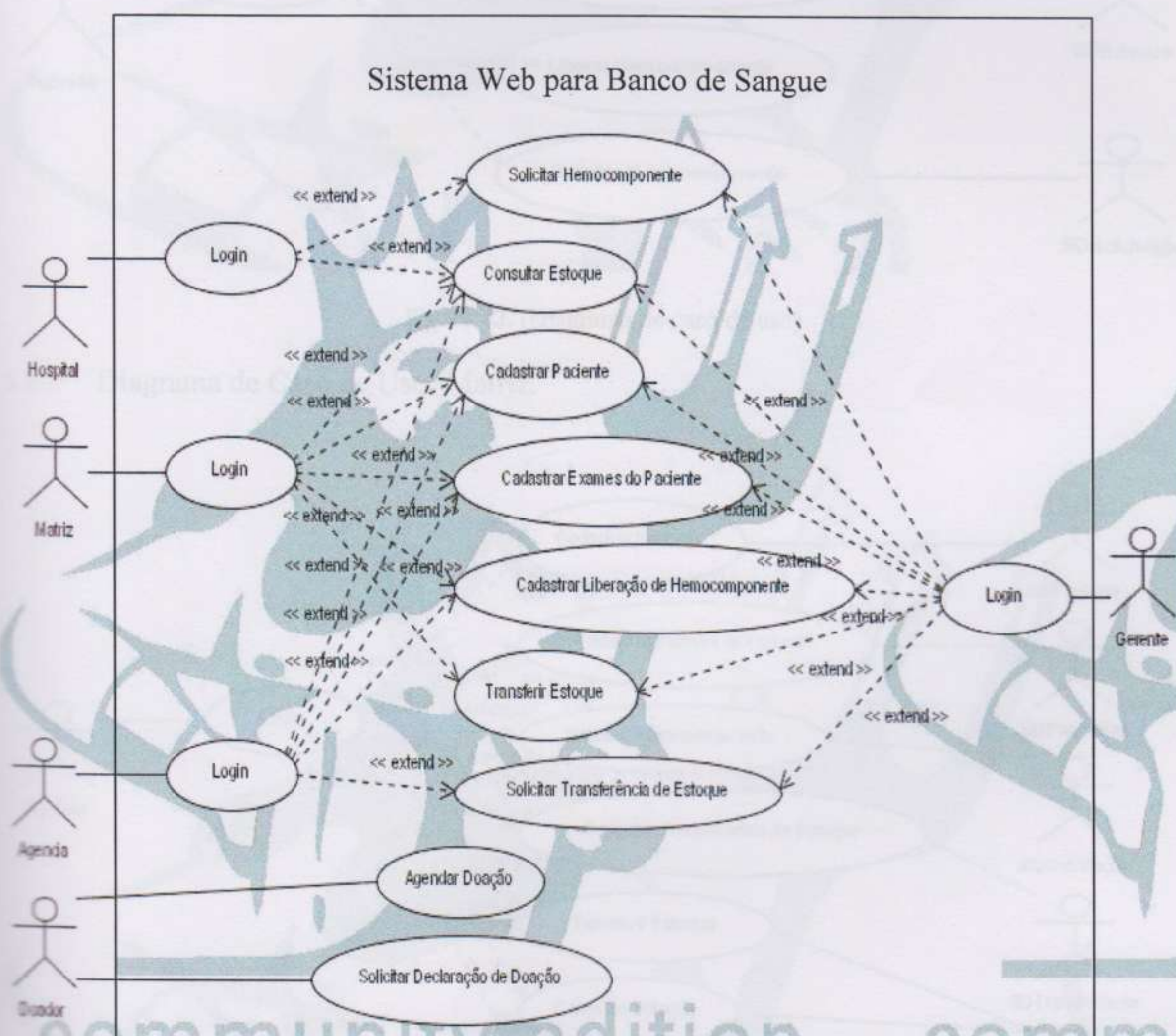


Figura 11. (Diagrama de caso de uso)

3.2. Diagramas de Caso de Uso

3.2.1. Diagrama de caso de uso: Agência

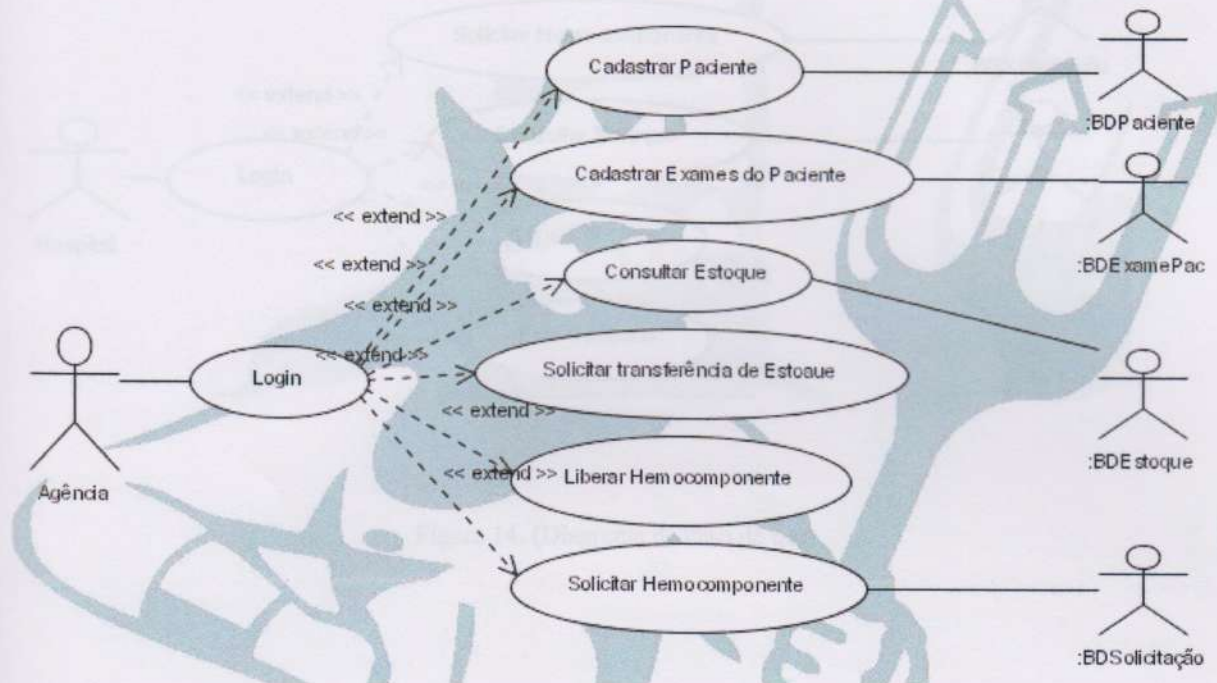


Figura 12. (Diagrama de caso de uso)

3.2.2. Diagrama de Caso de Uso: Matriz.

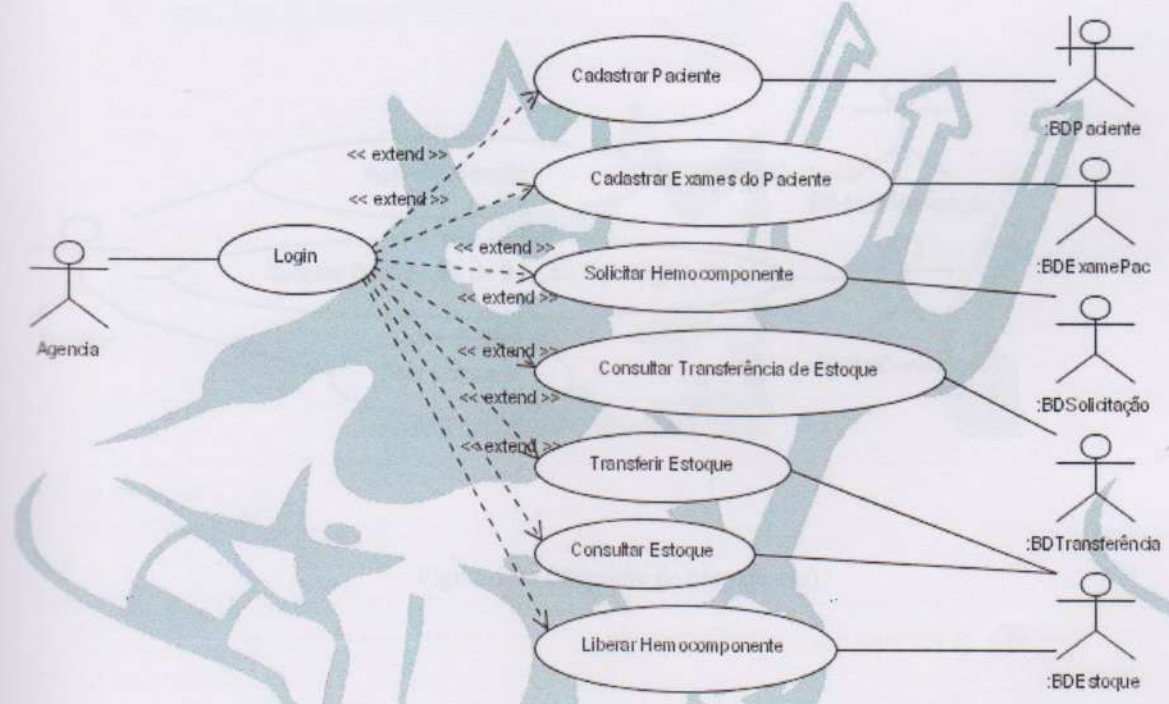


Figura 13. (Diagrama de caso de uso)

3.2.3. Diagrama de Caso de Uso: Hospital

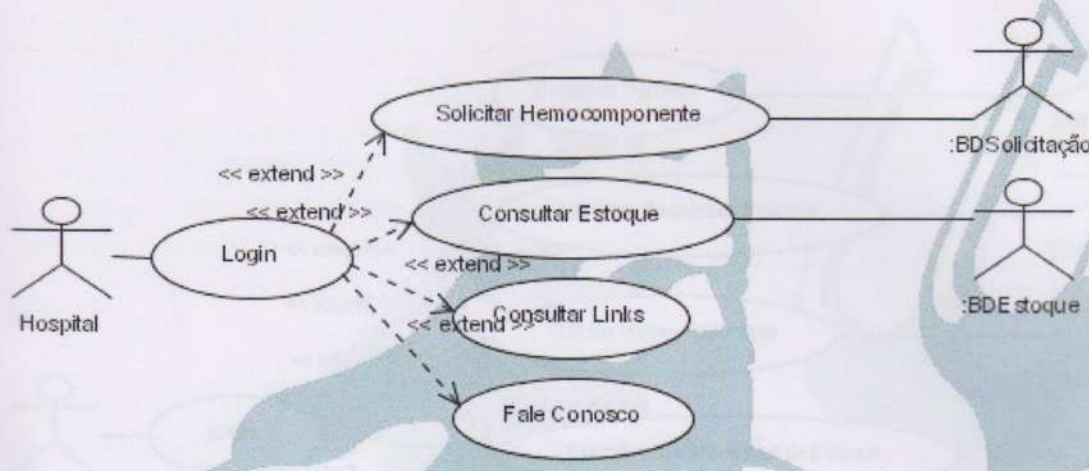


Figura 14. (Diagrama de caso de uso)

3.2.4. Diagrama de Caso de Uso: Doador

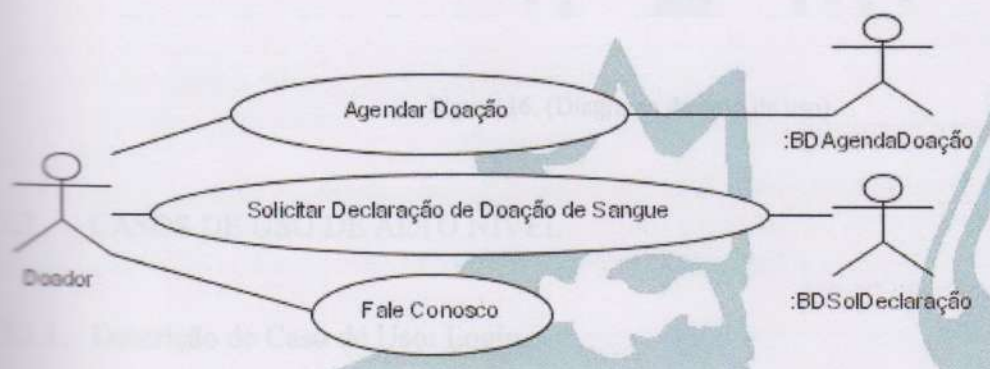


Figura 15. (Diagrama de caso de uso)

Caso de uso: Login

Assessor Técnico, Enfermeiro, médico.

Tipo Primário

Descrição: O usuário ao iniciar seu turno acessa o sistema e insere seu login e senha, após login validado o usuário tem acesso ao sistema limitado pelo seu perfil de usuário.

Figura 17. (Descrição do caso de uso)

3.2.5. Diagrama de Caso de Uso: Gerente

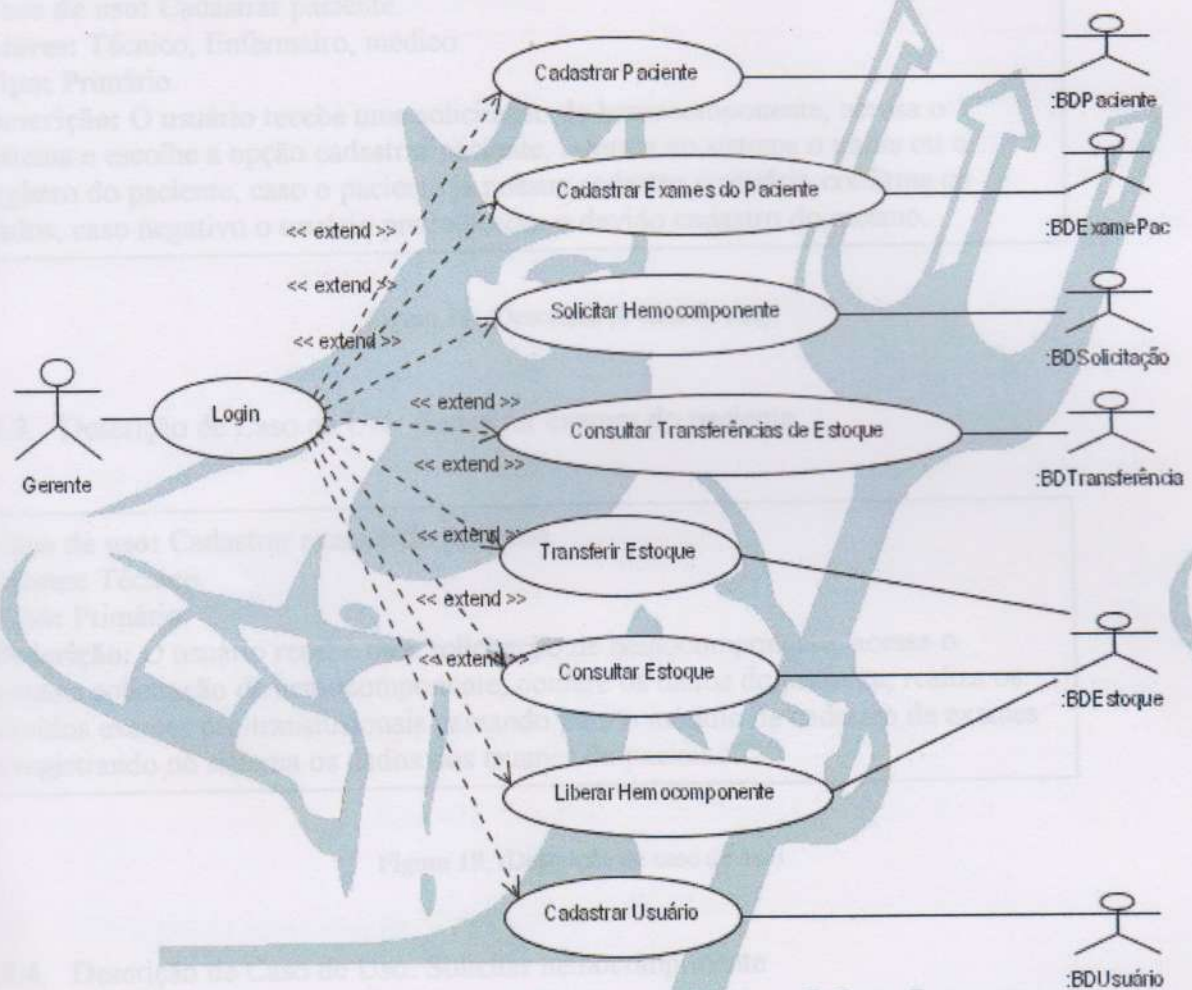


Figura 16. (Diagrama de caso de uso)

3.3. CASOS DE USO DE ALTO NÍVEL

3.3.1. Descrição de Caso de Uso: Login.

Caso de uso: Login.

Atores: Técnico, Enfermeiro, médico.

Tipo: Primário.

Descrição: O técnico ao iniciar seu turno abre o sistema e informa seu login e senha, após login validado o usuário tem acesso ao sistema limitado pelo seu perfil de usuário.

Figura 17. (Descrição de caso de uso)

3.3.2. Descrição de Caso de Uso: Cadastrar paciente

Caso de uso: Cadastrar paciente.

Atores: Técnico, Enfermeiro, médico.

Tipo: Primário.

Descrição: O usuário recebe uma solicitação de hemocomponente, acessa o sistema e escolhe a opção cadastrar paciente, informa ao sistema o nome ou o registro do paciente, caso o paciente já possua cadastro o usuário confirma os dados, caso negativo o usuário procede com o devido cadastro do mesmo.

Figura 18. (Descrição de caso de uso)

3.3.3. Descrição de Caso de Uso: Cadastrar exames do paciente

Caso de uso: Cadastrar exames do paciente.

Atores: Técnico.

Tipo: Primário.

Descrição: O usuário recebe uma solicitação de hemocomponente, acessa o sistema solicitação de hemocomponente, confere os dados do paciente, realiza os devidos exames pré-transfusionais passando para o módulo de cadastro de exames e registrando no sistema os dados dos exames do paciente.

Figura 19. (Descrição de caso de uso)

3.3.3. Descrição de Caso de Uso: Consultar estoque

3.3.4. Descrição de Caso de Uso: Solicitar hemocomponente

Caso de uso: Solicitar hemocomponente.

Atores: Técnico, enfermeiro, médico.

Tipo: Primário.

Descrição: O usuário necessita solicitar hemocomponente para transfusão, acessa o sistema no módulo solicitação de hemocomponente e informa ao sistema o nome do paciente, o tipo de hemocomponente, a quantidade, agência a quem se destina o pedido.

Figura 20. (Descrição de caso de uso)

3.3.5. Descrição de Caso de Uso: Solicitar transferência de estoque

Caso de uso: Solicitar transferência de estoque.

Atores: Técnico.

Tipo: Primário.

Descrição: O usuário necessita atualizar seu estoque de hemocomponentes, acessa o sistema no módulo solicitar transferência de estoque e informa ao sistema o tipo de hemocomponente, a quantidade, o grupo sanguíneo e a agência a quem se destina o pedido.

