



**UNIPAC**  
**UNIVERSIDADE PRESIDENTE ANTÔNIO CARLOS**  
**FACULDADE DE CIÊNCIA DA COMPUTAÇÃO E**  
**COMUNICAÇÃO SOCIAL**

**CURSO DE CIÊNCIA DA COMPUTAÇÃO**

**Epifanio Augusto de Lima**

**SISTEMA WEB PARA GERENCIAMENTO DE UMA CLÍNICA MÉDICA**

**BARBACENA**  
**JULHO DE 2005**

## **AGRADECIMENTOS**

Agradeço aos meus pais, meus irmão e à minha noiva Danyelle por estarem ao meu lado em todos os momentos, auxiliando e incentivando em mais esta etapa da minha vida.

Ao Isaque e ao Cleiton pela amizade, apoio e aulas de violão.

Ao orientador do trabalho de conclusão de curso, Prof. Emerson Rodrigo Alves Tavares, por toda atenção e apoio.

Aos familiares, professores e colegas de classe que direta ou indiretamente ajudaram na conclusão deste trabalho.

# SUMÁRIO

<b>LISTA DE FIGURAS.....</b>	<b>4</b>
<b>1 – INTRODUÇÃO.....</b>	<b>5</b>
1.1 – CONTEXTO.....	5
1.2 – PROPOSTA.....	5
1.3 – OBJETIVOS.....	6
1.4 – ESTRUTURA DO TRABALHO.....	6
<b>2 – REVISÃO BIBLIOGRÁFICA.....</b>	<b>8</b>
2.1 – HTML.....	8
2.2 – PHP.....	9
2.2.1 – <i>Requisitos de segurança</i> .....	12
2.2.2 – <i>Requisitos de avaliação de autenticação de controle de acesso</i> .....	13
2.2.3 – <i>Vulnerabilidades no PHP</i> .....	14
2.2.4 – <i>Usando PHP de modo seguro</i> .....	14
2.3 – SISTEMAS ESPECIALISTAS.....	15
2.4 – HIPERMÍDIA.....	16
2.5 – MYSQL.....	16
<b>3 – MODELAGEM DO SISTEMA.....</b>	<b>17</b>
3.1 – ANÁLISE DE REQUISITOS.....	18
3.2 – DIAGRAMA DE CASOS DE USO.....	19
3.3 – DIAGRAMA DE CLASSES.....	20
3.4 – DIAGRAMA DE SEQUÊNCIA.....	21
3.5 – MODELAGEM DO BANCO DE DADOS.....	24
3.7 – DIAGRAMA DE NAVEGABILIDADE.....	25
<b>4 – MODELAGENS DA HIPERMÍDIA.....</b>	<b>27</b>
4.1 – MODELO CONCEITUAL.....	27
4.2 – MODELO NAVEGACIONAL.....	28
4.3 – PROJETO DE INTERFACE ABSTRATA.....	29
<b>5 – IMPLEMENTAÇÃO.....</b>	<b>32</b>
5.1 – IMPLEMENTAÇÃO.....	32
<b>6 – CONSIDERAÇÕES FINAIS.....</b>	<b>45</b>
6.1 – TRABALHOS FUTUROS.....	45
<b>7 – REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>46</b>
<b>ANEXO 1 – CONFIGURANDO O SERVIDOR APACHE.....</b>	<b>48</b>
<b>ANEXO 2 – INSTALANDO E CONFIGURANDO O PHPMYADMIN.....</b>	<b>51</b>
<b>ANEXO 3 – SISCLIN.....</b>	<b>53</b>

## LISTA DE FIGURAS

FIGURA 1: CASOS DE USO: CENÁRIO ADMINISTRADOR.....	19
FIGURA 2: CASOS DE USO: CENÁRIO MÉDICO.....	20
FIGURA 3: CASOS DE USO: CENÁRIO PACIENTE.....	20
FIGURA 4: DIAGRAMA DE CLASSES.....	21
FIGURA 5: DIAGRAMA DE SEQÜÊNCIA – ADMINISTRADOR.....	22
FIGURA 6: DIAGRAMA DE SEQÜÊNCIA – MÉDICO.....	23
FIGURA 7: DIAGRAMA DE SEQÜÊNCIA - PACIENTE.....	24
FIGURA 8: MODELAGEM DO BANCO DE DADOS.....	25
FIGURA 9: DIAGRAMA DE NAVEGABILIDADE.....	26
FIGURA 10: MODELO CONCEITUAL DO SISCLIN.....	27
FIGURA 11: MODELO NAVEGACIONAL DO SISCLIN.....	29
FIGURA 12: CARTÃO ADV SISCLIN.....	30
FIGURA 13: CARTÃO DE CONFIGURAÇÃO - ADV MENU PRINCIPAL.....	30
FIGURA 14: CARTÃO DE CONFIGURAÇÃO - ADV MENU MEDICO.....	31
FIGURA 15: TELA INICIAL DO PHPMYADMIN.....	32
FIGURA 16: ARQUIVO CONF.PHP.....	33
FIGURA 17: CLASSE MYSQL.....	34
FIGURA 18: PÁGINA DE LOGIN DO ADMINISTRADOR.....	35
FIGURA 19: TELA DE ADMINISTRAÇÃO DO SISTEMA.....	36
FIGURA 20: PÁGINA INICIAL DO SISCLIN.....	37
FIGURA 21: MENU DO MÉDICO.....	38
FIGURA 22: VALIDAÇÃO DE USUÁRIO.....	38
FIGURA 23: TELA DE ADICIONANDO PACIENTE.....	39
FIGURA 24: FORMULÁRIO DE PESQUISA.....	40
FIGURA 25: EXEMPLO DE GRÁFICOS GERADOS PELO SISCLIN.....	41
FIGURA 26: CADASTRO DE SINTOMAS.....	43
FIGURA 27: TELA DE DIAGNÓSTICO.....	44

## **1 – INTRODUÇÃO**

Sisclin (sistema web para gerenciamento de uma clínica médica) é um sistema cujo objetivo é gerar uma base de conhecimento para informações integrando os cursos de saúde da Universidade Presidente Antônio Carlos.

Neste Capítulo será encontrada uma prévia sobre o Sisclin (seção 1.1), a proposta do trabalho (seção 1.2), os objetivos (seção 1.3) e as etapas do desenvolvimento do projeto (seção 1.4).

### **1.1 – CONTEXTO**

Os cursos da área médica na UNIPAC não possuem um sistema de integração onde os alunos e professores possam pesquisar e compartilhar dados de pacientes.

É necessário um sistema para gerenciar as consultas dividir informações entre os vários cursos de saúde da UNIPAC.

### **1.2 – PROPOSTA**

Este trabalho propõe a criação de um software baseado em sistema *web* para gerenciar uma clínica médica. Serão utilizadas informações na área de Inteligência Artificial sobre “Sistemas Especialistas”.

Na área de Engenharia de Software serão feitas as modelagens em vários níveis e será utilizado também conceitos de Hipermissão para criar o site onde o usuário poderá gerenciar o sistema.

A criação de um sistema *web* para este gerenciamento se justifica nos baixos custos de instalação e manutenção, na interface simplificada na interação homem e máquina e também na facilidade de acesso em qualquer computador conectado à internet.

### 1.3 – OBJETIVOS

O objetivo do Sisclin é gerar uma base de informações que poderá ser compartilhada entre os diversos cursos da área de saúde da UNIPAC. Esses cursos terão acesso a dados estatísticos sobre a ocorrência de doenças e gerenciamento de cadastro de pacientes e consultas em suas clínicas.

O usuário poderá acessar estatísticas sobre as doenças ocorridas em determinadas cidades ou pesquisar o número de doenças ocorridas em um intervalo de tempo.

O médico terá acesso ao histórico de doenças de seu paciente e também será desenvolvido um sistema especialista para ajudar o médico no diagnóstico de sua consulta a partir de dados cadastrados nos sintomas que o doente apresenta.

As ferramentas utilizadas para o desenvolvimento do Sisclin serão o HTML, a linguagem de programação PHP, o banco de dados MySQL e o editor *Macromedia Dreamweaver*.

O sisclin é desenvolvido no endereço: <http://www.sisclin.com.br/>

### 1.4 – ESTRUTURA DO TRABALHO

No capítulo 2 será feita uma revisão bibliográfica que apresentará alguns conceitos sobre o HTML, a linguagem de programação PHP, Sistemas Especialistas, Hipermídia e tópicos sobre segurança no PHP.

No capítulo 3 é apresentada a modelagem do sistema, a especificação da análise de requisitos, as modelagens e o porquê de sua escolha. Será apresentado o diagrama de casos de uso, o diagrama de classes, o diagrama de seqüência, o diagrama de estados, o diagrama de navegabilidade e a modelagem do banco de dados.

O capítulo 4 apresenta a modelagem da hipermídia com a modelagem conceitual, a modelagem navegacional e o projeto de interface abstrata.

No capítulo 5 a parte técnica é explicada, será visto a implementação do sistema, os principais comandos para a criação do sistema, algumas telas do sistema serão apresentadas para que se tenha uma idéia do sistema.

O capítulo seis apresenta a conclusão do sistema onde será mencionado o aprendizado adquirido e sugestões para novas versões e trabalhos futuros.

## 2 – REVISÃO BIBLIOGRÁFICA

Para o melhor entendimento do desenvolvimento do Sisclin são necessários diversos conceitos sobre as tecnologias envolvidas. As tecnologias serão brevemente descritas neste capítulo.

Na seção 2.1 será apresentado o HTML e sua utilização, a seção 2.2 será visto o PHP, sua história e algumas questões relevantes sobre segurança.

Na seção 2.3 serão citados conceitos sobre Sistemas Especialistas e na seção 2.4 conceitos sobre hipermídia. O banco de dados MySQL será apresentado na seção 2.5.

### 2.1 – HTML

HTML é uma linguagem de formatação de textos utilizada para definir páginas na Web, baseada em códigos embutidos em um documento que pode servir para ajustar fontes, criar listas, mostrar imagens, entre outros tipos de formatações de páginas. A linguagem HTML está fundamentada na ISO *Standard Generalized Markup Language* (ou SGML como é conhecida), que é um padrão internacional de formatação de documentos. Como um subconjunto deste padrão ISO, qualquer aplicação que possa interpretar o formato SGML poderá também ler o formato HTML.

Como o próprio nome diz, a HTML é uma linguagem de marcação hipertexto, e é considerada a “linguagem da Web”. Todos os documentos acessados na Web foram implementados em HTML por alguém. As formatações de páginas, as imagens coloridas, os “*hyperlinks*” que possibilitam “navegar” pelo mundo virtual, foram desenvolvidos utilizando HTML. Basicamente, os documentos escritos em HTML são arquivos no formato ASCII-texto. Assim, podem ser criados com a utilização de qualquer editor de texto que grave os arquivos em formato texto.