Figura 21. (Descrição de caso de uso)

3.3.6. Descrição de Caso de Uso: Transferir estoque

Caso de uso: Transferir estoque.

Atores: Técnico.

Tipo: Primário.

Descrição: O usuário recebe solicitação de transferência de estoque, acessa o sistema no módulo transferência de estoque e dá entrada no sistema os hemocomponentes a serem transferidos para a agência a quem se destina o pedido.

Figura 22. (Descrição de caso de uso)

3.3.7. Descrição de Caso de Uso: Consultar estoque

Caso de uso: Consultar estoque.

Atores: Técnico, Enfermeiro, médico.

Tipo: Primário.

Descrição: O usuário necessita verificar o estoque de hemocomponentes, acessa o sistema e escolhe a opção consultar estoque, informa ao sistema o tipo de consulta e o local onde o mesmo deseja consultar o estoque, podendo ser por hemocomponentes, por agência ou o estoque geral.

Figura 23. (Descrição de caso de uso)

3.3.8. Descrição de Caso de Uso: Cadastrar liberação de hemocomponentes

Caso de uso: Cadastrar liberação de hemocomponentes.

Atores: Técnico.

Tipo: Primário.

Descrição: O usuário recebe pedido de liberação de hemocomponente, após realizar os exames do paciente acessa o sistema e escolhe a opção liberação de hemocomponentes, informa ao sistema o número da solicitação ou o registro do paciente e procede com a confirmação dos dados de solicitação e informa ao sistema a liberação do mesmo.

Figura 24. (Descrição de caso de uso)

3.3.9. Descrição de Caso de Uso: Agendar doação

Caso de uso: Agendar doação.

Atores: Doador.

Tipo: Primário.

Descrição: O doador necessita agendar uma doação, acessa a página do sistema em qualquer terminal que possua conexão com a Internet e através da opção doador ele escolhe a opção agendamento de doação onde é informado o nome ou CPF do mesmo e em seguida à validação de sua identificação no sistema segue com o registro do agendamento da doação.

Figura 25. (Descrição de caso de uso)

3.3.10. Descrição de Caso de Uso: Solicitar declaração de doação

Caso de uso: Solicitar declaração de doação.

Atores: Doador.

Tipo: Primário.

Descrição: O doador necessita de uma declaração de doação, acessa a página do sistema em qualquer terminal que possua conexão com a Internet e através da opção doador ele escolhe a opção solicitar declaração de doação onde é informado o nome ou CPF do mesmo e em seguida à validação de sua identificação no sistema segue com o registro da solicitação de declaração de doação.

Figura 26. (Descrição de caso de uso)

3.4. CASOS DE USO EXPANDIDO

3.4.1. Caso de uso expandido: Consultar estoque

Ator Principal: Técnico enfermeiro, médico.

Interessados e Interesses:

- Usuário: deseja a opção de consultar estoque de hemocomponentes.

Pré-condições: O usuário deve ser identificado e autenticado para poder ter acesso ao sistema, limitado ao seu perfil de usuário.

Garantias de sucesso (Pós-condições): A consulta deve obter êxito.

Cenário de Sucesso Principal (ou Fluxo Básico):

1. O usuário seleciona a opção consultar estoque.
2. O sistema informa as opções de consultas.
3. O usuário informa o tipo de consulta que ele deseja.
4. O sistema informa na tela a consulta desejada e oferece a opção de impressão da mesma.

Extensões (ou Fluxos Alternativos):

A. (*) Sistema instável:

Para fornecer suporte para recuperação e a correta estabilização do sistema, garanta que todas as informações de estado sensíveis nas transações e todos os eventos possam ser recuperados a partir de qualquer passo do cenário:

- i. O usuário reinicia o Sistema, registra-se e solicita a recuperação do estado anterior.
- ii. O Sistema restaura o estado anterior
- iii. O Sistema detecta anomalias que impedem a restauração:
 - a. O Sistema avisa ao usuário sobre o erro, registra o erro e, então, entra em um novo estado estabilidade.
- iv. O usuário começa um novo registro.

Requisitos Especiais:

- IU - Interface com o usuário de fácil entendimento por monitor de 15" SVGA com boa visualização a distância de um metro.
- Impressões feitas por impressoras a jato de tinta.
- Respostas rápidas do sistema.
- De alguma forma, queremos seguranças na gravação e recuperação dos dados do sistema.

Lista de Variações Tecnológicas e de Dados:

Nenhuma.

Frequência de Ocorrência:

Poderia ser quase contínuo.

Problemas em Aberto:

- Atualizações das leis vigentes dos serviços de hemoterapia.
- Na utilização do sistema por mais de um usuário no mesmo horário e local.

3.4.2. Caso de uso expandido: Cadastrar paciente

Ator Principal: Técnico enfermeiro, médico.

Interessados e Interesses:

- Usuário: deseja a opção de cadastrar paciente.

Pré-condições: O usuário deve ser identificado e autenticado para poder ter acesso ao sistema, limitado ao seu perfil de usuário.

Garantias de sucesso (Pós-condições): O cadastro do paciente é gravado com sucesso em BD, em caso de negativa, retornar aviso.

Cenário de Sucesso Principal (ou Fluxo Básico):

1. O usuário seleciona a opção cadastrar paciente.
2. O sistema solicita dados do paciente para consulta em BD.
3. O usuário informa nome ou prontuário do paciente.

4. O sistema consulta se o paciente já possui cadastro e solicita os dados do paciente para cadastro.
5. O usuário informa dados do paciente.
6. O sistema registra cadastro do paciente e retorna na tela: "Paciente cadastrado com sucesso!".

Extensões (ou Fluxos Alternativos):

A. (*) Sistema instável:

Para fornecer suporte para recuperação e a correta estabilização do sistema, garanta que todas as informações de estado sensíveis nas transações e todos os eventos possam ser recuperados a partir de qualquer passo do cenário:

- i. O usuário reinicia o Sistema, registra-se e solicita a recuperação do estado anterior.
- ii. O Sistema restaura o estado anterior
- iii. O Sistema detecta anomalias que impedem a restauração:
 - a. O Sistema avisa ao usuário sobre o erro, registra o erro e, então, entra em um novo estado estabilidade.
- iv. O usuário começa um novo registro.

B. (4) Caso o paciente já seja cadastrado o sistema informa ao usuário que o paciente já possui cadastro.

Requisitos Especiais:

- IU - Interface com o usuário de fácil entendimento por monitor de 15" SVGA com boa visualização a distância de um metro.
- Impressões feitas por impressoras a jato de tinta.
- Respostas rápidas do sistema.
- De alguma forma, queremos seguranças na gravação e recuperação dos dados do sistema.

Lista de Variações Tecnológicas e de Dados:

Nenhuma.

Frequência de Ocorrência:

Poderia ser quase contínuo.

Problemas em Aberto:

- Atualizações das leis vigentes dos serviços de hemoterapia.
- Na utilização do sistema por mais de um usuário no mesmo horário e local.

3.4.3. Caso de uso expandido: Cadastrar exames do paciente

Ator Principal: Técnico.Interessados e Interesses:

- Técnico: deseja a opção de cadastrar exames do paciente.

Pré-condições: O usuário deve ser identificado e autenticado para poder ter acesso ao sistema, limitado ao seu perfil de usuário, o paciente já deverá possuir cadastrado no sistema.

Garantias de sucesso (Pós-condições): O cadastro dos exames do paciente é gravado com sucesso em B, em caso de negativa, retornar aviso.

Cenário de Sucesso Principal (ou Fluxo Básico):Lista de Variáveis Técnicas e de Dados:

1. O usuário seleciona a opção cadastrar de exames.
2. O sistema solicita dados do paciente.
3. O usuário informa o nome ou registro do paciente.
4. O sistema verifica se o paciente já possui cadastro.
5. O sistema imprime na tela dados do paciente.
6. O usuário entra com os dados dos exames do paciente.
7. O sistema retorna na tela: "Exames cadastrados com sucesso!".
8. O sistema imprime resultados de exames do paciente.

Extensões (ou Fluxos Alternativos):

3.4.4. Caso de uso expandido: Solicitar backup completo

A. (*) Sistema instável:

Para fornecer suporte para recuperação e a correta estabilização do sistema, garanta que todas as informações de estado sensíveis nas transações e todos os eventos possam ser recuperados a partir de qualquer passo do cenário:

- i. O usuário reinicia o Sistema, registra-se e solicita a recuperação do estado anterior.
- ii. O Sistema restaura o estado anterior

iii. O Sistema detecta anomalias que impedem a restauração:

a. O Sistema avisa ao usuário sobre o erro, registra o erro e, então, entra em um novo estado estabilidade.

iv. O usuário começa um novo registro.

B. (4) Paciente não cadastrado:

O sistema retorna impressão na tela "Paciente não possui cadastro, efetue cadastro do paciente!",

Cenário de Sucesso Principal (ou Fluxo Básico):

Requisitos Especiais:

- IU - Interface com o usuário de fácil entendimento por monitor de 15" SVGA com boa visualização a distância de um metro.
- Impressões feitas por impressoras a jato de tinta.
- Respostas rápidas do sistema.
- De alguma forma, queremos seguranças na gravação e recuperação dos dados do sistema.

Lista de Variações Tecnológicas e de Dados:

Nenhuma.

Frequência de Ocorrência:

Poderia ser quase contínuo.

Problemas em Aberto:

- Atualizações das leis vigentes dos serviços de hemoterapia.
- Na utilização do sistema por mais de um usuário no mesmo horário e local.

3.4.4. Caso de uso expandido: Solicitar hemocomponente

Ator Principal: Técnico enfermeiro, médico.

Interessados e Interesses:

- Técnico: deseja a opção de solicitar hemocomponente (Transfusão de hemocomponente).

Pré-condições: O usuário deve ser identificado e autenticado para poder ter acesso ao sistema, limitado ao seu perfil de usuário.

Garantias de sucesso (Pós-condições): O cadastro da solicitação de hemocomponentes é gravado com sucesso em BD, em caso de negativa, retornar aviso. O cadastro de solicitação de hemocomponentes é gravada com sucesso em BD.

Cenário de Sucesso Principal (ou Fluxo Básico):

1. O usuário seleciona a opção solicitação de hemocomponentes.
2. O sistema solicita nome ou registro do paciente a ser transfundido.
3. O usuário informa os dados do paciente.
4. O sistema verifica dados do paciente.
5. O sistema imprime na tela os dados do paciente.
6. O usuário confere os dados do paciente e informa os dados da solicitação.
7. O sistema imprime os dados da solicitação na tela e solicita confirmação.
8. O sistema retorna na tela: "Operação concluída com êxito, enviar pedido para hospital de destino" e imprime solicitação para ser enviada juntamente com a amostra do paciente.

Extensões (ou Fluxos Alternativos):

A. (*) Sistema instável:

Para fornecer suporte para recuperação e a correta estabilização do sistema, garanta que todas as informações de estado sensíveis nas transações e todos os eventos possam ser recuperados a partir de qualquer passo do cenário:

- i. O usuário reinicia o Sistema, registra-se e solicita a recuperação do estado anterior.
- ii. O Sistema restaura o estado anterior
- iii. O Sistema detecta anomalias que impedem a restauração:
 - a. O Sistema avisa ao usuário sobre o erro, registra o erro e, então, entra em um novo estado estabilidade.

iv. O usuário começa um novo registro.

B. (4) Paciente não cadastrado:

O sistema imprime na tela "Paciente não cadastrado, efetue cadastro do paciente!",

Requisitos Especiais:

- IU - Interface com o usuário de fácil entendimento por monitor de 15" SVGA com boa visualização a distância de um metro.
- Impressões feitas por impressoras a jato de tinta.
- Respostas rápidas do sistema.
- De alguma forma, queremos seguranças na gravação e recuperação dos dados do sistema.

Lista de Variações Tecnológicas e de Dados:

Nenhuma.

Frequência de Ocorrência:

Poderia ser quase contínuo.

Problemas em Aberto:

- Atualizações das leis vigentes dos serviços de hemoterapia.
- Na utilização do sistema por mais de um usuário no mesmo horário e local.

3.4.5. Caso de uso expandido: Liberação de hemocomponentes

Ator Principal: Técnico.Interessados e Interesses:

- Técnico: deseja a opção de liberar hemocomponente (Transusão de hemocomponente).

Pré-condições: O usuário deve ser identificado e autenticado para poder ter acesso ao sistema, limitado ao seu perfil de usuário, o paciente deverá possuir cadastrado e solicitação devidamente registrada no sistema, bem como os exames pré-transfusionais.

Garantias de sucesso (Pós-condições): O cadastro e liberação de hemocomponentes é gravado com sucesso em BD, em caso de negativa, retornar aviso, o cadastro de liberação de hemocomponentes é gravada com sucesso em BD.

Lista de Variações Tecnológicas e de Dados:

Cenário de Sucesso Principal (ou Fluxo Básico):

1. O usuário seleciona a opção liberação de hemocomponentes.
2. O sistema solicita dados da solicitação hemocomponentes.
3. O usuário informa o número da solicitação hemocomponentes.
4. O sistema verifica dados da solicitação hemocomponentes.
5. O sistema imprime na tela a solicitação de hemocomponentes.
6. O usuário confirma a liberação da solicitação de hemocomponentes.
7. O sistema retorna "Operação concluída com êxito, enviar pedido para hospital de destino".

3.4.6. Caso de uso expandido: Solicitar transferência de estoque

Extensões (ou Fluxos Alternativos):

Ater Principal Técnico

A. (*) Sistema instável:

Para fornecer suporte para recuperação e a correta estabilização do sistema, garanta que todas as informações de estado sensíveis nas transações e todos os eventos possam ser recuperados a partir de qualquer passo do cenário:

- i. O usuário reinicia o Sistema, registra-se e solicita a recuperação do estado anterior.
- ii. O Sistema restaura o estado anterior
- iii. O Sistema detecta anomalias que impedem a restauração:
 - a. O Sistema avisa ao usuário sobre o erro, registra o erro e, então, entra em um novo estado estabilidade.
- iv. O usuário começa um novo registro.

B. (4) Solicitação não cadastrada:

O sistema retorna impressão na tela "Solicitação inexistente, deseja cadastrar solicitação?", uma solicitação de dados para transferência de estoque.

Requisitos Especiais:

- IU - Interface com o usuário de fácil entendimento por monitor de 15" SVGA com boa visualização a distância de um metro.
- Impressões feitas por impressoras a jato de tinta.
- Respostas rápidas do sistema.
- De alguma forma, queremos seguranças na gravação e recuperação dos dados do sistema.

A (*) Sistema instável:

Lista de Variações Tecnológicas e de Dados:

Nenhuma.

Frequência de Ocorrência:

Poderia ser quase contínuo.

Problemas em Aberto:

- Atualizações das leis vigentes dos serviços de hemoterapia.
- Na utilização do sistema por mais de um usuário no mesmo horário e local.

3.4.6. Caso de uso expandido: Solicitar transferência de estoque

Ator Principal: Técnico.

Interessados e Interesses:

- Técnico: deseja a opção de solicitar transferência de estoque de hemocomponentes.

Pré-condições: O usuário deve ser identificado e autenticado para poder ter acesso ao sistema, limitado ao seu perfil de usuário.

Garantias de sucesso (Pós-condições): A solicitação de transferência de estoque é gravada com sucesso em BD, em caso de negativa, retornar aviso.

Cenário de Sucesso Principal (ou Fluxo Básico):

1. O usuário seleciona a opção de solicitação de transferência de estoque.
2. O sistema solicita os dados para transferência de estoque.
3. O usuário informa quantidade, grupo sanguíneo e tipo de hemocomponente a ser solicitado.
4. O sistema registra os dados da solicitação.
5. O sistema imprime na tela os dados da solicitação.

Extensões (ou Fluxos Alternativos):

A. (*) Sistema instável:

Para fornecer suporte para recuperação e a correta estabilização do sistema, garanta que todas as informações de estado sensíveis nas transações e todos os eventos possam ser recuperados a partir de qualquer passo do cenário:

- i. O usuário reinicia o Sistema, registra-se e solicita a recuperação do estado anterior.
- ii. O Sistema restaura o estado anterior
- iii. O Sistema detecta anomalias que impedem a restauração:
 - a. O Sistema avisa ao usuário sobre o erro, registra o erro e, então, entra em um novo estado estabilidade.
- iv. O usuário começa um novo registro.

Requisitos Especiais:

- IU - Interface com o usuário de fácil entendimento por monitor de 15" SVGA com boa visualização a distância de um metro.
- Impressões feitas por impressoras a jato de tinta.
- Respostas rápidas do sistema.
- De alguma forma, queremos seguranças na gravação e recuperação dos dados do sistema.

Lista de Variações Tecnológicas e de Dados:

Nenhuma.

Frequência de Ocorrência:

Poderia ser quase contínuo.

Problemas em Aberto:

- Atualizações das leis vigentes dos serviços de hemoterapia.
- Na utilização do sistema por mais de um usuário no mesmo horário e local.

3.4.7. Caso de uso expandido: Transferir estoque

Ator Principal: Técnico.

Interessados e Interesses:

- Técnico: deseja a opção de transferir estoque de hemocomponentes.

Pré-condições: O usuário deve ser identificado e autenticado para poder ter acesso ao sistema, limitado ao seu perfil de usuário.

Garantias de sucesso (Pós-condições): A transferência de estoque é gravada com sucesso em BD, em caso de negativa, retornar aviso.

Cenário de Sucesso Principal (ou Fluxo Básico):

- Atualização das informações dos serviços de hemoterapia.

1. O usuário seleciona a opção transferência de estoque.
2. O sistema solicita os números dos hemocomponentes a serem transferidos.
3. O usuário informa os números dos hemocomponentes a serem transferidos.
4. O sistema registra os dados da transferência.
5. O sistema imprime na tela os dados da transferência.

- Fluxo Principal: Doador.

Extensões (ou Fluxos Alternativos):

- Doador: deseja a opção de solicitar agendamento de doação.

A. (*) Sistema instável:

Para fornecer suporte para recuperação e a correta estabilização do sistema, garanta que todas as informações de estado sensíveis nas transações e todos os eventos possam ser recuperados a partir de qualquer passo do cenário:

- i. O usuário reinicia o Sistema, registra-se e solicita a recuperação do estado anterior.
- ii. O Sistema restaura o estado anterior
- iii. O Sistema detecta anomalias que impedem a restauração:
 - a. O Sistema avisa ao usuário sobre o erro, registra o erro e, então, entra em um novo estado estabilidade.
- iv. O usuário começa um novo registro.

2. O sistema solicita os dados do doador, através, nome ou CPF.

Requisitos Especiais:

- IU - Interface com o usuário de fácil entendimento por monitor de 15" SVGA com boa visualização a distância de um metro.
- Impressões feitas por impressoras a jato de tinta.
- Respostas rápidas do sistema.
- De alguma forma, queremos seguranças na gravação e recuperação dos dados do sistema.

Lista de Variações Tecnológicas e de Dados:

Entrada de dados poderá ser por teclado ou por leitura ótica (código de barras).

A. (*) Sistema instável;

Frequência de Ocorrência:

Poderia ser quase contínuo.

Problemas em Aberto:

- Atualizações das leis vigentes dos serviços de hemoterapia.
- Na utilização do sistema por mais de um usuário no mesmo horário e local.

3.4.8. Caso de uso expandido: Agendar doação

Ator Principal: Doador.

Interessados e Interesses:

- Doador: deseja a opção de solicitar agendamento de doação.

Pré-condições: O doador deve possuir cadastro no sistema para poder agendar doação.

Garantias de sucesso (Pós-condições): A solicitação de agendamento de doação é gravada com sucesso em BD, em caso de negativa, retornar aviso.

Cenário de Sucesso Principal (ou Fluxo Básico):

Lista de Variações Tecnológicas e de Dados:

1. O doador acessa o sistema via terminal Web, seleciona a opção de solicitação agendamento de doação.
2. O sistema solicita os dados do doador, como: nome ou CPF.
3. O doador informa dados solicitados.
4. O sistema verifica se o doador já possui cadastro.
5. O sistema imprime na tela os dados do doador e solicita dados de agendamento.
6. O doador informa dados solicitados.
7. O sistema imprime na tela: "Agendamento efetuado com sucesso!".

3.4.9 Extensões (ou Fluxos Alternativos):

A. (*) Sistema instável:

Para fornecer suporte para recuperação e a correta estabilização do sistema, garanta que todas as informações de estado sensíveis nas transações e todos os eventos possam ser recuperados a partir de qualquer passo do cenário:

- i. O usuário reinicia o Sistema, registra-se e solicita a recuperação do estado anterior.
- ii. O Sistema restaura o estado anterior
- iii. O Sistema detecta anomalias que impedem a restauração:

- a. O Sistema avisa ao usuário sobre o erro, registra o erro e, então, entra em um novo estado estabilidade.

- iv. O usuário começa um novo registro.

B.(4) Doador não cadastrado.

O sistema retorna impressão na tela "Doador não possui cadastro em nosso sistema!".

Requisitos Especiais:

- IU - Interface com o usuário de fácil entendimento por monitor de 15" SVGA com boa visualização a distância de um metro.
- Impressões feitas por impressoras a jato de tinta.
- Respostas rápidas do sistema.
- De alguma forma, queremos seguranças na gravação e recuperação dos dados do sistema.

Lista de Variações Tecnológicas e de Dados:

Nenhuma.

Frequência de Ocorrência:

Poderia ser diário.

A. (*) Sistema instável:

Problemas em Aberto:

- Atualizações das leis vigentes dos serviços de hemoterapia.
- Na utilização do sistema por mais de um usuário no mesmo horário e local.

- i. O usuário reinicia o Sistema, registra-se e solicita a recuperação do estado anterior.

- ii. O Sistema restaura o estado anterior.

3.4.9. Caso de uso expandido: Solicitar declaração de doação

Ator Principal: Doador.

Interessados e Interesses:

- Doador: deseja a opção de solicitar declaração de doação.

Pré-condições: O doador deve possuir cadastro no sistema e ter efetuado no mínimo três doações no período de 12 meses.

Garantias de sucesso (Pós-condições): A solicitação de declaração de doação é gravada com sucesso em BD, em caso de negativa, retornar aviso.

Cenário de Sucesso Principal (ou Fluxo Básico):

1. O doador acessa o sistema via terminal Web, seleciona a opção de solicitação de declaração de doação.
2. O sistema solicita os dados do doador, como: nome ou CPF.
3. O doador informa dados solicitados.
4. O sistema verifica se o doador já possui cadastro e se possui três doações no período de doze meses.
5. O sistema imprime na tela os dados do doador e solicita confirmação do mesmo.
6. O doador confirma os dados.
7. O sistema imprime na tela: "Solicitação efetuada com sucesso, compareça ao banco de sangue em dois dias úteis para receber sua declaração!".

Extensões (ou Fluxos Alternativos):

A. (*) Sistema instável:

Para fornecer suporte para recuperação e a correta estabilização do sistema, garanta que todas as informações de estado sensíveis nas transações e todos os eventos possam ser recuperados a partir de qualquer passo do cenário:

- i. O usuário reinicia o Sistema, registra-se e solicita a recuperação do estado anterior.
- ii. O Sistema restaura o estado anterior

iii. O Sistema detecta anomalias que impedem a restauração:

a. O Sistema avisa ao usuário sobre o erro, registra o erro e, então, entra em um novo estado estabilidade.

iv. O usuário começa um novo registro.

B.(4) Doador não cadastrado ou não possui três doações no período de dose meses.

O sistema retorna impressão na tela "Doador não possui cadastro em nosso sistema!" ou "Doador não possui doações suficiente no período!".

Requisitos Especiais:

- IU - Interface com o usuário de fácil entendimento por monitor de 15' SVGA com boa visualização a distância de um metro.
- Impressões feitas por impressoras a jato de tinta.
- Respostas rápidas do sistema.
- De alguma forma, queremos seguranças na gravação e recuperação dos dados do sistema.

Lista de Variações Tecnológicas e de Dados:

Nenhuma.

Frequência de Ocorrência:

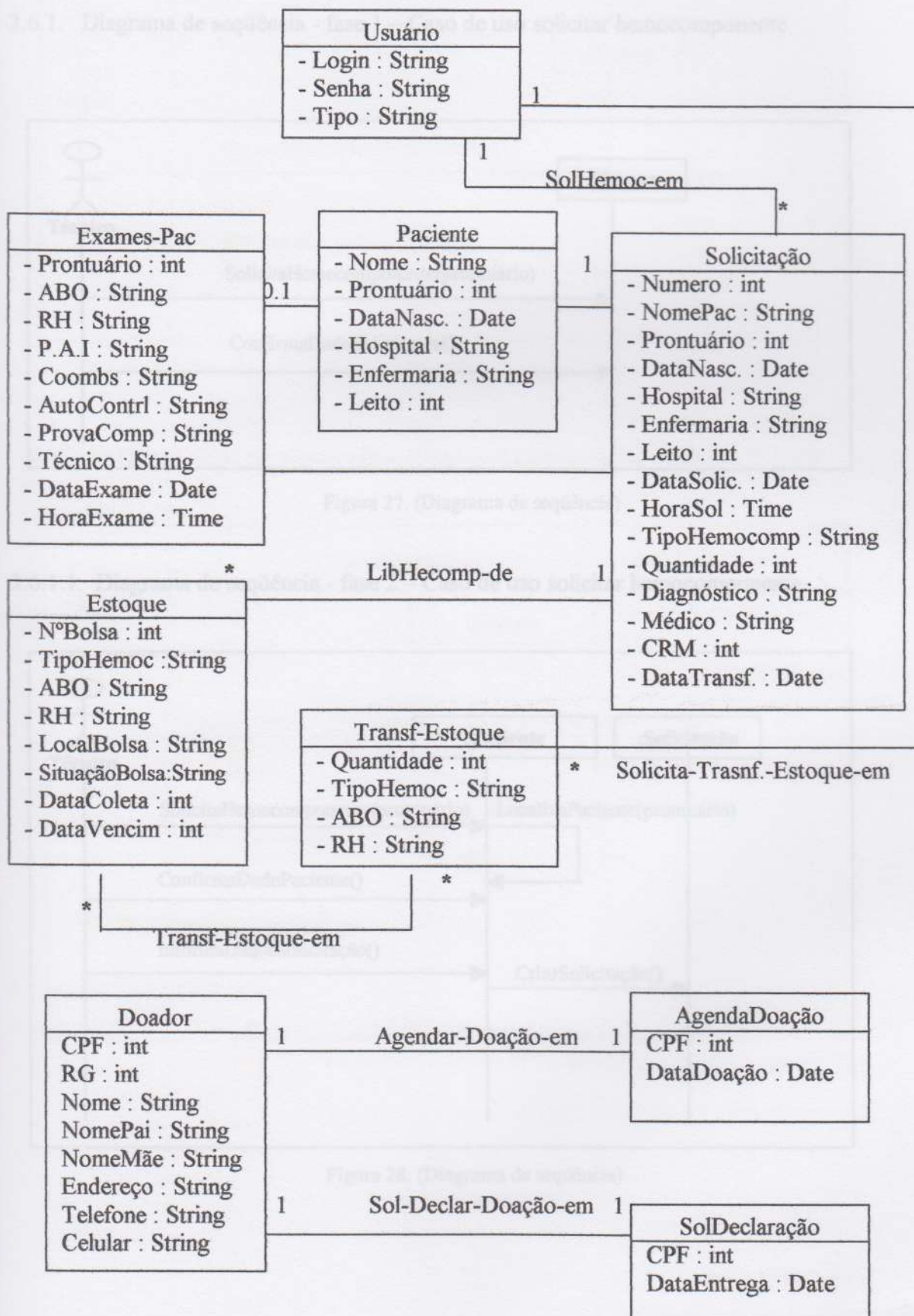
Poderia ser diário.

Problemas em Aberto:

- Atualizações das leis vigentes dos serviços de hemoterapia.
- Na utilização do sistema por mais de um usuário no mesmo horário e local.

3.5. MODELO DE DOMÍNIO

3.5.1. Modelo de Domínio do sistema



3.6. DIAGRAMAS DE SEQUÊNCIA DO SISTEMA

3.6.1. Diagrama de sequência - fase 1 – Caso de uso solicitar hemocomponente.

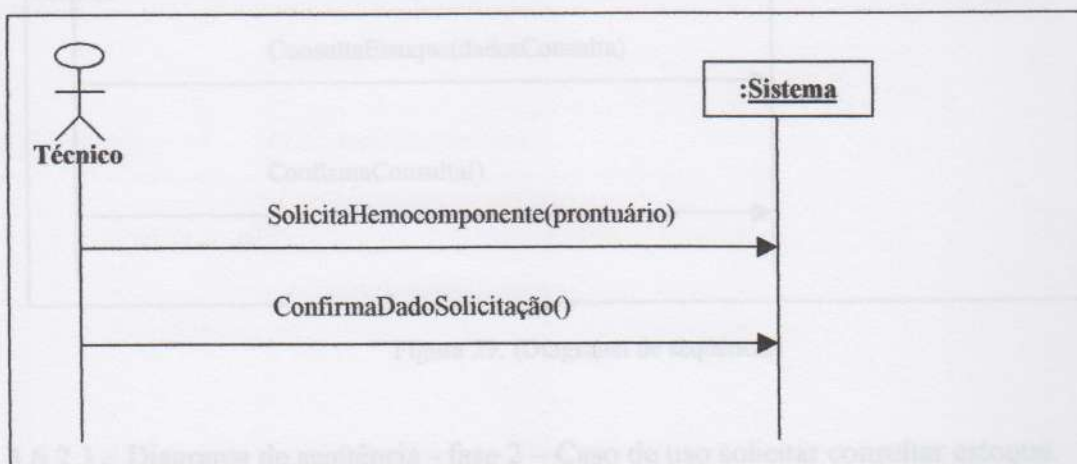


Figura 27. (Diagrama de sequência)

3.6.1.1. Diagrama de sequência - fase 2 – Caso de uso solicitar hemocomponente.

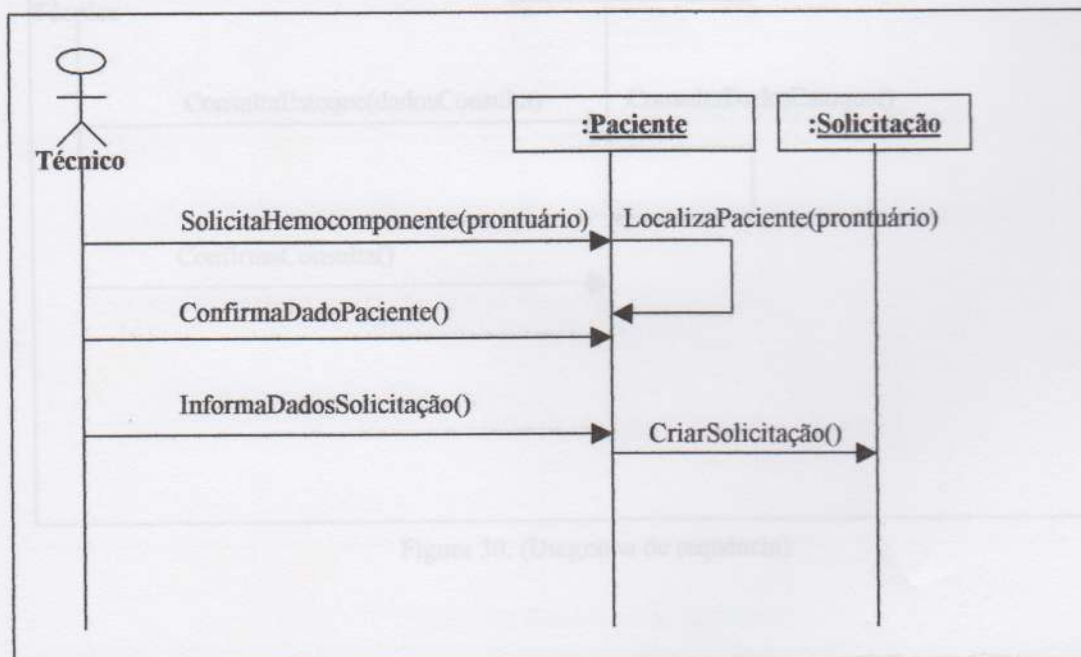


Figura 28. (Diagrama de sequência)

3.6.2. Diagrama de seqüência - fase 1 – Caso de uso consultar estoque.

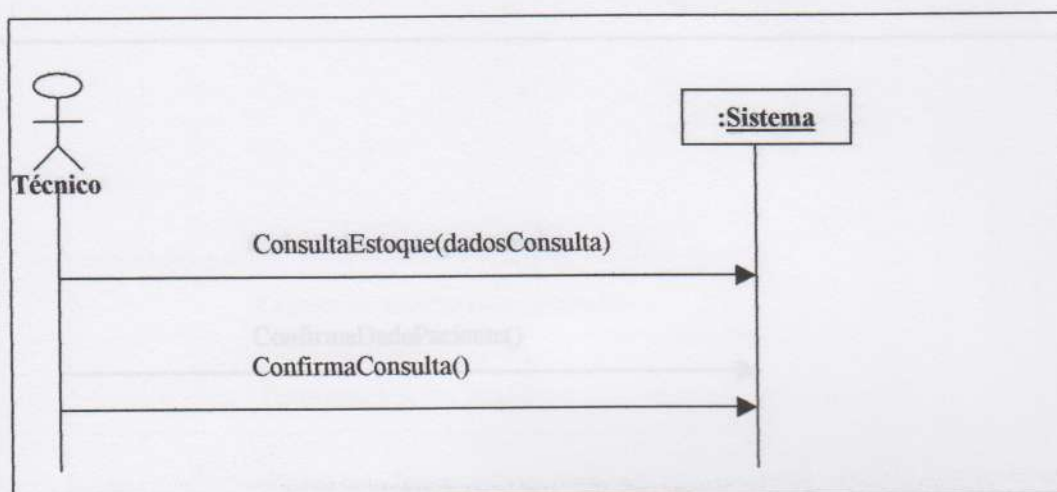


Figura 29. (Diagrama de seqüência)

3.6.2.1. Diagrama de seqüência - fase 2 – Caso de uso solicitar consultar estoque.

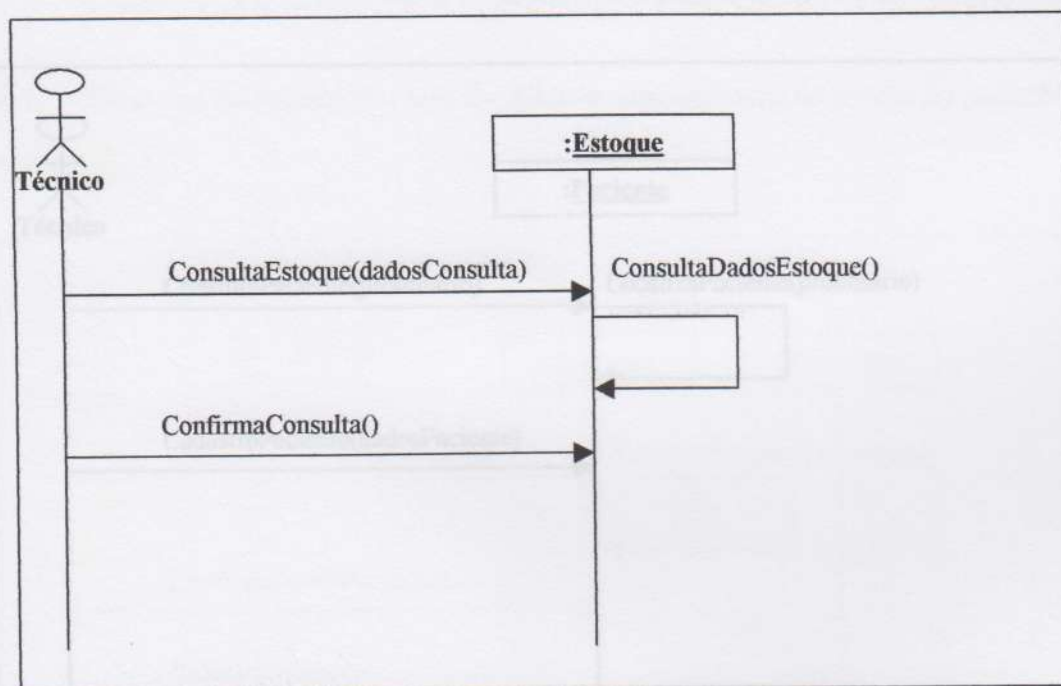


Figura 30. (Diagrama de seqüência)

3.6.3. Diagrama de seqüência - fase 1 – Caso de uso cadastro de paciente.

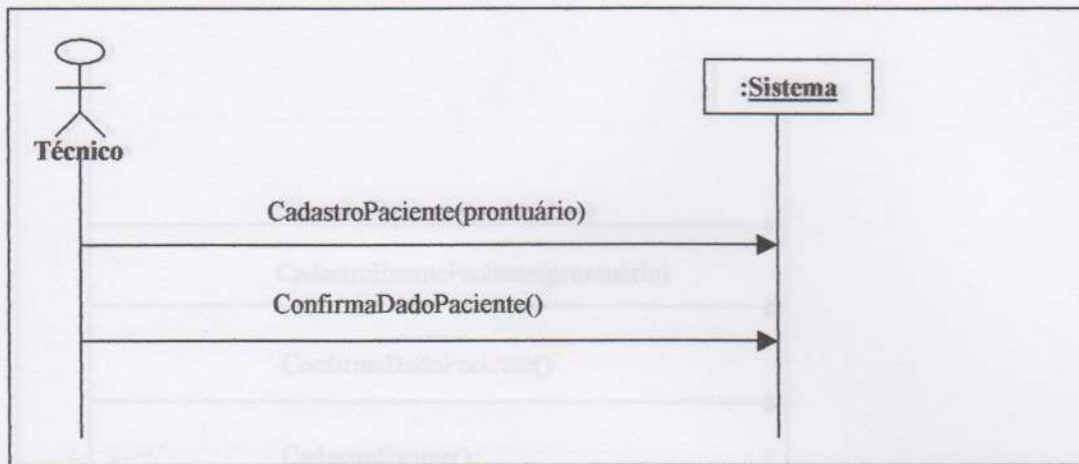


Figura 31. (Diagrama de seqüência)