HTML é uma linguagem simples, porém poderosa e com muitos recursos. Um fator importante para o seu aprendizado é compreender seus conceitos, seus propósitos, e o que ela pode nos oferecer.

Algumas das principais características do HTML são:

- Formatação de documentos
- Organização de listas
- Capacidade de incluir hipertexto /hipermídia em documentos Web
- Capacidade de incluir imagens clicáveis.

O HTML faz uso de “*tags*” e elementos que proporcionam uma maneira de especificar qual formato utilizar e quando um formato começa e termina. Para isso, utiliza marcações específicas e distintas para dizer ao browser do usuário como exibir um documento. A sintaxe básica para estas marcações HTML geralmente são especificadas da seguinte maneira:

Portanto, a estrutura básica de um documento HTML deve conter:

- uma identificação do tipo HTML: `<HTML> </HTML>`
- uma seção de cabeçalho delimitada por: `<HEAD> </HEAD>`
- o título do documento dentro da seção de cabeçalho:  
`<TITLE>título</TITLE>`
- uma área como corpo do documento:  
`<BODY>corpo_do_conteúdo</BODY>`

## 2.2 – PHP

O PHP é a linguagem de desenvolvimento Web escrita por desenvolvedores Web e para desenvolvedores Web [1].

PHP significa: PHP: *Hypertext Preprocessor*. O produto foi originalmente chamado de *Personal Home Page Tools* e muitas pessoas ainda pensam que esse é o significado de acrônimo. Conforme ele se expandiu em objetivos, um nome novo e mais apropriado (embora recursivo como o GNU) foi selecionado através do voto da comunidade. Você pode utilizar qualquer extensão que desejar para designar

arquivos de PHP, mas os recomendados são .php, .php3 e .phtml. O PHP está atualmente em sua quinta versão PHP5 ou simplesmente PHP [1].

O PHP é uma linguagem de criação de scripts do lado servidor, que pode ser incorporada em HTML ou utilizada como um binário independente (embora a primeira utilização seja muito mais comum). Os produtos “proprietários” ou patenteados nesse setor do mercado são o ASP (*Active Server Pages*) da Microsoft, o ColdFusion da Macromedia e o JSP (*Java Server Pages*) da Sun [1].

Estritamente falando, o PHP tem pouco a ver com layout, eventos ou qualquer coisa relacionada à aparência de uma página Web. De fato, a maior parte do que o PHP realiza é invisível para o usuário final. Alguém visualizando uma página de PHP não será definitivamente capaz de afirmar que ela não foi escrita em HTML, porque o resultado final do PHP é HTML [1].

Um breve exemplo da utilização do HTML com PHP:

```
<html>
<head>
<title>Exemplo</title>
</head>
<body>
<?php echo "Olá, Eu sou um script PHP!"; ?>
</body>
</html>
```

O PHP é um módulo oficial do servidor http Apache, o servidor Web líder de mercado e gratuito, que executa aproximadamente 55% da *World Wide Web* (segundo a pesquisa sobre servidores Web da Netcraft, amplamente citada). Isso significa que o mecanismo de script do PHP pode ser construído no próprio servidor Web, tornando a manipulação de dados mais rápida. Como o servidor Apache, o PHP é inteiramente multiplataforma, significa que executa nativamente em várias versões de Unix, bem como no Windows e agora no Mac OS X. Todos os projetos sob a égide da *Apache software Foundation* – incluindo o PHP – são softwares de código-fonte aberto [1].

Hoje mais de 18.000.000 de domínios utilizam a linguagem PHP em sua programação. Este crescimento é justificado por várias vantagens que a linguagem PHP apresenta, entre elas são citadas [1]:

O PHP não custa nada. Até recentemente acreditava-se que softwares que não sustentavam nada geralmente dividiam-se em três categorias [1]:

- programas atendendo a pequenos setores não comerciais.
- programas fazendo trabalho pesado de baixo nível.
- programas para pessoas com “questões” sociopolíticas peculiares.

Estamos claramente em meio a um turbilhão de mudanças no negócio de software. Hoje, muitos dos mais importantes dos softwares para o consumidor são distribuídos sem custo: clientes de correio eletrônico, navegadores Web, jogos e até conjuntos completos de serviços de escritório estão sendo distribuídos tão rapidamente quanto seus criadores podem lançar versões Web ou configurar servidores de FTP. O software para o consumidor final é visto, cada vez mais, como uma forma de vender de servidor, sistemas operacionais, conectividade, publicidade, dispositivos opcionais ou ações. Portanto, o preço de varejo total de um software não é mais uma medida confiável de sua qualidade [1].

O PHP é fácil de aprender, em comparação com outras maneiras de se obter funcionalidades semelhantes. Diferentemente das páginas de servidor Java ou CGI baseado em C, o PHP não exige que você possua uma compreensão profunda das mais importantes linguagens de programação antes de poder criar um banco de dados ou uma chamada de servidor remoto triviais. Ao contrário do Perl, o PHP tem uma sintaxe fácil de analisar e de fácil utilização. E, ao contrário do ASP.NET, o PHP é estável e está pronto para resolver seus problemas hoje [1].