3.6.3.1. Diagrama de seqüência - fase 2 – Caso de uso cadastro de paciente.

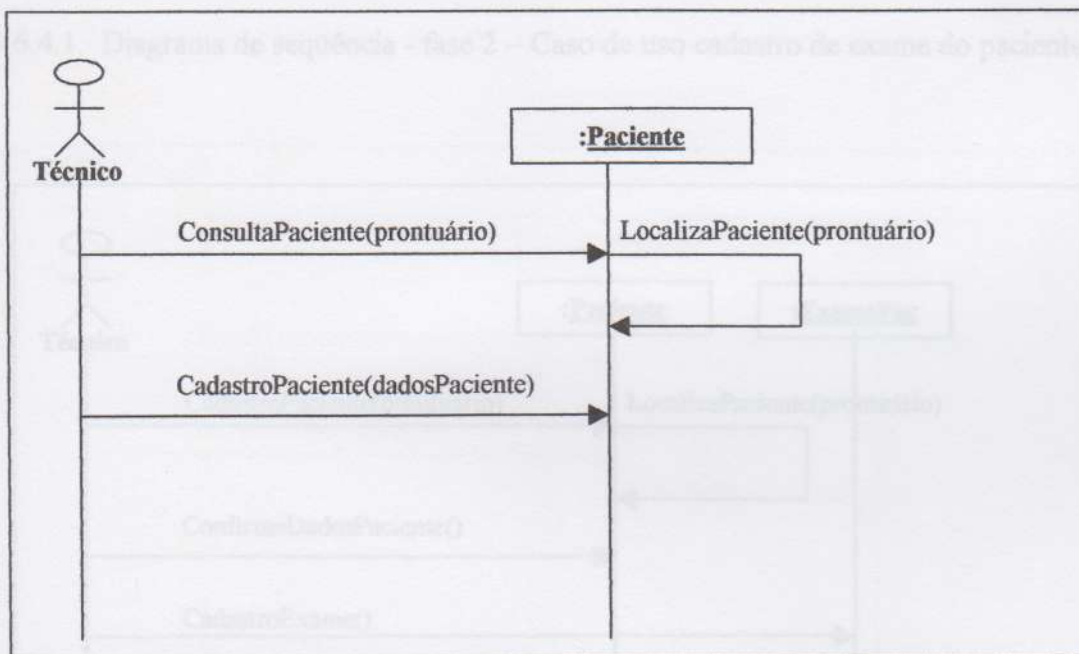


Figura 32. (Diagrama de seqüência)

3.6.4. Diagrama de seqüência - fase 1 – Caso de uso cadastro de exames do paciente.

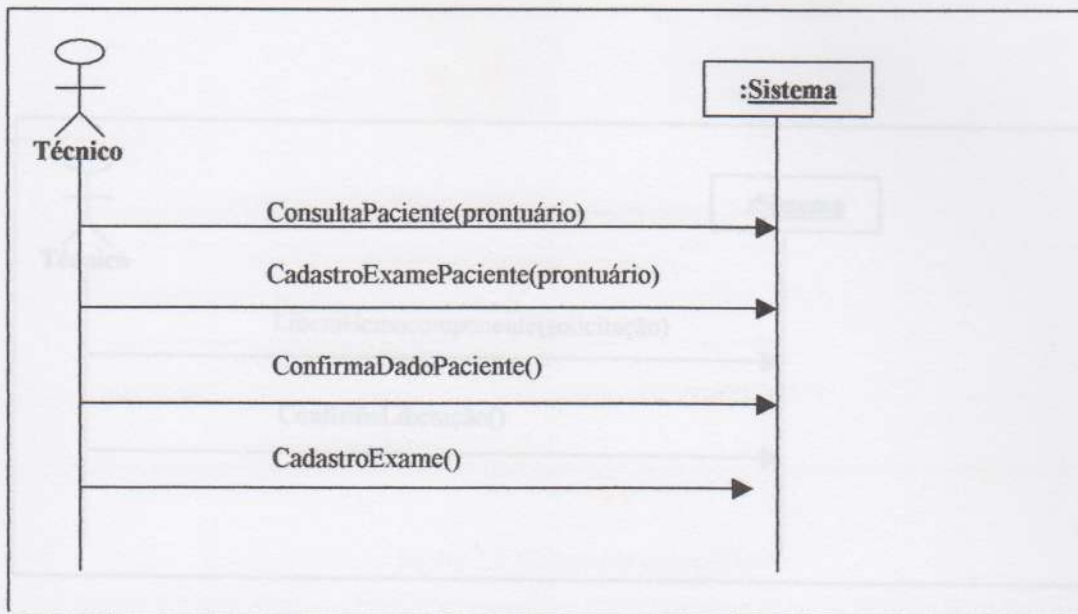


Figura 33. (Diagrama de seqüência)

3.6.4.1. Diagrama de seqüência - fase 2 – Caso de uso cadastro de exame do paciente.

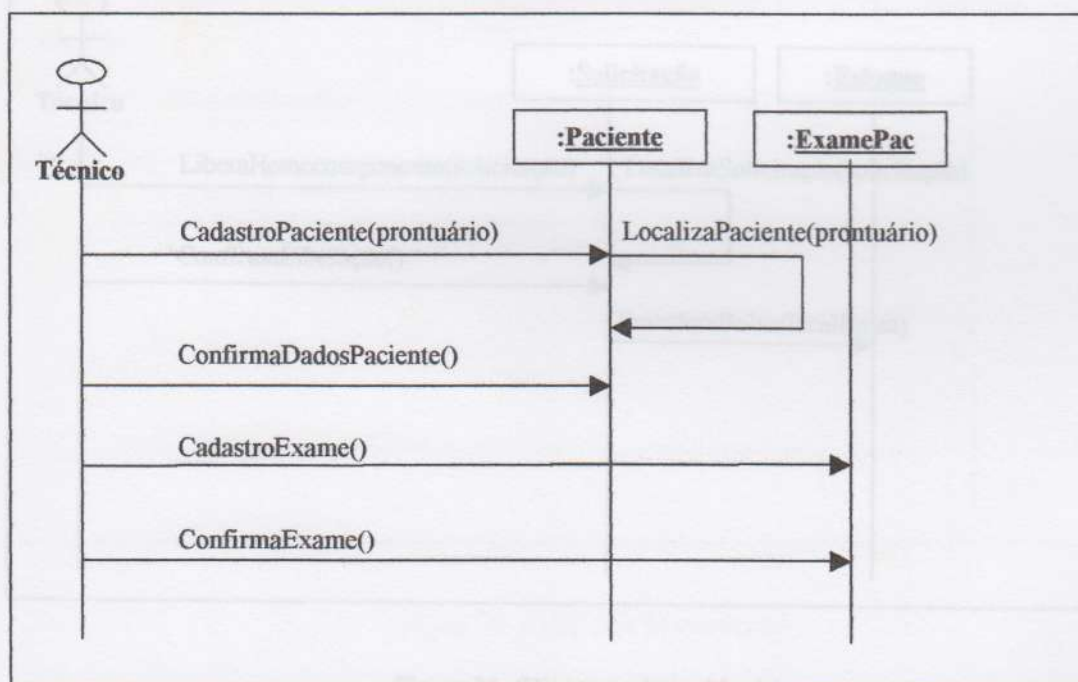


Figura 34. (Diagrama de seqüência)

3.6.5. Diagrama de seqüência - fase 2 – Caso de uso liberar hemocomponente.

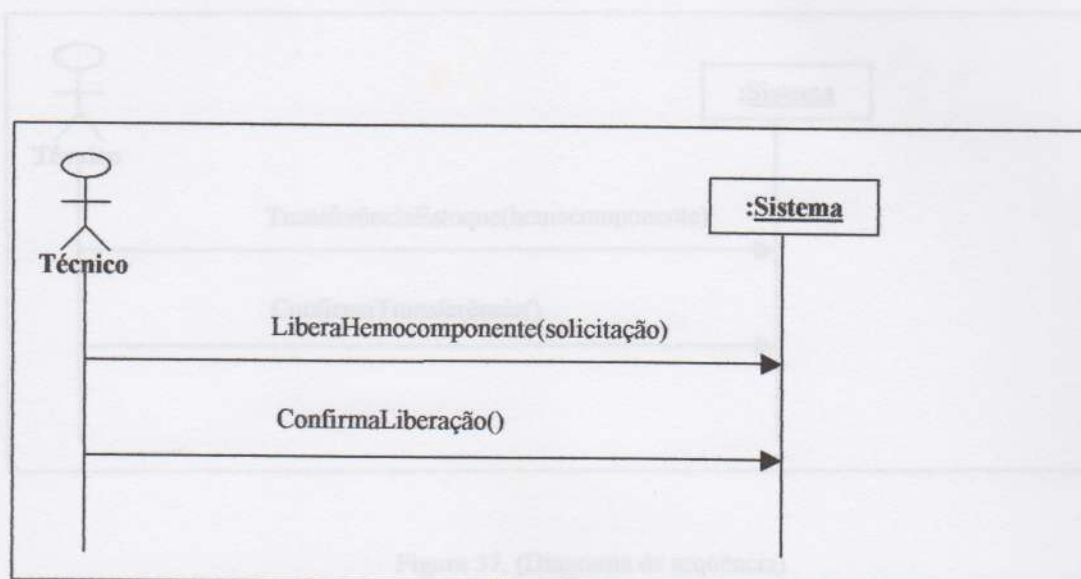


Figura 35. (Diagrama de seqüência)

3.6.5.1. Diagrama de seqüência - fase 2 – Caso de uso liberar hemocomponente.

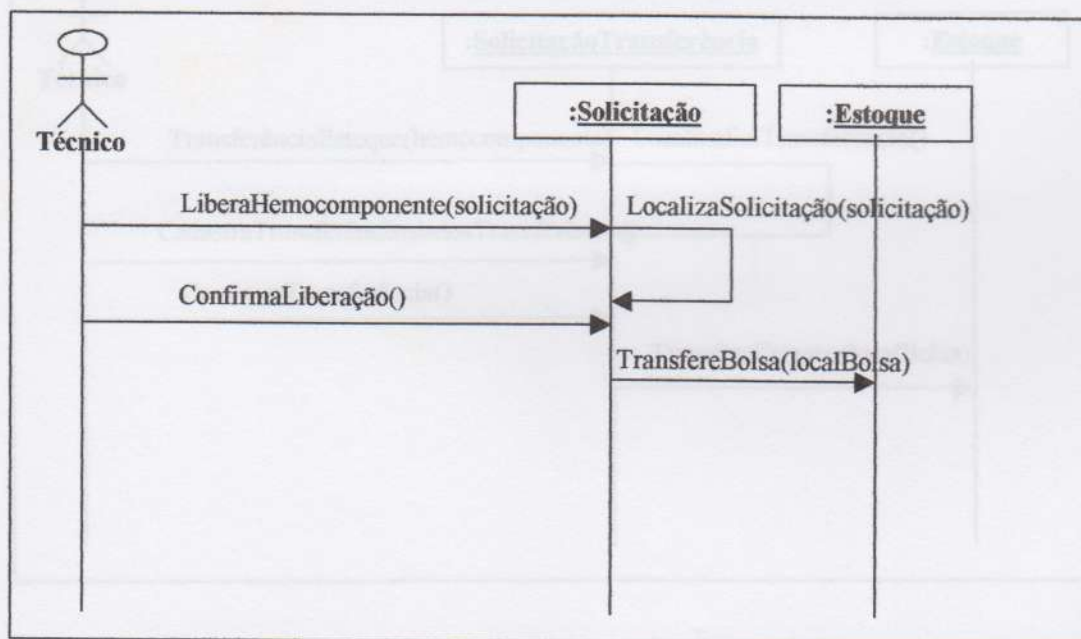


Figura 36. (Diagrama de seqüência)

3.6.6. Diagrama de seqüência - fase 1 – Caso de uso transferir estoque.

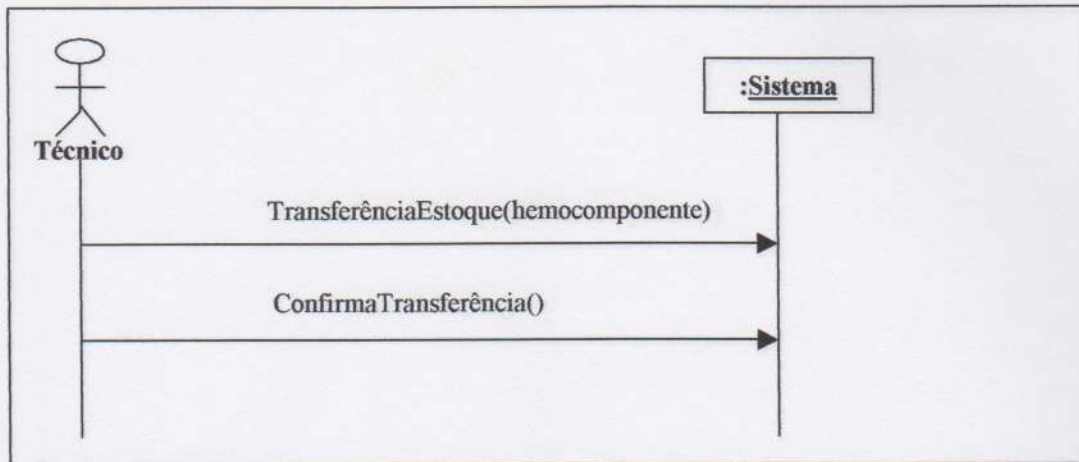


Figura 37. (Diagrama de seqüência)

3.6.6.1. Diagrama de seqüência - fase 2 – Caso de uso transferir estoque.

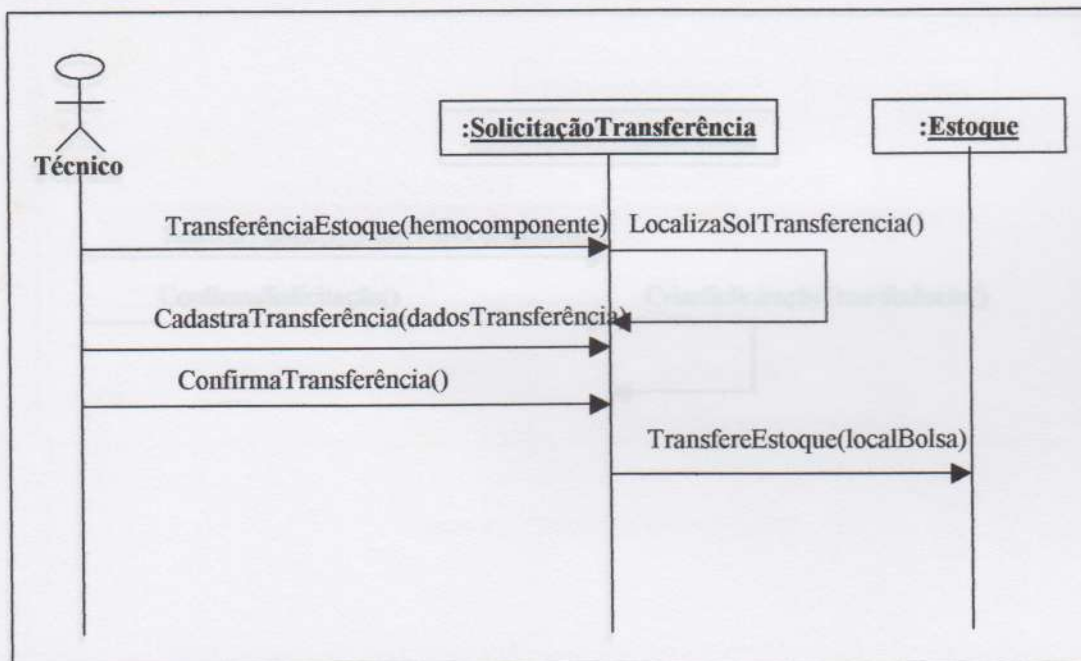


Figura 38. (Diagrama de seqüência)

3.6.7. Diagrama de seqüência - fase 1 – Caso de uso solicitar transferência de estoque.

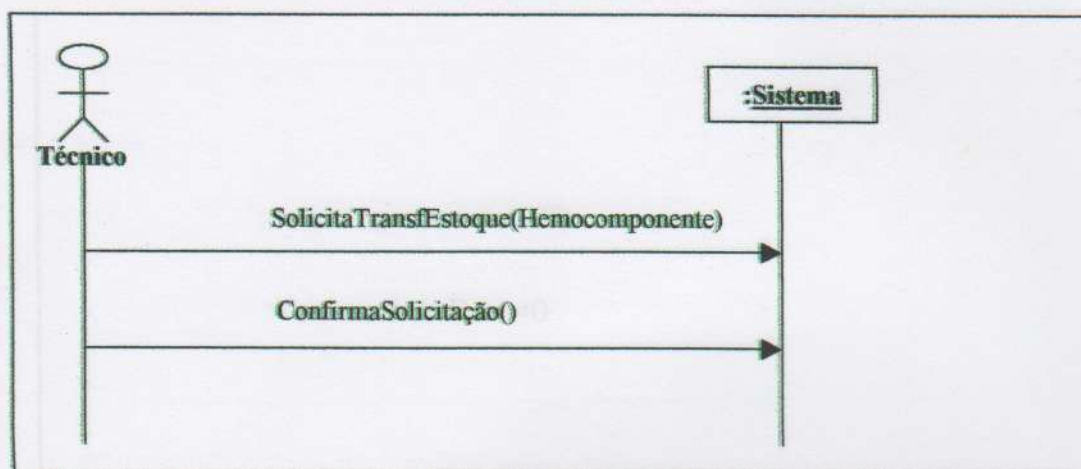


Figura 39. (Diagrama de seqüência)

3.6.7.1. Diagrama de seqüência - fase 2 – Caso de uso solicitar transferência de estoque.

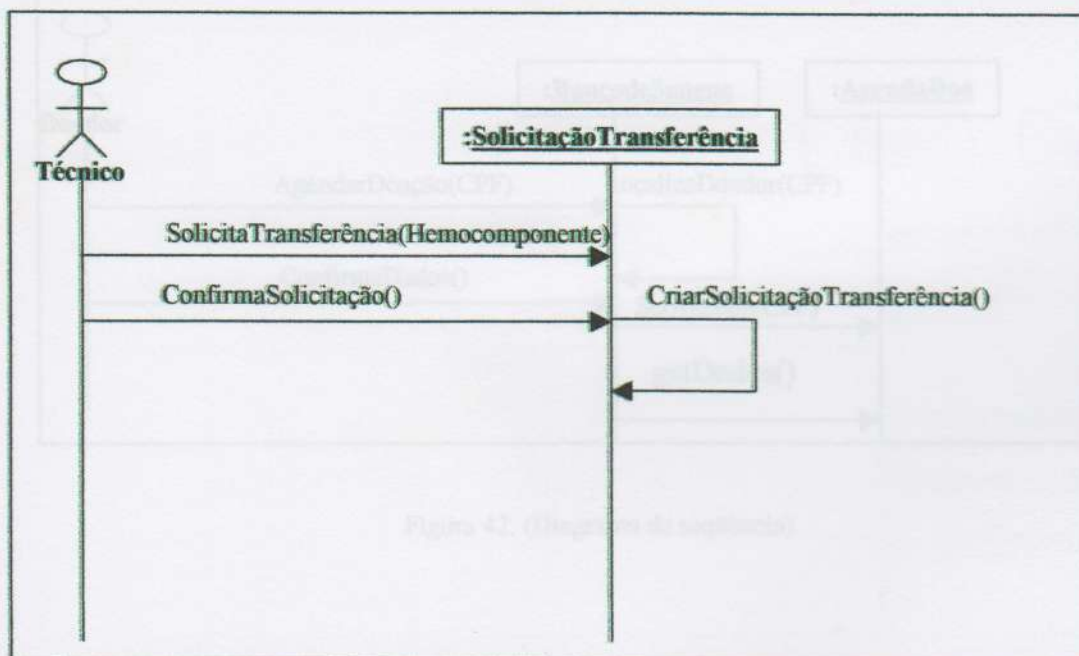


Figura 40. (Diagrama de seqüência)