O PHP executa nativamente em todas as versões populares do Unix, e Windows. Uma enorme porcentagem de servidores de http em todo o mundo executa uma dessas duas classes de sistema operacional [1].

O PHP é compatível com os três mais importantes servidores *web*: o Apache para Unix e Windows, o *Microsoft Internet Information Server* e o *Netscape Enterprise Server* e também com vários servidores menos conhecidos [1].

A segurança é um dos fatores primordiais para o programador, um script mal desenvolvido pode abrir brechas no servidor tornando informações sigilosas disponíveis na web para milhares de visitantes.

A flexibilidade do PHP o torna uma ferramenta útil ao desenvolvimento de sistemas para internet, mas esta mesma flexibilidade pode ser usada de forma inadequada por usuários mal intencionados e proporcionar vulnerabilidades no servidor.

É fundamental restringir ao máximo a visualização das informações que trafegam pelos cabeçalhos HTTP, e certificar-se de que estas sejam postadas de um lado e recebidas de outro dentro de critérios definidos pelo programador. A má utilização das funções `include()`, `require()` e `fopen()` além da configuração das variáveis globais no servidor é fundamental para manter um nível de segurança adequado.

### **2.2.1 – Requisitos de segurança**

Em conjunto com a identificação dos riscos conhecidos relacionados com a segurança de um aplicativo específico ou um gênero de aplicativo, os desenvolvedores devem avaliar os requisitos de segurança para o aplicativo. Essa análise deve chegar a uma análise equilibrada do nível de segurança necessário para o aplicativo. Ela não tem de ponderar os extremos do espectro de segurança. Tendo o entendimento de que segurança verdadeira e garantida não existe, proteger contra riscos conhecidos e minimizar o número de ataques bem-sucedidos e seus efeitos é geralmente um nível aceitável de segurança. As pessoas envolvidas com o ciclo de desenvolvimento do aplicativo devem determinar seu próprio nível de segurança “aceitável”, examinando os riscos conhecidos, os objetivos do aplicativo e os métodos utilizados para implementar o nível de segurança desejado [2].

Para chegar aos requisitos de segurança, gerentes e desenvolvedores podem achar útil se concentrar nas seguintes áreas de risco comumente conhecidas [2]:

- Controle de autenticação e de acesso do usuário
- Armazenamento de dados de informações confidenciais
- Segurança em comunicações externas de rede
- Segurança de pontos de entrada para aplicativos externos e sistema operacional

Dessas quatro áreas gerais projetistas e desenvolvedores de aplicativos podem identificar um conjunto mínimo de recursos importantes para análise. Dependendo da funcionalidade do aplicativo, algumas áreas de risco são mais pertinentes que outras [2].

### **2.2.2 – Requisitos de avaliação de autenticação de controle de acesso**

Autenticação de usuários é frequentemente tratada pelo sistema operacional no qual o aplicativo é executado, mas varias classes de aplicativos podem ser necessárias para lidar com a própria autenticação. Aplicativos embutidos, aplicativos que funcionam independentemente do sistema operacional e aplicativos da Web distribuídos frequentemente precisam acomodar algum nível de autenticação de usuário e controle de acesso. Exemplos comuns de aplicativos que necessitam desses métodos de segurança são os aplicativos de comercio da internet, nos quais os usuários fazem compras por meio de um site da Web ou via acesso ao banco de dados de clientes. Em ambos os casos, a possibilidade de vários usuários ou grupos diferentes de usuários utilizarem o sistema exige um controle rigoroso dos dados acessíveis. Os aplicativos precisam de métodos para permitir que usuários separados acessem os sistemas por meio de um método de login; eles também precisam de restrições relacionadas aos dados financeiros dos respectivos usuários individuais [2].

Para determinar a autenticação e requisitos de controle de acesso, os projetistas devem examinar as interações dentro do aplicativo e com o mundo que o cercam. Isso inclui os métodos pelo quais os usuários acessam o aplicativo – acessar diretamente por meio de um terminal fixo ou acessar pela rede são dois métodos de autenticação do sistema operacional no qual ele é executado. Os aplicativos de rede que são acessados por múltiplos usuários simultaneamente ou por meio dos quais os usuários acessam dados de bancos de dados comuns fornecem um ímpeto forte para o controle de acesso e autenticação. O nível de granularidade e flexibilidade do controle de acesso e das capacidades de autenticação fornecida pelo sistema operacional ajudam a determinar se métodos

proprietários devem ser envolvidos. Aplicativos independentes ou embutidos frequentemente são desenvolvidos a partir do zero e, portanto, necessitam de seus próprios métodos especializados [2].

### 2.2.3 – Vulnerabilidades no PHP

Foram descobertas duas vulnerabilidades em scripts PHP. A primeira foi o problema de validação de entradas típico que infestavam muitos dos primeiros scripts e permitia que atacantes lessem qualquer arquivo do sistema [3].

A segunda foi descoberta em Abril de 1997 pelo grupo *Secure Networks Inc.* A vulnerabilidade descoberta era uma condição de estouro de buffer no `php.cgi` das versões 2.0beta10 e anteriores do servidor HTTPD do NCSA. O problema ocorre quando um atacante passa uma string grande (derivada de parâmetros do script) para a função `FixFilename()` e sobrescreve a pilha da máquina, permitindo que código arbitrário seja executado no sistema local [3].

Há duas maneiras de impedir a exploração das vulnerabilidades de scripts PHP [3]:

- Remover os scripts vulneráveis, ou
- Atualizar para a última versão do PHP, que corrige o problema.

### 2.2.4 – Usando PHP de modo seguro

PHP pode ser configurado de forma que execute scripts em um ambiente restrito para diminuir o dano que pode ser infligido por programas inseguros. Este modo de operação pode ser chamado de *'safe mode'*. A diretiva de configuração `safe_mode` no `php.ini` ativa e desativa o safe mode. A diretiva `safe_mode_exec_dir` especifica um diretório onde os scripts podem ser carregados. PHP não irá executar um script se ele não estiver neste diretório. Além disto, PHP não irá deixar um script chamar outro programa que não esteja neste diretório. Deste modo, mesmo se existir um buraco na segurança no script que permita invasores executarem

arbitrariamente comandos no script, eles irão estar limitados aos que estejam no diretório de executáveis do *safe mode* [4].

Para prevenir ocupação de variáveis de ambiente, o *safe mode* faz uso de outra configuração no `php.ini` que restringe a habilidade do usuário para modificá-las. O campo `safe_mode_allowed_env_vars` contém uma lista de prefixos que identificam os nomes de variáveis de ambiente que o usuário tem permissão de modificar. Deste modo, quaisquer variáveis de ambiente cujos nomes comecem com algo não listado no `safe_mode_allowed_env_vars` não pode ser alterado de dentro de um script PHP. A lista padrão consiste no prefixo "PHP\_" apenas. Como nos temos visto, algumas das variáveis PHP\_ contêm informações sensíveis, então esta restrição nem sempre resolve o problema completamente [4].

### **2.3 – SISTEMAS ESPECIALISTAS**

Sistema Especialista é uma aplicação da IA. São chamados assim porque contêm o conhecimento de um especialista na área de aplicação dentro de um programa [5].

O Sistema Especialista é ferramenta fundamental para áreas como indústria, educação, medicina, comércio, finanças e jurídica, ou seja, em quase todos os segmentos onde se requer um especialista. Sua utilização destaca-se, especialmente, em sistemas de apoio à decisão [5].

As aplicações resultam em projeções, diagnósticos, prognósticos, monitoração, simulação, manutenção, treinamento e controle [5].

Programas de computador que tentam resolver problemas que os seres humanos resolveriam emulando o raciocínio de um especialista, aplicando conhecimentos específicos e inferências são ditos Sistemas Especialistas [5].

Sistema Convencional é baseado em um algoritmo, emite um resultado final correto e processa um volume de dados de maneira repetitiva enquanto que um Sistema Especialista é baseado em uma busca heurística e trabalha com problemas para os quais não existe uma solução convencional organizada de forma algorítmica disponível ou é muito demorada [6].

## 2.4 – HIPERMÍDIA

Hipertexto é um aplicativo que permite criar, manter e manipular trechos de informação (textos e gráficos) interligados de forma não seqüência ou não-linear.

Multimídia é a combinação de textos, gráficos, desenhos, imagens, áudio e vídeo em um conjunto ou apresentação exclusiva de computador [7].

Hipermídia: junção dos tipos de dados da multimídia com os mecanismos e semânticas dos hipertextos, ou seja, hipermídia constitui um sistema ou aplicativo hipertexto que, além de textos e gráficos, suporta outros tipos de mídia, tais como desenhos, imagem, som ou vídeo [7].

A noção de hipermídia esta ligada a uma maneira de apresentar e recuperar informações de forma não linear, segundo um modelo de redes - nós interconectados por ligações. Estes nós podem conter tanto informações de textos e gráficos, como informações de outro tipo, tais como imagens, sons digitalizados, seqüência de animação e vídeo interativo. Um dos principais fatores para a popularização de aplicativos hipermídia é a possibilidade de o usuário poder "navegar" sobre um conjunto de unidades de informação [7].

## 2.5 – MYSQL

O MySQL tornou-se o banco de dados *open source* mais popular do mundo porque as empresas pretendem reduzir os seus custos de operação e estão usando para comoditizar adicionalmente sua infra-estrutura de TI [8].

Depois de obter uma grande eficiência no uso de programas *open source*, como Linux e Apache, muitas companhias estão visando sua infra-estrutura de banco de dados como o próximo alvo de economia de custos pela comoditização (*commoditization*). Realmente, disseram alguns gerentes de TI estar explorando o



uso de um pacote inteiro de softwares *open source* conhecido como LAMP (Linux, Apache, MySQL, PHP) como uma forma de aumentar a eficiência operacional [8].

A Web tem se tornado de fato a interface para distribuição de aplicativos; quase todos os aplicativos Web utilizam banco de dados. Estes aplicativos são geralmente públicos e implementados para fornecer um novo canal para vendas ou para reduzir custos de serviços. Eles devem ser robustos e permitirem suporte a picos de carga sem atraso; acomodar mudanças de frequência e integrar facilmente a sistemas de *back-end* [8].

MySQL é ideal para aplicações Web e conduz alguns dos principais sites de Internet do mundo. De fato, a divisão de finanças e outras divisões do *Yahoo!* usam o MySQL para implementar diversos recursos do site. A *Travelocity* usa o MySQL para busca de passagens aéreas mais baratas. Interfaces com base na Web estão em uso crescente para aplicações internas das empresas, principalmente portais corporativos e aplicações de Internet [8].