3.6.8. Diagrama de seqüência - fase 1 – Caso de uso agendar doação.

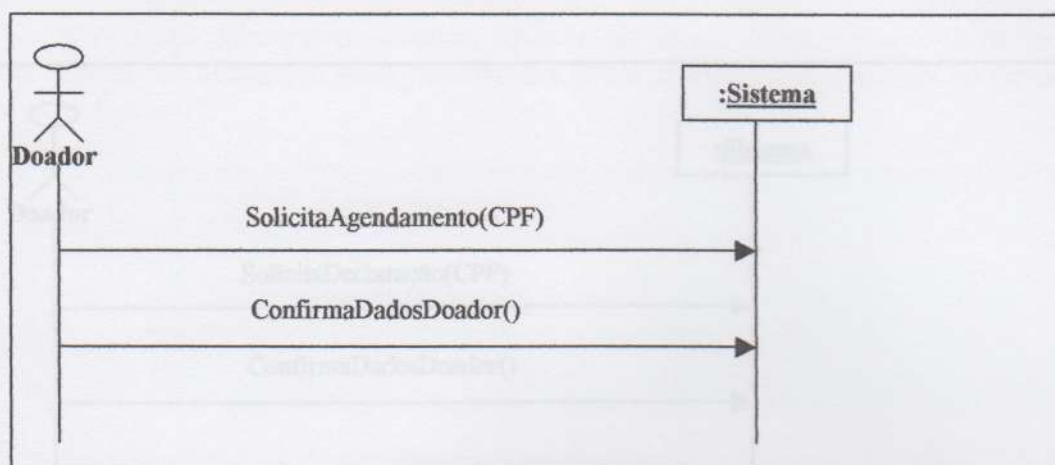


Figura 41. (Diagrama de seqüência)

3.6.8.1. Diagrama de seqüência - fase 2 – Caso de uso agendar doação.

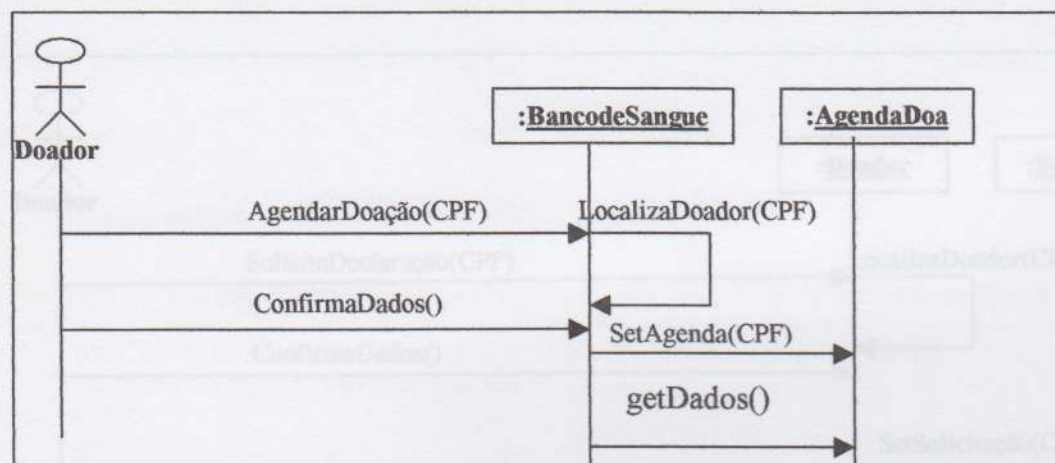


Figura 42. (Diagrama de seqüência)

3.6.9. Diagrama de seqüência - fase 1 – Caso de uso solicitar declaração de doação.

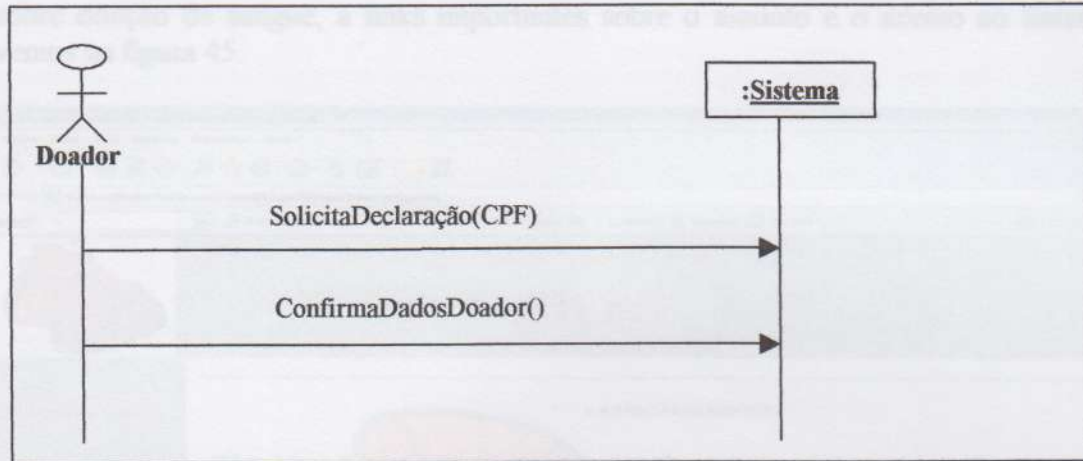


Figura 43. (Diagrama de seqüência)

3.6.9.1. Diagrama de seqüência - fase 2 – Caso de uso solicitar declaração de doação.

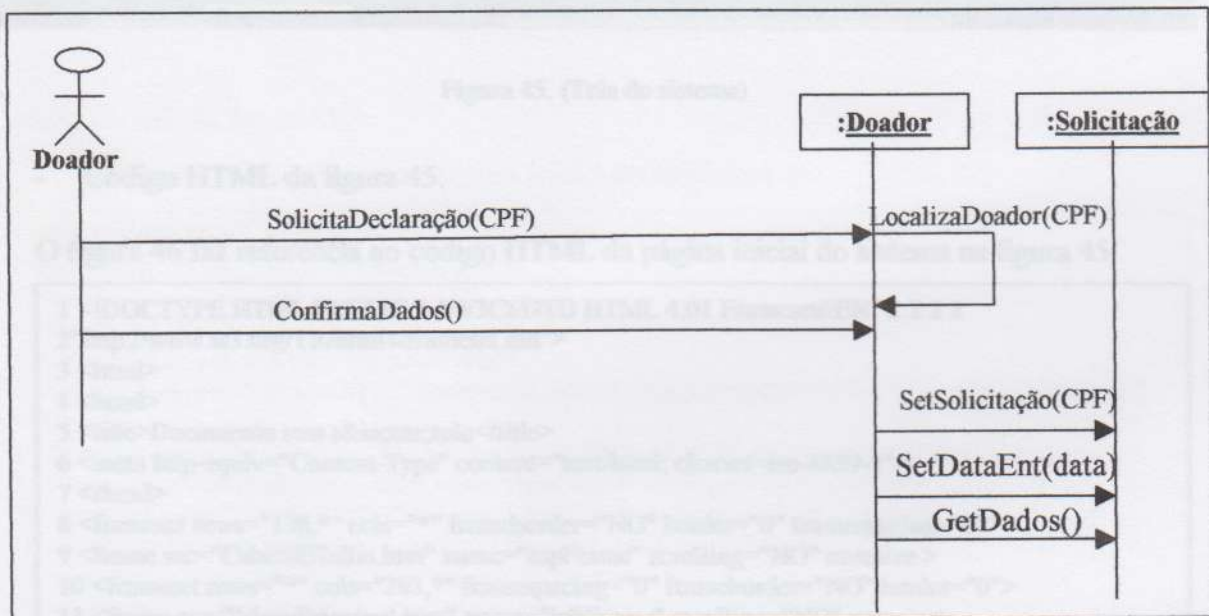


Figura 44. (Diagrama de seqüência)

3.7. TELAS DO SISTEMA

3.7.1. Tela do index principal do sistema

Na página principal do sistema, através do menu, o usuário dispõe de informações sobre doação de sangue, a links importantes sobre o assunto e o acesso ao sistema, como vemos na figura 45.

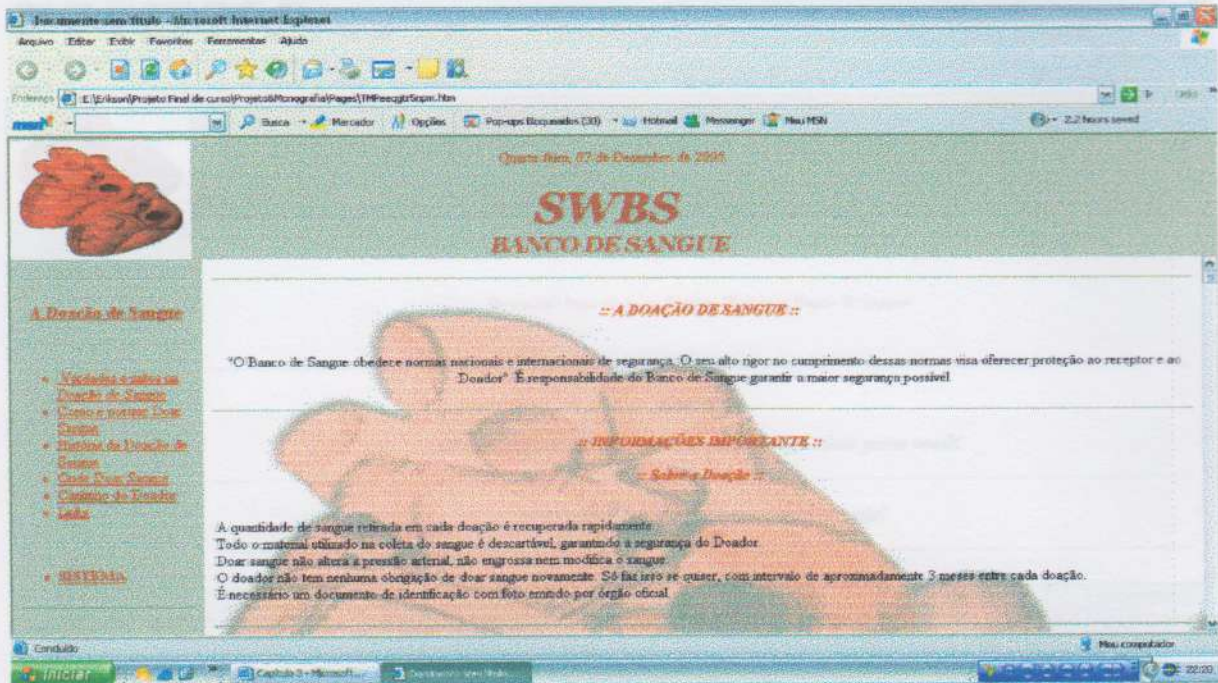


Figura 45. (Tela do sistema)

- Código HTML da figura 45:

O figura 46 faz referência ao código HTML da página inicial do sistema na figura 45.

```
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" 2 2 2 2
2 "http://www.w3.org/TR/html4/frameset.dtd">
3 <html>
4 <head>
5 <title>Documento sem título</title>
6 <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
7 </head>
8 <frameset rows="138,*" cols="*" frameborder="NO" border="0" framespacing="0">
9 <frame src="Cabe%7alho.htm" name="topFrame" scrolling="NO" noresize >
10 <frameset rows="*" cols="203,*" framespacing="0" frameborder="NO" border="0">
11 <frame src="MenuPrincipal.htm" name="leftFrame" scrolling="NO" noresize>
12 <frame src="Conteudo.htm" name="mainFrame">
13 </frameset>
14 </frameset>
15 <noframes><body>
16 </body></noframes>
17 </html>
```

Figura 46. (Código HTML)

3.7.2. Tela do módulo doador

Na página do módulo doador (figura 47) o acesso é feito através do menu inicial na página principal do sistema necessitando apenas da identificação do usuário (doador) pelo CPF e nela o doador dispõe de serviços como agendamento de doação e solicitação de declaração de doação.

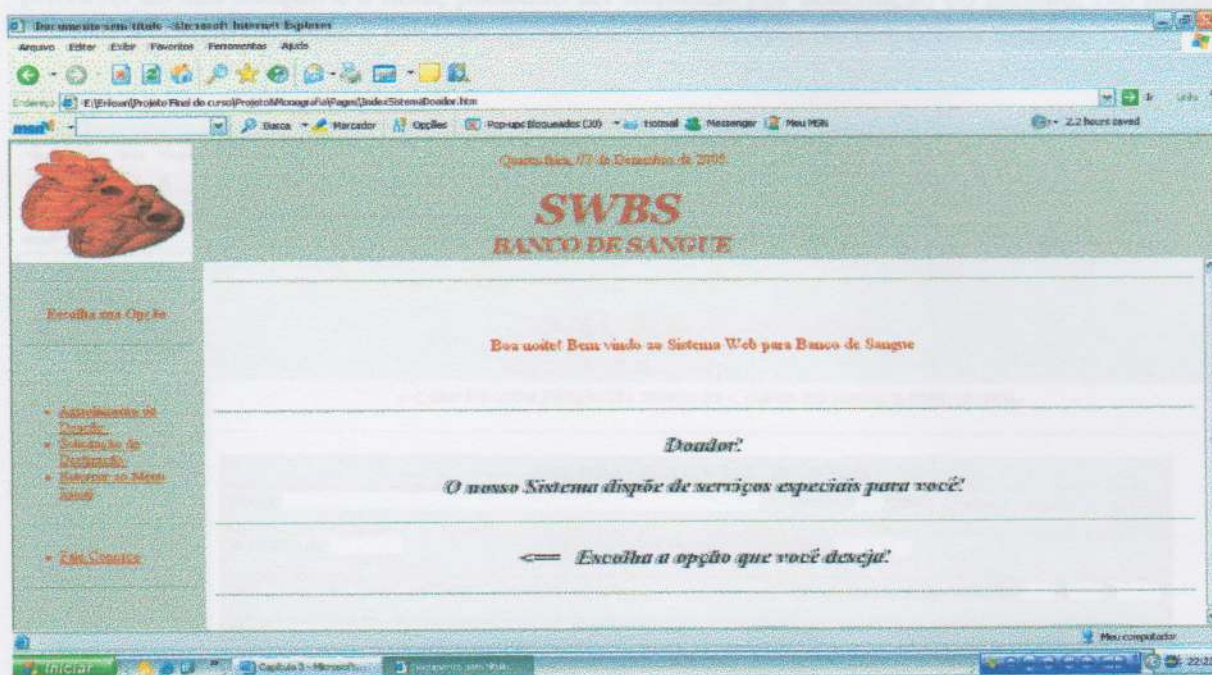


Figura 47. (Tela do sistema)

Código HTML da figura 47:

A figura 48 faz referência a página inicial do módulo doador (figura 47).

```

1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN"
2 "http://www.w3.org/TR/html4/frameset.dtd">
3 <html>
4 <head>
5 <title>Documento sem título</title>
6 <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
7 </head>
8 <frameset rows="140,*" cols="*" frameborder="NO" border="0" framespacing="0">
9 <frame src="UntitledFrame-6.htm" name="topFrame" scrolling="NO" noresize >
10 <frameset rows="*" cols="203,*" framespacing="0" frameborder="NO" border="0">
11 <frame src="MenuDoador.htm" name="leftFrame" scrolling="NO" noresize>
12 <frame src="ConteudoDoador.htm" name="mainFrame">
13 </frameset>
14 </frameset>
15 <noframes><body>
16 </body></noframes>
17 </html>

```

Figura 48. (Código HTML)

3.7.3. Tela solicitação agendamento de doação

Na figura abaixo vemos o agendamento de doação que poderá ser feito pelo usuário (doador), seu acesso é feito através do menu na lateral esquerda da página (figura 49). Após o usuário preencher e confirmar o formulário de agendamento será impresso na tela a data provável de sua doação e a hora, podendo ser impresso por solicitação do mesmo. No mesmo instante a central receberá o aviso de agendamento de doação do respectivo doador.

Documentos sem título - Microsoft Internet Explorer

Arquivo Editar Exibir Favoritos Ferramentas Ajuda

Endereço: E:\Trabalho\Projeto Final de curso\Projeto\Monografia\Pages\Index\IndexDoador.htm

SWBS
BANCO DE SANGUE

Caro DOADOR PREENCHA TODOS OS CAMPOS DO FORMULÁRIO ABAIXO:

AGENDAMENTO DE DOAÇÃO		
NOME	IDADE	SEXO ☐ NENHO ☐ Masc. ☐ Fem.
IDENTIDADE	CPF	
ARG ☐ A ☐ B ☐ AB ☐ O	RH ☐ Positivo ☐ Negativo	DATA PREVISTA DA DOAÇÃO / /
RUA	Nº	
BAIRRO	CORRESPONDENTE	UF
TELEFONE	CELULAR	EMAIL

Concluído

Meu computador

Figura 49. (Tela do sistema)

Código HTML da figura 49:

Na figura 50 encontra-se o código HTML do formulário de agendamento de doação, onde observamos na linha 7 a utilização de tabela pelas tags `<table>` passando todos os parâmetros da tabela como espessura de linha e colunas e cores da mesma e dentro dela o uso de formulários, como vêm na linha 12 pela tag `<form>` e seus métodos `post`, linha 12, `<input>` e a utilização de campos de múltipla escolha `<radio>` na linha 19 entre outros comando HTML para preenchimento do formulário.