### **3 – MODELAGEM DO SISTEMA**

Modelagem de software é a atividade de construir modelos que expliquem características ou o comportamento de um software ou de um sistema de software. Na construção do software os modelos podem ser usados na identificação das características e funcionalidades que o software deverá prover (análise de requisitos), e no planejamento de sua construção [9].

Frequentemente a modelagem de software usa algum tipo de notação gráfica e são apoiados pelo uso de Ferramentas CASE [9].

A modelagem de softwares normalmente implica na construção de modelos gráficos que simbolizam os artefatos de software utilizados e seus inter-relacionamentos. Uma forma comum de modelagem de programas procedurais (não orientados a objeto) é através de fluxogramas, enquanto que a modelagem de programas orientados a objeto normalmente usam a linguagem gráfica UML [9].

A modelagem é uma parte central de todas as atividades que levam à implantação de um bom software. Construimos modelos para comunicar a estrutura e o comportamento desejados do sistema. Construimos modelos para visualizar e controlar a arquitetura do sistema. Construimos modelos para compreender melhor o sistema que estamos elaborando, muitas vezes expõe oportunidade de simplificação e reaproveitamento. Construimos modelos para gerenciar os riscos [10].

### **3.1 – ANÁLISE DE REQUISITOS**

Em um processo de software, a engenharia de requisitos é a primeira atividade importante, após a conclusão de necessidades de um processo de pré desenvolvimento. A engenharia de requisitos é definida em função de suas atividades principais: entendimento dos problemas, determinação de soluções e especificação de uma solução que é estável, compreensível, manutenível e que satisfaça às diretrizes de qualidade do projeto [11].

O Sisclin possui a seguinte análise de requisitos:

O sistema permitirá ao administrador cadastrar clínicas, especialidades, anamneses e médicos onde cada médico pertencerá a uma determinada clínica universitária. O acesso do médico será feito a partir de um nome de usuário e senha cadastrados pelo administrador.

O médico será responsável por cadastrar novos pacientes o cadastro do paciente terá os seguintes dados: nome, idade, data de nascimento, sexo, nome do pai, profissão do pai, idade do pai, nome da mãe, profissão da mãe, idade da mãe, endereço, telefone, data de cadastro e motivo da consulta.

O cadastro da consulta pelo médico deverá ter os seguintes dados: data, hora, e médico que está realizando a consulta. O médico ao atender o paciente deverá utilizar uma anamnese correspondente à sua área para auxiliar no atendimento. Na consulta o médico deverá cadastrar também o diagnóstico, receita e sintomas.

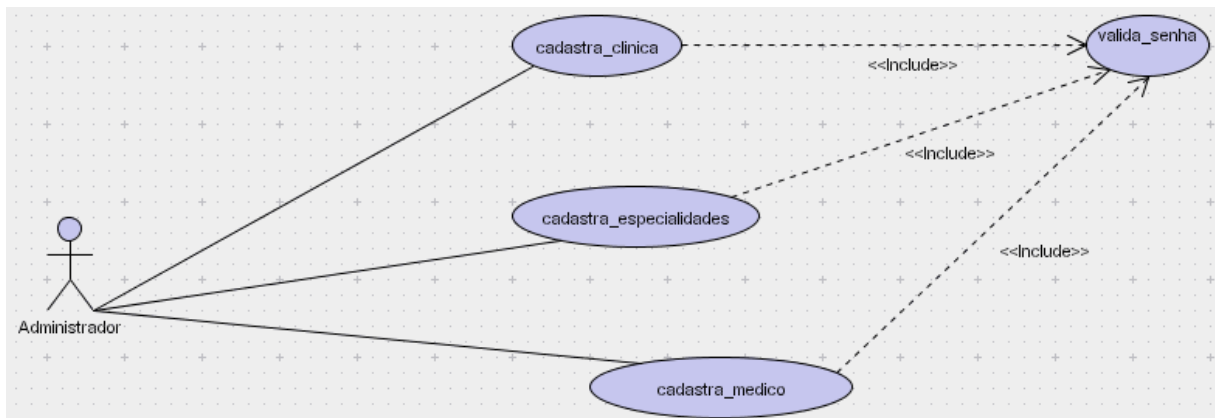
O Sisclin deverá auxiliar o atendimento médico buscando sintomas parecidos para determinada doença e listar as possíveis doenças que o paciente possui a partir de um sistema especialista.

O médico terá acesso ao histórico de todos os pacientes cadastrados no sistema. Esse histórico terá os dados cadastrados pelo médico e as consultas já realizadas com diagnósticos, receitas e sintomas do paciente.

O Sisclin irá gerar dados estatísticos e gráficos relevantes aos médicos, como número de ocorrências de determinadas doenças em uma região, por exemplo.

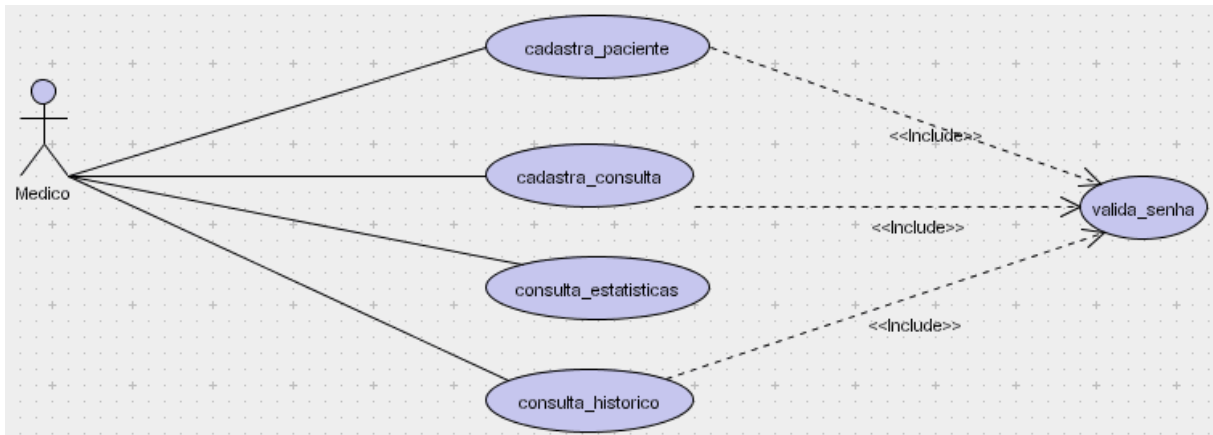
### 3.2 – DIAGRAMA DE CASOS DE USO

Os diagramas de casos de uso são importantes para visualizar, especificar e documentar o comportamento de um elemento. Esses diagramas fazem com que sistemas, subsistemas e classes fiquem acessíveis e compreensíveis, por apresentarem uma visão externa sobre como esses elementos podem ser utilizados no contexto [10].



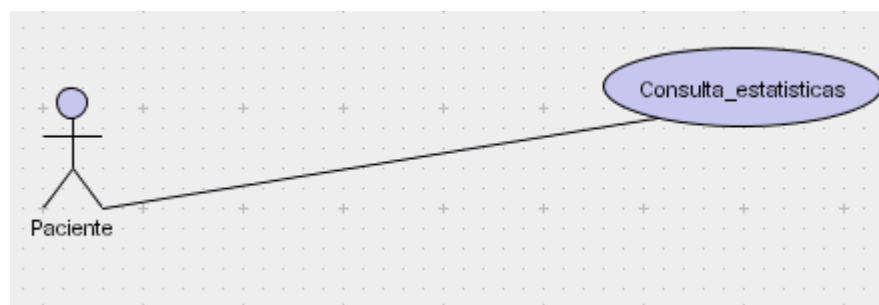
**Figura 1: Casos de uso: Cenário Administrador**

A figura 1 mostra o Diagrama de casos de uso para o Cenário Administrador. Aqui estão descritas as possíveis ações para o Administrador. O administrador terá acesso ao cadastro de clínicas, especialidades de médicos e o cadastro de médicos. O administrador deverá ter um nome de usuário e uma senha para acessar o Sisclin. As senhas que os médicos utilizarão serão cadastradas pelo administrador do sistema.



**Figura 2: Casos de uso: Cenário Médico**

A figura 2 mostra o Diagrama de casos de uso para o Cenário Médico. O médico terá acesso a cadastrar um novo paciente, cadastrar uma nova consulta, consultar o histórico do paciente e consulta às estatísticas do sistema. O cadastro de consulta será auxiliado por um sistema especialista e a consulta ao histórico trará o resultado de todas as consultas já realizadas pelo paciente.



**Figura 3: Casos de uso: Cenário Paciente.**

A figura 3 apresenta o Diagrama de casos de uso para o cenário Paciente. O paciente assim como os demais visitantes do Sisclin terão acesso às estatísticas do sistema. As estatísticas serão gráficos onde o usuário poderá pesquisar a ocorrência de doenças em determinado intervalo de tempo ou pesquisar as doenças ocorridas em uma região.

### 3.3 – DIAGRAMA DE CLASSES

Os diagramas de classes são os diagramas encontrados com maior frequência na modelagem de sistemas orientados a objetos. Um diagrama de

classes mostra um conjunto de classes, interfaces e colaborações e seus relacionamentos [10].

Os diagramas de classes são importantes não só para a visualização , a especificação e a documentação de modelos estruturais, mas também para a construção de sistemas executáveis por intermédio de engenharia de produção reversa [10].

A figura 4 demonstra o diagrama de classes, os atributos existentes e métodos do SISClin.