Figura 50. (Código HTML)

```

1 <html><head>
2 <title>Documento sem título</title>
3 <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"> </head>
4 <body background="file:///E:/Erikson/Projeto%20Final%20de%20curso/ProjetoFinal/Image/cora%E7%E3o_2.png"><hr>
5 <p align="center"><font color="#990000" size="4"><strong><em>: CARO DOADOR PREENCHA
6 TODOS OS CAMPOS DO FORMULÁRIO ABAIXO:</em></strong></font></p><hr>
7 <table width="94%" height="50%" border="2" bordercolor="#990000" bgcolor="#CCCCCC"> <tr bgcolor="#CCCCCC">
8 <td height="23" colspan="4">
9 <div align="center"><strong><font size="5">AGENDAMENTO
10 &nbsp;&nbsp;&nbsp;DE &nbsp;&nbsp;&nbsp;DOA&Ccedil;&Atilde;O</font></strong></div></td> </tr> <tr bgcolor="#CCCCCC">
11 <td height="11%" colspan="2" nowrap>
12 <form name="form1" method="post" action=""> <strong>NOME:</strong>
13 <input name="NOMEDOADOR" type="text" id="NOMEDOADOR" size="80" maxlength="80"> </form></td>
14 <td width="10%" height="11%" nowrap>
15 <form name="form11" method="post" action=""> <strong>IDADE:</strong>
16 <input name="IDADEDOADOR" type="text" id="IDADE" size="1" maxlength="3"> </form></td> <td
17 width="32%" nowrap>
18 <form name="form12" method="post" action=""> <strong>SEXO:</strong>
19 <input type="radio" name="SEXODOADOR" value="MASCULINO"> Masc.
20 <input type="radio" name="SEXODOADOR" value="FEMININO"> Fem. </form></td> </tr> <tr
21 bgcolor="#CCCCCC"> <td height="8%" colspan="2" nowrap> <form name="form13" method="post" action=""> <div
22 align="left"> <strong>IDENTIDADE:</strong>: <input name="RG" type="text" id="RG3" size="9" maxlength="10"> </div>
23 </form></td>
24 <td height="8%" colspan="2" nowrap> <form name="form4" method="post" action=""> <strong>CPF:</strong>:
25 <input name="CPF" type="text" id="CPF2" size="10" maxlength="11"> </form></td> </tr>
26 <tr bgcolor="#CCCCCC">
27 <td height="8%" nowrap>
28 <form name="form2" method="post" action="">
29 <strong>ABO:
30 <input type="radio" name="ABODOADOR" value="A"> A
31 <input type="radio" name="ABODOADOR" value="B"> B
32 <input type="radio" name="ABODOADOR" value="AB"> AB
33 <input name="ABODOADOR" type="radio" value="O"> O </strong> </form></td>
34 <td height="8%" nowrap>
35 <form name="form6" method="post" action=""> <strong>RH:
36 <input type="radio" name="RHODOADOR" value="POSITIVO"> Positivo
37 <input type="radio" name="RHODOADOR" value="NEGATIVO"> Negativo </strong> </form></td>
38 <td height="8%" colspan="2" nowrap>
39 <form name="form8" method="post" action="">
40 <strong>DATA PREVISTA DA DOA&Ccedil;&Atilde;O:</strong>
41 <input name="DIA" type="text" id="DIA" size="2" maxlength="2">
42 <input name="MES" type="text" id="MES" size="2" maxlength="2">
43 <input name="ANO" type="text" id="ANO" size="4" maxlength="4"> </form></td> </tr>
44 <tr bgcolor="#CCCCCC">
45 <td height="8%" colspan="2" nowrap>
46 <form name="form3" method="post" action=""> <strong>RUA:</strong>
47 <input name="RUA" type="text" id="RUA2" size="80" maxlength="80"> </form></td>
48 <td height="8%" colspan="2" nowrap>
49 <form name="form10" method="post" action=""> <strong>N&ordm;:</strong>
50 <input name="NUMERO" type="text" id="NUMERO" size="9" maxlength="10"> </form></td> </tr>
51 <tr bgcolor="#CCCCCC">
52 <td height="11%" nowrap>
53 <form name="form5" method="post" action=""> <strong>BAIRRO:</strong>
54 <input name="BAIRRO" type="text" id="BAIRRO" size="21" maxlength="20"> </form></td> <td height="11%"
55 nowrap> <form name="form7" method="post" action=""> <strong>COMPLEMENTO:</strong>
56 <input name="COMPLEMENTO" type="text" id="COMPLEMENTO"> </form></td> <td height="11%" colspan="2"
57 nowrap> <form name="form17" method="post" action=""> <strong>UF:</strong>
58 <input name="UF" type="text" id="UF" size="1" maxlength="2"> </form></td> </tr> <tr bgcolor="#CCCCCC">
59 <td width="20%" height="11%" nowrap> <form name="form18" method="post" action="">
60 <strong>TELEFONE:</strong> <input name="DDD2" type="text" id="DDD2" size="1" maxlength="2"> &#8212;
61 <input name="TELEFONE" type="text" id="TELEFONE" size="7" maxlength="8"> </form></td> <td width="38%" nowrap>
62 <form name="form9" method="post" action=""> <strong>CELULAR:</strong>
63 <input name="DDD" type="text" id="DDD" size="1" maxlength="2">
64 <input name="CELULAR" type="text" id="CELULAR" size="7" maxlength="8"> </form></td> <td colspan="2"
65 nowrap> <form name="form16" method="post" action=""> <strong>EMAIL:</strong>
66 <input name="EMAIL" type="text" id="EMAIL" size="48" maxlength="50"> </form></td> </tr> <table> <form
67 name="form14" method="post" action=""> <div align="center">
68 <input name="SOLICITARDECLARACAO" type="submit" id="SOLICITARDECLARACAO" value="AGENDAR
69 DOA&Ccedil;&Atilde;O"> </div></form> <p align="center"><a href="IndexSistemaDoador.htm" target="_parent"><font
color="#000000" size="4"><strong>voltar</strong></font></a></p></body></html>

```

Figura 50. (Código HTML)

3.7.4. Tela de solicitação de declaração de doação

Na figura 51 vemos o formulário para solicitação de declaração de doação onde o usuário tem a opção de solicitar uma declaração de doação através de uma conexão com a internet utilizando o menu da esquerda no botão Solicitação de Declaração. Após preenchimento do formulário e confirmação do mesmo será impresso na tela a data para recebimento da declaração. O usuário receberá a declaração via e-mail sendo necessário que o mesmo tenha cadastro no banco de sangue e possua pelo menos três doações no período de doze meses.

Quarta-Feira, 07 de Dezembro de 2005

SWBS
BANCO DE SANGUE

-- CARO DOADOR PREENCHA O FORMULÁRIO ABAIXO --

SOLICITAÇÃO DE DECLARAÇÃO		
NOME	IDADE	SEXO ☐ Masc ☐ Fem.
IDENTIDADE	CPF	
RUA	Nº	
BARRIO	COMPLEMENTO	UF
TELEFONE	CELULAR	E-MAIL

Figura 51. (Tela do sistema)

Figura 52. (Código HTML)

Código HTML da figura 51:

A figura 52 faz referência ao código HTML da página de solicitação de declaração de doação como vimos na figura 51.

```

1 <html><head>
2 <title>Documento sem título</title>
3 <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
4 </head>
5 <body background="file:///E:/Erikson/Projeto%20Final%20de%20curso/ProjetoFinal/Image/cora%E7%E3o_2.png"><hr>
6 <p align="center"><font color="#990000" size="4"><strong><em>:: CARO DOADOR PREENCHA
7 O FORMUL&Aacute;RIO ABAIXO ::</em></strong></font></p><hr><BR>
8 <table width="20%" height="43%" border="2" bordercolor="#990000" bgcolor="#CCCCCC"> <tr bgcolor="#CCCCCC">
9 <td height="23" colspan="4"> <div align="center"><strong><font size="5">SOLICITA&Ccedil;&Atilde;O
10 DE&nbsp;&DECLARA&Ccedil;&Atilde;O</font></strong></div></td> </tr> <tr>
11 <td height="11%" colspan="2" nowrap bgcolor="#CCCCCC"> <form name="form1" method="post" action="">
12 <strong>NOME:</strong>
13 <input name="NOMEDOADOR" type="text" id="NOMEDOADOR2" size="80" maxlength="80"> </form></td>
14 <td height="11%" nowrap bgcolor="#CCCCCC"><form name="form11" method="post" action="">
15 <strong>IDADE:</strong>
16 <input name="IDADEDOADOR" type="text" id="IDADE2" size="1" maxlength="3"> </form></td>
17 <td nowrap bgcolor="#CCCCCC"><form name="form12" method="post" action=""> <strong>SEXO:</strong>
18 <input type="radio" name="SEXODOADOR" value="MASCULINO"> Masc.
19 <input type="radio" name="SEXODOADOR" value="FEMININO"> Fem. </form></td> </tr> <tr>
20 <td height="8%" colspan="2" nowrap><form name="form13" method="post" action="">
21 <div align="left"><strong>IDENTIDADE:</strong>
22 <input name="RGDOADOR" type="text" id="RGDOADOR" size="9" maxlength="10"> </div> </form></td>
23 <td height="8%" colspan="2" nowrap><form name="form4" method="post" action=""> <strong>CPF:</strong>
24 <input name="CPFDOADOR" type="text" id="CPFDOADOR" size="10" maxlength="11"> </form></td> </tr> <tr>
25 <td height="8%" colspan="2" nowrap><form name="form3" method="post" action=""> <strong>RUA:</strong>
26 <input name="RUADOADOR" type="text" id="RUA2" size="80" maxlength="80"> </form></td>
27 <td height="8%" colspan="2" nowrap><form name="form10" method="post" action=""> <strong>N&ordm;:</strong>
28 <input name="NUMERODOADOR" type="text" id="NUMERODOADOR" size="9" maxlength="10"> </form></td></tr> <tr>
29 <td height="11%" colspan="2" nowrap><form name="form5" method="post" action=""> <strong>BAIRRO:</strong>
30 <input name="BAIRRODOADOR" type="text" id="BAIRRODOADOR" size="21" maxlength="20"> </form></td>
31 <td height="11%" colspan="2" nowrap><form name="form7" method="post" action=""> <strong>COMPLEMENTO:</strong>
32 <input name="COMPLEMENTODOADOR" type="text" id="COMPLEMENTODOADOR"> </form></td>
33 <td height="11%" colspan="2" nowrap><form name="form17" method="post" action=""> <strong>UF:</strong>
34 <input name="UFDOADOR" type="text" id="UFDOADOR" size="1" maxlength="2"> </form></td> </tr> <tr>
35 <td width="15%" height="11%" nowrap><form name="form18" method="post" action=""> <strong>TELEFONE:</strong>
36 <input name="DDDADOADOR" type="text" id="DDD23" size="1" maxlength="2">
37 <input name="TELEFONADOADOR" type="text" id="TELEFONE3" size="7" maxlength="8"> </form></td>
38 <td width="33%" colspan="2" nowrap><form name="form9" method="post" action=""> <strong>CELULAR:</strong>
39 <input name="DDDADOADOR" type="text" id="DDD5" size="1" maxlength="2">
40 <input name="CELULARDOADOR" type="text" id="CELULAR4" size="7" maxlength="8"> </form></td>
41 <td width="25%" colspan="2" nowrap><form name="form16" method="post" action=""> <strong>EMAIL:</strong>
42 <input name="EMAILDOADOR" type="text" id="EMAIL3" size="48" maxlength="50"> </form></td> </tr></table>
43 <p align="center">
44 <input name="SOLICITARDECLARACAO" type="submit" id="SOLICITARDECLARACAO" value="SOLICITAR
45 DECLARA&Ccedil;&Atilde;O"></p>
46 <p align="center"><a href="IndexSistemaDoador.htm" target="_parent"><font color="#000000"
47 size="4"><strong>voltar</strong></font></a></p>
48 <p align="center">&nbsp;</p>
49 <p align="center">&nbsp;</p>
50 <p align="center">&nbsp;</p>
51 <p align="center">&nbsp;</p>
52 <p align="center">&nbsp;</p>
53 <p align="center">&nbsp;</p>
54 <p align="center">&nbsp;</p>
55 <p align="center">&nbsp;</p>
56 <p align="center"><a href="IndexSistemaDoador.htm" target="_parent">voltar</a>
57 </p>
58 </body>
59 </html>

```

Figura 52. (Código HTML)

3.7.5. Tela de login do sistema

A figura 53 é onde será feito o login do sistema, o acesso ao login é feito pelo menu da página principal a esquerda no botão SISTEMA, através do login o usuário terá acesso ao sistema, podendo serem aberto as páginas Agência, Hospital ou Gerenciamento, conforme perfil do usuário.

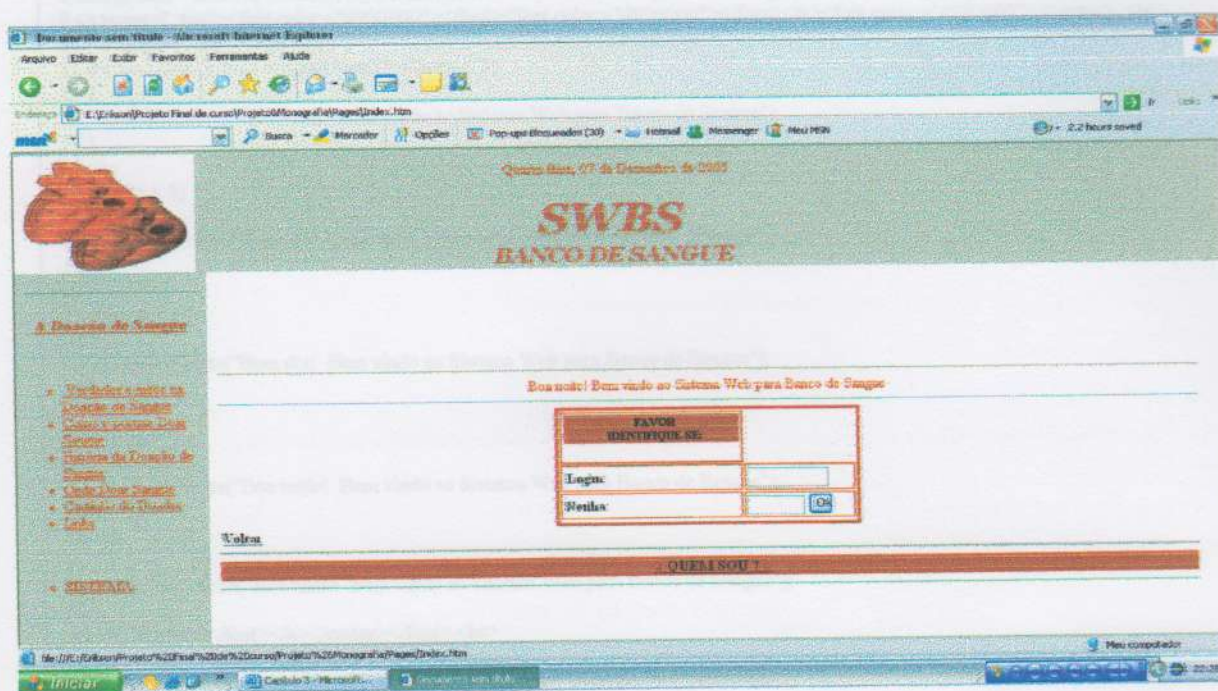


Figura 53. (Tela do sistema)

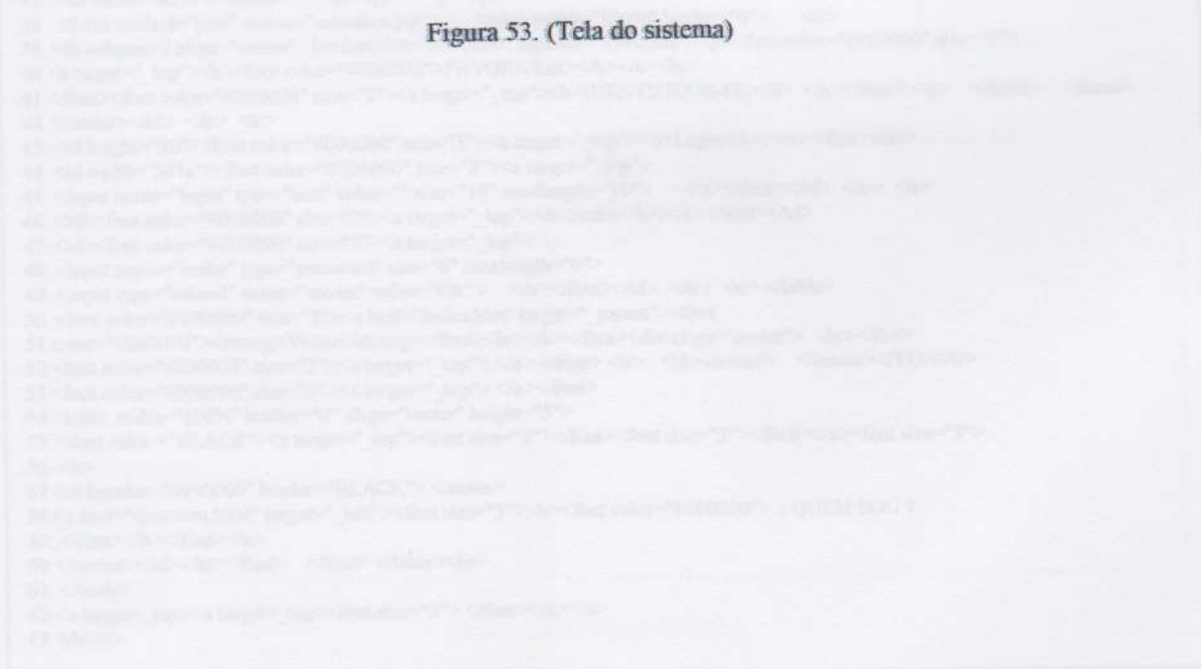


Figura 54. (Codigo HTML)

Código HTML da figura 53:

A figura 54 faz referência a página login do sistema e utiliza de comando e códigos em JavaScript como vemos na linha 10 até a linha 34, estes comando são responsáveis pela data q se apresenta na parte superior da figura 53.

```

1 <html>
2 <head>
3 <title>Untitled Document</title>
4 <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
5 </head>
6 <a target=_top><BR><BR><BR><BR><BR>
7 <p align="center"><a target=_top><font size="3"> </font></a></a> <body bgcolor="#FFFFFF"
8 background="file:///E:/Erikson/Projeto%20Final%20de%20curso/ProjetoFinal/Image/cora%E7%E3o_2.png">
9 <a target=_top><font color="#990000"></font><font color="#990000"></font></a><font color="#990000"> <center> <hr>
10 <a target=_top><font size="3"> <SCRIPT LANGUAGE="JavaScript">
11 d = new Date(); hour = d.getHours(); if(hour < 5)
12 {
13 document.write("Boa noite! Bem vindo ao Sistema Web para Banco de Sangue");
14 }
15 else
16 if(hour < 8)
17 {
18 document.write("Bom dia! Bem vindo ao Sistema Web para Banco de Sangue");
19 }
20 else
21 if(hour < 12)
22 {
23 document.write("Bom dia! Bem vindo ao Sistema Web para Banco de Sangue");
24 }
25 else
26 if(hour < 18)
27 {
28 document.write("Boa tarde! Bem vindo ao Sistema Web para Banco de Sangue");
29 }
30 else
31 {
32 document.write("Boa noite! Bem vindo ao Sistema Web para Banco de Sangue");
33 }
34 </SCRIPT> </font></a></center></font> <hr>
35 <div align="center"></div>
36 <table align="center" width="31%" border="3" height="" bordercolor="#990000">
37 <td width="62%"><center> <p></p> <p></p>
38 <form method="post" action="autentica.jsp"> <table width="100%" border="0"> <tr>
39 <th colspan=2 align="center" bordercolor="#990000" bgcolor="#990000"><p><font color="#990000" size="2">
40 <a target=_top><b><font color="#000000">FAVOR</font></b></a></b></a><br>
41 </font><font color="#000000" size="2"><a target=_top><b>IDENTIFIQUE-SE:</b></a></font> </p> </table> </form>
42 </center></td> </tr> <tr>
43 <td height="30"><font color="#000000" size="3"><a target=_top><b>Login</b></a></font></td>
44 <td width="38%"><font color="#000000" size="3"><a target=_top>
45 <input name="login" type="text" value="" size="10" maxlength="10"> </a></font></td> </tr> <tr>
46 <td><font color="#000000" size="3"><a target=_top><b>Senha</b></a></font></td>
47 <td><font color="#000000" size="3"><a target=_top>
48 <input name="senha" type="password" size="6" maxlength="6">
49 <input type="submit" name="enviar" value="Ok" > </a></font></td> </tr> </table>
50 <font color="#000000" size="3"><a href="Index.htm" target=_parent><font
51 color="#000000"><strong>Voltar</strong></font><br></a></font><div align="center"> <hr></div>
52 <font color="#990000" size="3"><a target=_top> </a> </font> <tr> <td><center> </center></TD></tr>
53 <font color="#990000" size="3"><a target=_top> </a></font>
54 <table width="100%" border="0" align="center" height="5">
55 <font color="BLACK"><a target=_top><font size="3"></font><font size="3"></font></a><font size="3">
56 <tr>
57 <td bgcolor="#990000" border="BLACK"> <center>
58 <a href="quemsou.html" target=_self><font size="3"><b><font color="#000000"> :: QUEM SOU ?
59 </font></b></font></a>
60 </center></td></tr></table> </font> </table><hr>
61 </body>
62 <a target=_top><a target=_top><font size="3"> </font></a></a>
63 </html>

```

Figura 54. (Código HTML)

3.7.6. Tela do módulo index agência

A figura 55 faz referência à página inicial do sistema no perfil de usuário agência. O mesmo dispõe de um menu a esquerda onde limita o uso do sistema conforme o seu perfil. Podendo ser solicitado hemocomponentes para transfusão e consulta de estoque.

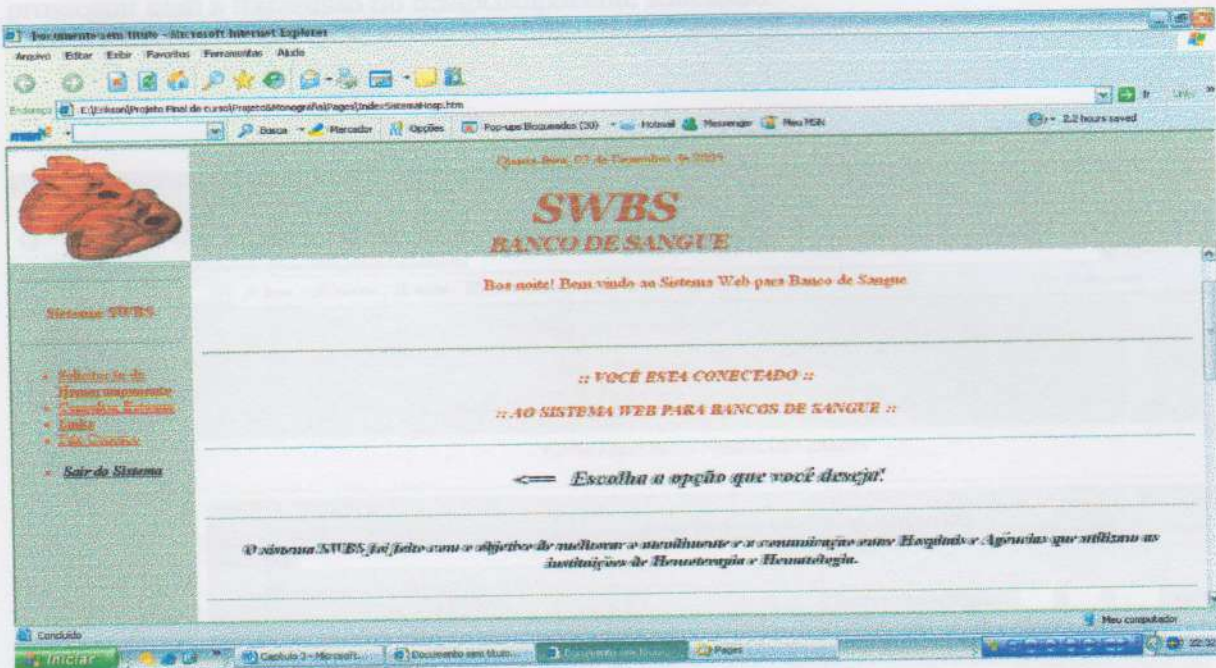


Figura 55. (Tela do sistema)

Código HTML da figura 55:

O código HTML da figura 56 faz referência a tela do perfil de usuário agência.

```

1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">
2 <html>
3 <head>
4 <title>Documento sem título</title>
5 <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
6 </head>
7 <frameset rows="137,*" cols="*" frameborder="NO" border="0" framespacing="0">
8 <frame src="Cabe%7alho.htm" name="topFrame" scrolling="NO" noresize >
9 <frameset rows="*" cols="192,*" framespacing="0" frameborder="NO" border="0">
10 <frame src="MenuSistemaAgencia.htm" name="leftFrame" scrolling="NO" noresize>
11 <frame src="ConteudoSistema.htm" name="mainFrame">
12 </frameset>
13 </frameset>
14 <noframes><body>
15 </body></noframes>
16 </html>

```

Figura 56. (Código HTML)

3.7.7. Tela de solicitação de hemocomponentes

A figura 57 faz parte dos perfis agência e hospital onde o usuário dispõe de um formulário onde poderá ser feito a solicitação de transfusão de hemocomponentes. Após o preenchimento do formulário com os dados do paciente e do tipo de transfusão o usuário confirma os dados clicando no botão “Confirmar Solicitação” e a central receberá um aviso visual e sonoro da solicitação para efetuar os devidos exames pré-transfusionais para prosseguir com a transfusão do hemocomponente solicitado.

Figura 57. (Tela do sistema)

Código HTML da figura 57:

A figura 58 faz referência ao formulário da figura 57 solicitação de hemocomponente.

```

1 <html><head>
2 <title>Documento sem título</title>
3 <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1"></head>
4 <body background="file:///E:/Erikson/Projeto%20Final%20de%20curso/ProjetoFinal/Image/cora%E7%E3o_2.png"><br>
5 <p align="center"><strong> <font color="#990000">:.&nbsp;. FAVOR PREENCHER O
6 FORMUL&Aacute;RIO ABAIXO:</font></strong></em><strong></strong></p><br>
7 <table width="60%" height="37%" border="2" bordercolor="#990000" bgcolor="#CCCCCC"> <tr bgcolor="#CCCCCC">
8 <td height="23" colspan="5"> <div align="center"><strong><font size="4">SOLICITA&Ccedil;&Atilde;O&nbsp;.
9 DE&nbsp;. HEMOCOMPONENTE</font></strong></div></td> </tr>
10 <tr bgcolor="#CCCCCC"> <td height="11%" colspan="3" nowrap> <form name="form1" method="post" action="">
11 <strong>PACIENTE:</strong> <input name="NOMEpaciente" type="text" id="NOMEpaciente" size="80"
12 maxlength="80"> </form></td> <td colspan="2" rowspan="4" nowrap> <form name="form11" method="post" action="">
13 &nbsp;. &nbsp;. DATA DE NASCIMENTO:&nbsp;.
14 <input name="DIANASC" type="text" id="DIANASC" size="2" maxlength="2"> /
15 <input name="MESnascimento" type="text" id="MESnascimento" size="2" maxlength="2"> /
16 <input name="ANONascimento" type="text" id="ANONascimento" size="4" maxlength="4">
17 &nbsp;. IDADE:</strong>
18 <input name="IDAEpaciente" type="text" id="IDAEpaciente" size="1" maxlength="3"> <strong></strong>
19 </form> <form name="form12" method="post" action=""> <strong>&nbsp;. SEXO:</strong>
20 <input checked="checked" name="SEXOPAC" type="radio" value="MASC"> Masc.
21 <input type="radio" name="SEXOPAC" value="FEM"> Fem. </form> <form name="form13" method="post" action="">
22 <div align="left"><strong>&nbsp;. PRONTU&Aacute;RIO:</strong>
23 <input name="PRONTUARIO" type="text" id="RG3" size="8" maxlength="8"> </div> </form>
24 <form name="form4" method="post" action=""> <strong>&nbsp;. ENFERMARIA</strong> :
25 <input name="ENFERMARIA" type="text" id="CPF2" size="8" maxlength="8"> </form>
26 <form name="form15" method="post" action=""> <strong>&nbsp;. LEITO:</strong>
27 <input name="LEITO" type="text" id="LEITO" size="3" maxlength="3">
28 </form> <form name="form8" method="post" action="">
29 <strong>&nbsp;. DATA DA TRANSFU&Atilde;O:</strong>
30 <input name="DIATRANSF" type="text" id="DIATRANSF" size="2" maxlength="2"> /
31 <input name="MESTRANSF" type="text" id="MESTRANSF" size="2" maxlength="2"> /
32 <input name="ANOTRANSF" type="text" id="ANOTRANSF" size="4" maxlength="4"> </form></td> </tr>
33 <tr bgcolor="#CCCCCC">
34 <td height="97%" rowspan="3" nowrap><form name="form19" method="post" action=""> <strong>ABO:
35 <input checked="checked" name="ABO" type="radio" value="A"> A
36 <input type="radio" name="ABO" value="B"> B
37 <input type="radio" name="ABO" value="AB"> AB
38 <input name="ABO" type="radio" value="O"> O </strong> </form> <br>
39 <form action="" method="get" name="form6"> <strong>RH:
40 <input checked="checked" name="RH" type="radio" value="POSITIVO"> Positivo
41 <input type="radio" name="RH" value="NEGATIVO"> Negativo </strong></form></td>
42 <td width="15%" height="22" nowrap><form name="form3" method="post" action=""> <p>
43 <input checked="checked" name="HEMOCOMP" type="radio" value="CHA"> <strong> CHA&nbsp;. <br>
44 <input type="radio" name="HEMOCOMP" value="CHP"> CHP </strong><strong><br>
45 <input type="radio" name="HEMOCOMP" value="PF"> PF<br>
46 <input type="radio" name="HEMOCOMP" value="CP"> CP<br>
47 <input type="radio" name="HEMOCOMP" value="AF"> AFERESE<br>
48 <input type="radio" name="HEMOCOMP" value="CRIO"> CRIO PRECIPITADO</strong></p> </form></td> <br>
49 <td width="16%" height="97%" rowspan="3" nowrap><form name="form2" method="post" action=""> <div align="left">
50 <input checked="checked" type="radio" name="TIPOSOLIC" value="PREOPERATORIO"> <strong> Pr&eacute;-eratoacuteriorio<br>
51 <br> <br> </strong>
52 <input name="TIPOSOLIC" type="radio" value="ROTINA"> <strong> Rotina </strong></div>
53 </form></td> </tr> <tr bgcolor="#CCCCCC">
54 <td height="23" nowrap><form name="form20" method="post" action=""> <strong>
55 <input checked="checked" name="HEMOCOMPTIPO" type="radio" value="NORMAL"> NORMAL<br>
56 <input type="radio" name="HEMOCOMPTIPO" value="IRRADIADO"> IRRADIADO <br>
57 <input type="radio" name="HEMOCOMPTIPO" value="PL"> POBRE EM LEUC&Oacute;, CITOS</strong><br> <strong>
58 <input type="radio" name="HEMOCOMPTIPO" value="DELEUCOTIZADO"> DELEUCOTIZADO</strong><br> <strong>
59 <input type="radio" name="HEMOCOMPTIPO" value="LAVADO"> LAVADO</strong></form></td> </tr>
60 <tr bgcolor="#CCCCCC"> <td height="48" nowrap><form name="form16" method="post" action="">
61 <strong>QUANTIDADE:</strong>
62 <input name="QUANTIDADE" type="text" id="QUANTIDADE" size="2" maxlength="2"> </form></td> </tr>
63 <tr bgcolor="#CCCCCC"> <td height="11%" colspan="3" nowrap> <div align="center"><strong>ENDERE&Ccedil;O
64 <strong></div> <form name="form10" method="post" action="">
65 <strong>RUA:</strong> <input name="RUAPAC" type="text" id="RUAPAC" size="80" maxlength="80"> </form>
66 <form name="form5" method="post" action="">
67 <strong>BAIRRO:</strong>
68 <input name="BAIROPAC" type="text" id="BAIRO2" size="30" maxlength="30">

```

Continuação. da consulta de cirurgias

```

67 </form>
68 <form name="form7" method="post" action="">
69 <strong>COMPLEMENTO:</strong>
70 <input name="COMPLEMENTOPAC" type="text" id="COMPLEMENTO3" size="20" maxlength="20">
71 &nbsp;<strong>CIDADE:&nbsp;</strong>
72 <input name="CIDADEPAC" type="text" id="CIDADEPAC" size="20" maxlength="20">
73 </strong>&nbsp;<strong>UF</strong>
74 <input name="UFPAC" type="text" id="UFPAC" size="2" maxlength="2">
75 </form></td>
76 <td height="11%" colspan="2" nowrap><strong>OBSERVA&Ccedil;&Otilde;ES:</strong>
77 <form name="form21" method="post" action="">
78 <div align="center">
79 <textarea name="OBSERVACAO" cols="40" rows="8" id="OBSERVACAO"></textarea>
80 </div> </form></td> </tr> <tr bgcolor="#CCCCC"> <td width="16%" height="26" nowrap> <form name="form18"
81 method="post" action=""> <strong>TELEFONE:</strong> <input name="DDDPAC" type="text" id="DDD22" size="1"
82 maxlength="2"> &#8212; <input name="TELEFONEPAC" type="text" id="TELEFONEPAC" size="7" maxlength="8">
83 </form></td> <td colspan="2" nowrap> <form name="form9" method="post" action=""> <strong>CELULAR:</strong>
84 <input name="DDDPAC" type="text" id="DDDPAC" size="1" maxlength="2"> &#8212; <input name="CELULARPAC"
85 type="text" id="CELULARPAC" size="7" maxlength="8">
86 </form></td> <td colspan="2" nowrap> <form name="form17" method="post" action=""> <div
87 align="left"><strong>M&Eacute;DICO:</strong>&nbsp;</div>
88 <input name="MEDICO" type="text" id="MEDICO" size="40" maxlength="40">
89 &nbsp;<strong>CRM:&nbsp;</strong> <input name="CRM" type="text" id="CRM" size="8" maxlength="8">
90 </div> </form></td></tr></table><form name="form14" method="post" action=""> <div align="center">
91 <input name="SOLICITARHEMOCOMPONENTE" type="submit" id="SOLICITARHEMOCOMPONENTE"
92 value="SOLICITAR HEMOCOMPONENTE"> </div> </form> <p align="center"><strong><font color="#000000" size="3">
93 <a href="IndexSistemaHosp.htm" target="_parent">Voltar</a></font></strong></p>
94 </body>
95 </html>

```

Figura 58. (Código HTML)


```

1 <html><head>
2 <title>Documento sem título</title>
3 <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
4 </head>
5 <body background="file:///E:/Erikson/Projeto%20Final%20de%20curso/ProjetoFinal/Image/cora%E7%E3o_2.png"><hr>
6 <p align="center"><strong><font color="990000" size="4"><em>: CONSULTAR ESTOQUE :
7 </em></font></strong></p><hr>
8 <p><table border="2" width="100%" align="CENTER">
9 <tr><th colspan="9" bgcolor="990000"><font size="3" color="white"><b>ESTOQUE</b></font> <tr>
10 <td width="52%" bgcolor="990000" align="center"><font
11 color="FFFFFF"><strong>HEMOCOMPONENTE</strong></font></td>
12 <td width="6%" bgcolor="990000" align="center"><font size="3" color="white"><b>A
13 </b></font></td>
14 <td width="6%" bgcolor="990000" align="center"><font size="3" color="white"><b>B
15 </b></font></td> <td width="6%" bgcolor="990000" align="center"><strong><font color="FFFFFF">AB
16 </font></strong></td> <td width="6%" bgcolor="990000" align="center"><strong><font color="FFFFFF">O
17 </font></strong></td> <td width="6%" bgcolor="990000" align="center"><strong><font color="FFFFFF">A
18 </font></strong></td> <td width="6%" bgcolor="990000" align="center"><font size="3" color="white"><strong>B
19 </strong></font></td> <td width="6%" bgcolor="990000" align="center"><strong><font color="FFFFFF">AB
20 </strong></font></td> <td width="6%" bgcolor="990000" align="center"><font color="FFFFFF"><strong>O
21 </strong></font></td> <tr> <td width="52%" bgcolor="990000" align="center"><div align="left"><strong><font
22 color="FFFFFF">CHA</font></strong></div></td>
23 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
24 <td width="6%" bgcolor="990000" align="center"><font size="3" color="white"><strong>&nbsp;</strong></font></td>
25 <td width="6%" bgcolor="990000" align="center"><font size="3" color="white">&nbsp;</font></td>
26 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
27 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
28 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
29 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
30 <td width="6%" bgcolor="990000" align="center">&nbsp;</td> <tr>
31 <td width="52%" bgcolor="990000" align="center"><div align="left"><strong><font color="FFFFFF">CH
32 POBRE EM LEUC&Oacute;CITOS</font></strong></div></td>
33 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
34 <td width="6%" bgcolor="990000" align="center"><font size="3" color="white"><b></b></font></td>
35 <td width="6%" bgcolor="990000" align="center"><font size="3" color="white">&nbsp;</font></td>
36 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
37 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
38 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
39 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
40 <td width="6%" bgcolor="990000" align="center">&nbsp;</td> <tr>
41 <td width="52%" bgcolor="990000" align="center"><div align="left"><font color="FFFFFF"><strong>CH
42 PEDI&Aacute;TRICO</strong></font></div></td>
43 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
44 <td width="6%" bgcolor="990000" align="center"><font size="3" color="white"><b></b></font></td>
45 <td width="6%" bgcolor="990000" align="center"><font size="3" color="white">&nbsp;</font></td>
46 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
47 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
48 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
49 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
50 <td width="6%" bgcolor="990000" align="center">&nbsp;</td> <tr>
51 <td bgcolor="990000" align="center"><div align="left"><font color="FFFFFF"><strong>CH
52 LAVADO</strong></font></div></td>
53 <td bgcolor="990000" align="center">&nbsp;</td>
54 <td bgcolor="990000" align="center">&nbsp;</td>
55 <td bgcolor="990000" align="center">&nbsp;</td>
56 <td bgcolor="990000" align="center">&nbsp;</td>
59 <td bgcolor="990000" align="center">&nbsp;</td>
60 <td bgcolor="990000" align="center">&nbsp;</td>
61 <td bgcolor="990000" align="center">&nbsp;</td>
62 <td bgcolor="990000" align="center">&nbsp;</td> <tr>
63 <td bgcolor="990000" align="center"><div align="left"><strong><font color="FFFFFF">CH
64 IRADIADO</font></strong></div></td>
65 <td bgcolor="990000" align="center">&nbsp;</td>

```

Continuação.

```

66 <td bgcolor="990000" align="center">&nbsp;</td>
67 <td bgcolor="990000" align="center">&nbsp;</td>
68 <td bgcolor="990000" align="center">&nbsp;</td>
69 <td bgcolor="990000" align="center">&nbsp;</td>
70 <td bgcolor="990000" align="center">&nbsp;</td>
71 <td bgcolor="990000" align="center">&nbsp;</td>
72 <td bgcolor="990000" align="center">&nbsp;</td> <tr>
73 <td bgcolor="990000" align="center"><div align="left"><font color="#FFFFFF"><strong>PLASMA
74 FRESCO</strong></font></div></td>
75 <td bgcolor="990000" align="center">&nbsp;</td>
76 <td bgcolor="990000" align="center">&nbsp;</td>
77 <td bgcolor="990000" align="center">&nbsp;</td>
78 <td bgcolor="990000" align="center">&nbsp;</td>
79 <td bgcolor="990000" align="center">&nbsp;</td>
80 <td bgcolor="990000" align="center">&nbsp;</td>
81 <td bgcolor="990000" align="center">&nbsp;</td>
82 <td bgcolor="990000" align="center">&nbsp;</td> <tr>
83 <td bgcolor="990000" align="center"><div align="left"><strong><font color="#FFFFFF">CRIO
84 PRECIPITADO</font></strong></div></td>
85 <td bgcolor="990000" align="center">&nbsp;</td>
86 <td bgcolor="990000" align="center">&nbsp;</td>
87 <td bgcolor="990000" align="center">&nbsp;</td>
88 <td bgcolor="990000" align="center">&nbsp;</td>
89 <td bgcolor="990000" align="center">&nbsp;</td>
90 <td bgcolor="990000" align="center">&nbsp;</td>
91 <td bgcolor="990000" align="center">&nbsp;</td>
92 <td bgcolor="990000" align="center">&nbsp;</td> <tr>
93 <td bgcolor="990000" align="center"><div align="left"><strong><font color="#FFFFFF">POOL
94 DE CRIO</font></strong></div></td>
95 <td bgcolor="990000" align="center">&nbsp;</td>
96 <td bgcolor="990000" align="center">&nbsp;</td>
97 <td bgcolor="990000" align="center">&nbsp;</td>
98 <td bgcolor="990000" align="center">&nbsp;</td>
99 <td bgcolor="990000" align="center">&nbsp;</td>
100 <td bgcolor="990000" align="center">&nbsp;</td>
101 <td bgcolor="990000" align="center">&nbsp;</td>
102 <td bgcolor="990000" align="center">&nbsp;</td> <tr>
103 <td bgcolor="990000" align="center"><div align="left"><font color="#FFFFFF"><strong>PLAQUETAS
104 </strong></font></div></td>
105 <td bgcolor="990000" align="center">&nbsp;</td>
106 <td bgcolor="990000" align="center">&nbsp;</td>
107 <td bgcolor="990000" align="center">&nbsp;</td>
108 <td bgcolor="990000" align="center">&nbsp;</td>
109 <td bgcolor="990000" align="center">&nbsp;</td>
110 <td bgcolor="990000" align="center">&nbsp;</td>
111 <td bgcolor="990000" align="center">&nbsp;</td>
112 <td bgcolor="990000" align="center">&nbsp;</td> <tr>
113 <td width="52%" bgcolor="990000" align="center"><div align="left"><font color="#FFFFFF"><strong>PLAQUETAS
114 POR AFERESE</strong></font></div></td>
115 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
116 <td width="6%" bgcolor="990000" align="center"><font size="3" color="white"><b></b></font></td>
117 <td width="6%" bgcolor="990000" align="center"><font size="3" color="white">&nbsp;</font></td>
118 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
119 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
120 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
121 <td width="6%" bgcolor="990000" align="center">&nbsp;</td>
122 <td width="6%" bgcolor="990000" align="center">&nbsp;</td></table>
123 <tr><td align="center">&nbsp;</td></tr>
124 <tr><td align="center">&nbsp;</td></tr>
125 <tr><td align="center">&nbsp;</td></tr>
126 <tr><td align="center">&nbsp;</td></tr>
127 <tr><td align="center">&nbsp;</td></tr>
128 <tr><td align="center">&nbsp;</td></tr>
129 <p align="center"><a href="IndexSistemaHosp.htm" target="_parent"><font
130 color="#000000">"4"><strong>Voltar</strong></font></a>
131 </p></body></html>

```

Figura 60. (Código HTML)

3.8. CÓDIGOS JSP Cadastro de Paciente

3.8.1. Código JSP Autentica Login

o código JSP do formulário de cadastro de paciente.
Podemos observar a utilização de métodos da classe Java Paciente e PacienteBD.

A figura 61 faz referência ao código JSP login que faz referência com a figura 53. Podemos observar que os códigos JSP estão entre as tag's de inicialização de uma página HTML `<html>` e `</html>` nas linhas 1 e 20 o mesmo dá início no comando `<%` e termina no comando `%>` como vemos nas linhas 4 e 19 dentro destes comandos observamos a utilização de classes, métodos e comandos em Java, como vemos nas linhas 7, 14, 16, 18.

```

1 <html>
2 <head><title>Resultado da autenticação</title></head>
3
4 <%@ page import="java.sql.*"%>
5
6 <%
7 Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
8 Connection banco=DriverManager.getConnection("jdbc:odbc:BancoDeSangue","","");
9 Statement stmt=banco.createStatement();
10 ResultSet consulta=stmt.executeQuery("SELECT* from senha ORDER BY nome");
11 %>
12 <body>
13 <%
14 String login=request.getParameter("login");
15 String login=request.getParameter("senha");
16 if (login!=null && (!login.equals("")) && senha!=null&&(!senha.equals(""))){
17 autenticador auth=new autenticador();
18 if (auth.autentica(login,senha)){
19 %>
20 </html>

```

Figura 61. (Código JSP)

Figura 62. (Código JSP)

3.8.2. Código JSP Cadastro de Paciente

A figura 62 faz referência ao código JSP do formulário de cadastro de paciente. Podemos observar a utilização de métodos da classe Java Paciente e PacienteBD.

```

1 <%@ page import="IndexSistemaCadastroPaciente" %>
2 <%@ page import="IndexSistemaCadastroPacienteBD" %>
3 <html>
4 <head>
5 <title>Cadastro de paciente</title>
6 </head>
7 <body>
8 <% Paciente paciente = new Paciente();
9     paciente.setPaciente(request.getParameter("nome"));
10    paciente.setPaciente(request.getPaciente("diaNasc"));
11    paciente.setPaciente(request.getPaciente("mesNasc"));
12    paciente.setPaciente(request.getPaciente("anoNasc"));
13    paciente.setPaciente(request.getPaciente("idade"));
14    paciente.setPaciente(request.getPaciente("sexo"));
15    paciente.setPaciente(request.getPaciente("prontuario"));
16    paciente.setPaciente(request.getPaciente("enfermaria"));
17    paciente.setPaciente(request.getPaciente("leito"));
18    paciente.setPaciente(request.getPaciente("nomePai"));
19    paciente.setPaciente(request.getPaciente("nomeMae"));
20    paciente.setPaciente(request.getPaciente("rua"));
21    paciente.setPaciente(request.getPaciente("bairro"));
22    paciente.setPaciente(request.getPaciente("complemento"));
23    paciente.setPaciente(request.getPaciente("cidade"));
24    paciente.setPaciente(request.getPaciente("uf"));
25    paciente.setPaciente(request.getPaciente("numero"));
26    paciente.setPaciente(request.getPaciente("codTel"));
27    paciente.setPaciente(request.getPaciente("telefone"));
28    paciente.setPaciente(request.getPaciente("codCel"));
29    paciente.setPaciente(request.getPaciente("celular"));
30    paciente.setPaciente(request.getPaciente("obs"));
31    paciente.setPaciente(request.getPaciente("medico"));
32    paciente.setPaciente(request.getPaciente("crm"));
33    PacienteBD pacienteBD = new PacienteBD();
34    pacienteBD.salva(paciente);
35
36 %>
37
38 Paciente cadastrado com sucesso !!!<br><br><br><br>
39 <center>
40
41 <a href="PacienteBD.jsp"><font color="orange"><center>Voltar</font></center>
42 </body>
43 </html>

```

Figura 62. (Código JSP)

3.9. Códigos Java

3.9.1. Código Java PacienteBD

A figura 63 faz referência a classe Java PacienteBD que utiliza de comando SQL para inserir dados em banco de dados e métodos para conexão com banco de dados.

```

1 package SistemaSWBS;
2 import java.sql.*;
3 import java.util.*;
4 public class PacienteBD {
5     public PacienteBD() {
6     }
7     public void salva (Paciente paciente){
8         try {
9             Connection con = Banco.getConexao();
10            Statement stmt = con.createStatement();
11            String query = "INSERT INTO Paciente(NOME, DIANASC, MESNASC, ANONASC, IDADE,
12 SEXO, PRONTUARIO, ENFERMARIA, LEITO, NOMEPAI, NOMEMAE, RUA, BAIRRO,
13 COMPLEMENTO, CIDADE, UF, NUMERO, CODTEL, TELEFONE, CODCEL, CELULAR, OBS,
14 MEDICO, CRM) VALUES
15 (" + paciente.getNome() + "," +
16 paciente.getEndereco() + "," + paciente.getDianasc() + "," + paciente.getMesnasc()
17 + "," + paciente.getAnonasc() + "," +
18 paciente.getIdade() + "," + paciente.getsexo() + "," + "paciente.getProntuario() + "," +
19 "paciente.getEnfermaria() + "," +
20 paciente.getLeito() + "," + paciente.getNomepai() + "," + "paciente.getNomemae() + "," +
21 "paciente.getRua() + "," +
22 paciente.getBairro() + "," + paciente.getComplemento() + "," + "paciente.getCidade() +
23 " + " + paciente.getUF() + "," +
24 paciente.getNumero() + "," + paciente.getCodtel() + "," + "paciente.getTelefone() + "," +
25 paciente.getCodcel() + "," +
26 paciente.getCelular() + "," + "paciente.getOBS() + "," + "paciente.getMedico() + "," +
27 "paciente.getCRM() + "," +
28 stmt.executeUpdate(query);
29 stmt.close();
30 con.commit();
31 con.close();
32 } catch (SQLException e){
33 System.out.println("Problemas ao abrir a conexao com o BD");
34 }
35 }
36 public Vector getAll(){
37     Vector listaPaciente= new Vector();
38     Paciente paciente = null;
39     try {
40         Connection con = Banco.getConexao();
41         Statement stmt = con.createStatement();
42         String query = "SELECT * FROM Paciente" ;
43         ResultSet resp = stmt.executeQuery(query);
44         while (resp.next() ){
45             paciente = new Paciente(resp.getString(1),resp.getString(2),resp.getString(3), resp.getString(4),
46             resp.getString(5),resp.getString(6),resp.getString(7),resp.getString(8),resp.getString(9),
47             resp.getString(10),resp.getString(11));
48             listaPaciente.add(paciente);
49         }
50     }

```

Continuação.

```
51      stmt.close();
52      con.close();
53      } catch (SQLException e){
54          System.out.println("Problemas ao abrir a conexao com o BD");
55      }
56      return listaPaciente;
57  }
58  }
```

Figura 63. (Código Java)

3.9.2. Código Java Classe Paciente

A figura 64 faz referência ao código Java da classe Paciente. Nesta classe observamos os seus atributos declarados a partir da linha 4 até a linha 28 e seus métodos a partir da linha 30 até a linha 132.

```
1 package SistemaSWBS;
2 public class Paciente {
3
4     private int id_paciente;
5     private String nome;
6     private int diaNasc;
7     private int mesNasc;
8     private int anoNasc;
9     private int idade;
10    private String sexo;
11    private int prontuario;
12    private int enfermaria;
13    private int leito;
14    private String nomePai;
15    private String nomeMae;
16    private String rua;
17    private String bairro;
18    private String complemento;
19    private String cidade;
20    private String uf;
21    private int numero;
22    private int codTel;
23    private int telefone;
24    private int codCel;
25    private int celular;
26    private String obs;
27    private String medico;
28    private int crm;
29
30    public Paciente(String nome,int diaNasc, int mesNasc, int anoNasc, int idade, String
31    sexo, int prontuario, int enfermaria, int leito, String nomePai, String nomeMae, String
32    rua, String bairro,String complemento, String cidade,String uf, int numero, int codTel,
33    int telefone, int codCel, int celular, String obs, String medico, int crm){
```

Continuação.

```

34         this.nome = nome;
35         this.diaNasc = diaNasc;
36         this.mesNasc = mesNasc;
37         this.anoNasc = anoNasc;
38         this.idade = idade;
39         this.sexo = sexo;
40         this.prontuario = prontuario;
41         this.enfermaria = enfermarias;
42         this.leito = leito;
43         this.nomePai = nomePai;
44         this.nomeMae = nomeMae;
45         this.rua = rua;
46         this.bairro = bairro;
47         this.complemento = complemento;
48         this.cidade = cidade;
49         this.uf = uf;
50         this.numero = numero;
51         this.codTel = codTel;
52         this.telefone = telefone;
53         this.codCel = codCel;
54         this.celular = celular;
55         this.obs = obs;
56         this.medico = medico;
57         this.crm = crm;
58     }
59     /**
60      * @return Returns the nome.          */
61     public String getNome() {
62         return nome;
63     }
64     /**
65      * @param nome The nome to set.        */
66     public void setNome(String nome) {
67         this.nome = nome;
68     }
69     /**
70      * @return Returns the bairro.          */
71     public String getBairro() {
72         return bairro;
73     }
74     /**
75      * @param bairro The bairro to set.      */
76     public void setBairro(String bairro) {
77         this.bairro = bairro;
78     }
79     /**
80      * @return Returns the cidade.          */
81     public String getCidade() {
82         return cidade;
83     }
84     /**
85      * @param cidade The cidade to set.      */
86     public void setCidade(String cidade) {
87         this.cidade = cidade;
88     }
89     /**
90      * @return Returns the prontuario.        */
91     public String getProntuario() {
92         return prontuario;
93     }
94     /**
95      * @param prontuario The prontuario to set.  */
96     public void setProntuario(String prontuario) {
97         this.prontuario = prontuario;
98     }
99     /**
100      * @return Returns the telefone.          */
101     public String getTelefone() {
102         return telefone;
103     }

```

```

92      * @param telefone The telephone to set.          */
93      public void setTelefone (String telefone) {
94          this.telefone = telefone;
95      } /**
96      * @return Returns the uf.                          */
97      public String getUf() {
98          return uf;
99      } /**
100     * @param uf The uf to set.                          */
101     public void setUf(String uf) {
102         this.uf = uf;
103     } /**
104     * @return Returns the numero.                       */
105     public String getNumero() {
106         return numero;
107     } /**
108     * @param numero The numero to set.                  */
109     public void setNumero(String numero) {
110         this.numero = numero;
111     } /**
112     * @return Returns the endereco.                    */
113     public String getEndereco() {
114         return endereco;
115     } /**
116     * @param endereco The endereco to set.             */
117     public void setEndereco(String endereco) {
118         this.endereco = endereco;
119     } /**
120     * @return Returns the medico.                      */
121     public String getMedico () {
122         return medico;
123     } /**
124     * @param medico The medico to set.                 */
125     public void setMedico(String medico) {
126         this.medico = medico;
127     } /**
128     * @return Returns the medico.                      */
129     public String getMedico() {
130         return medico;
131     }
132 }

```

Figura 64. (Código Java)

REFERÊNCIAS BIBLIOGRÁFICAS

- ALCÂNTARA, Andréia Almeida et al. **Home Pages: Recursos e técnicas para criação de página WWW.** 2.ed. Rio de Janeiro: Campus, 1997.
- ARROYO, Alexandre; BARCELLOS, Magali. **DreamWeaver MX.** Disponível em: 2005. <<http://www.apostilando.com/download.php?cod=405>>. Acesso em: 20 de Dezembro de 2005.
- BLAZUS, Diogo de Oliveira. **PostgreSQL.** Disponível em: 2003. <<http://www.postgresql.org.br/pmwiki.php?n=Main.PostgreSQL>>. Acesso em: 20 de Dezembro de 2005.
- CASTRO, Maria Alice Soares. Disponível em: 1995 <<http://www.icmc.usp.br/ensino/material/html>>. Acesso em: 10 Abril de 2005.
- COMER, Douglas E.; STEVENS, David I. **Interligação em Redes com TCP/IP.** v.2. Rio de Janeiro: Campus, 1999.
- CONSTANTINE, Larry. **Fundamentos do Desenho Orientado a Objeto com UML.** São Paulo: Makron Books, 2000. 462p.
- DEITEL, H. M. ; DEITEL, P. J. **Java: Como Programar.** 3.ed. Porto Alegre: Editora Artmed LTDA, 2000, 1.200p
- EISEIN, Hemoterapia. Disponível em: 2004<
<<http://www.einstein.br/bancodesangue/>> Acesso em: 10 Maio de 2005.
- FIELDS, Duane K.; KOLB, Mark A. **Desenvolvendo na Web com JSP: Um guia prático para projetar e construir na Web.** Rio de Janeiro: Ciência Moderna LTDA, 2000.
- GEYER, Cláudio. **Comparativo entre tecnologias JSP e ASP.** Disponível em: 2001, <<http://www.inf.ufrgs.br/procar/disc/cmp167/trabalhos/sem2001-1/T2/alex/>> Acesso em: 10 Abril de 2005.
- HAAF Serviço de Hemoterapia . Disponível em: 07 Maio de 2005 <<http://www.haaf.aer.mil.br>>. Acesso em: 07 de Maio de 2005.
- HEIDE, Ann; STILBORNE, Linda. **Guia do professor para Internet: Completo e Fácil.** 2.ed. Porto Alegre: Artes Médicas Sul LTDA, 1999.
- LARMAN, Craig. **Utilizando UML e Padrões: Uma Introdução a Análise e ao Projeto Orientado a Objeto.** Porto Alegre: Bookman, 2000. 592p.
- LEGISLAÇÃO para Bancos de Sangue. Disponível em: 10 Abril de 2005 <<http://www.anvisa.gov.br/sangue/legis/index.htm>>. Acesso em: 10 Abril de 2005.

LEITE, Mario; Sanra, Nelson A. **Programação Orientada ao Objeto: uma abordagem didática.** Disponível em: 10 Maio de 2005<http://www.revista.unicamp.br/infotec/artigos/leite_rahall.html>. Acesso em: 10 Maio de 2005.

NIEMAYER, Patrick; KNUDSEN, Jonathan. **Aprendendo Java.** Rio de Janeiro: Editora Campus LTDA, 2000, 700p.

NORMAS Técnicas para Coleta e Transfusão de Sangue, Componentes e Derivados. 8.ed. Brasília: Ministério da Saúde, 2002. 80p.

PRICE, Tom. **Programa de especialização: opção para o Mestrado.** Porto Alegre, UFRGS, 1997.

REGULAMENTO Técnico dos Serviços de hemoterapia: Resolução – RDC Nº 153. Brasília: Agência Nacional de Vigilância Sanitária, 2004. 48p.

TODD, Nick; SZOLKOWSKI, Mark. **JavaServer Pages: O Guia do Desenvolvedor.** Rio de Janeiro: Editora Campus, 2003. 621p.