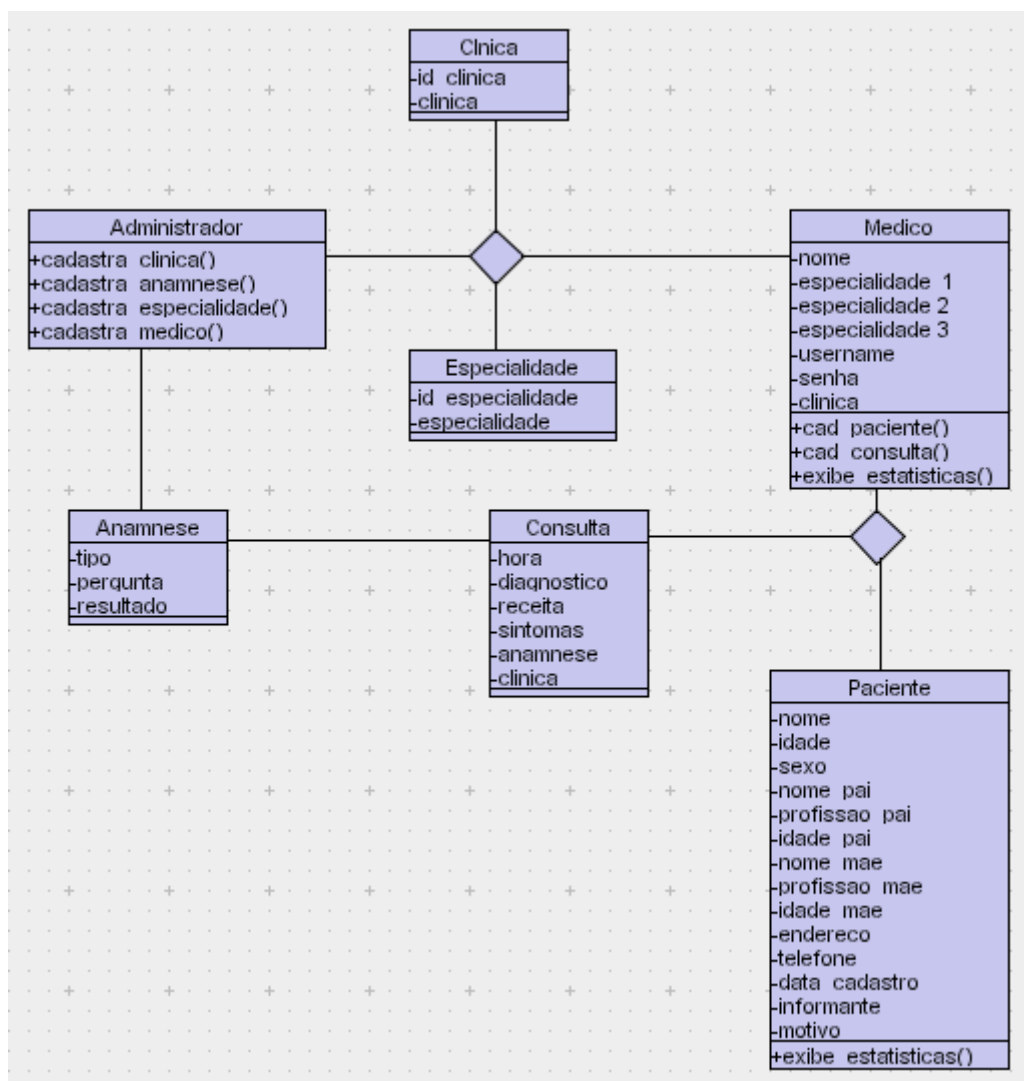
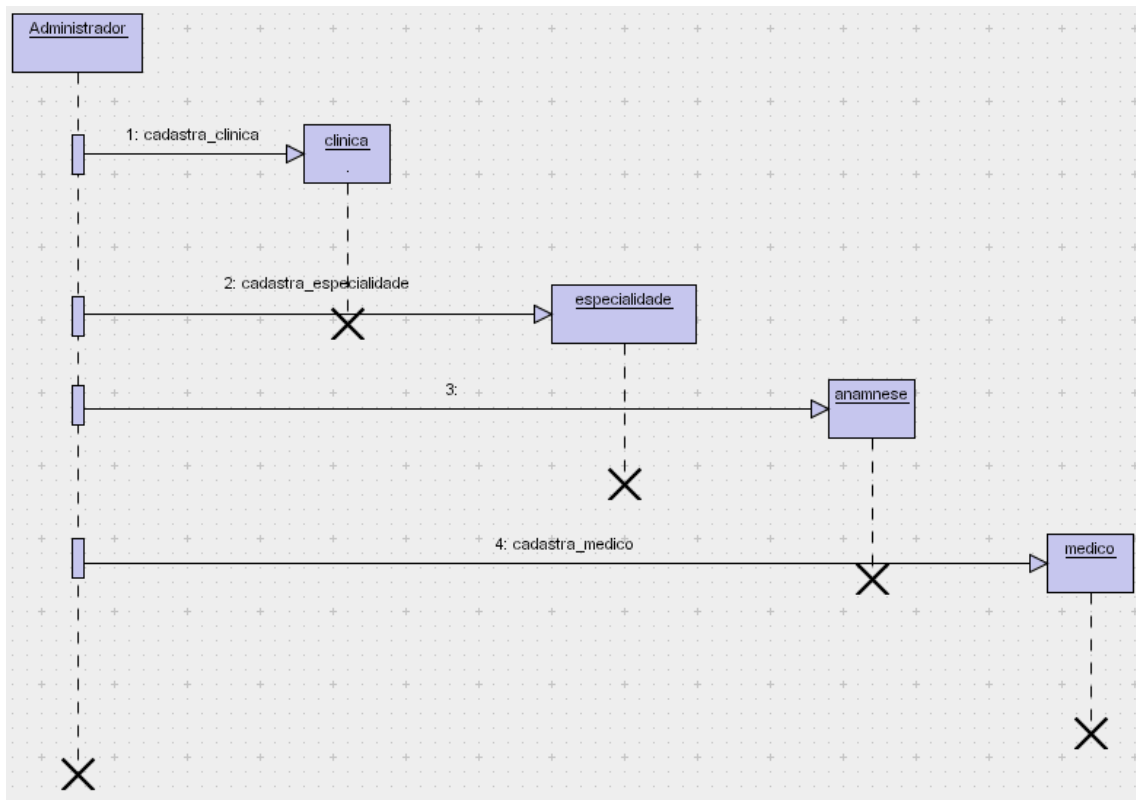


Figura 4: Diagrama de Classes

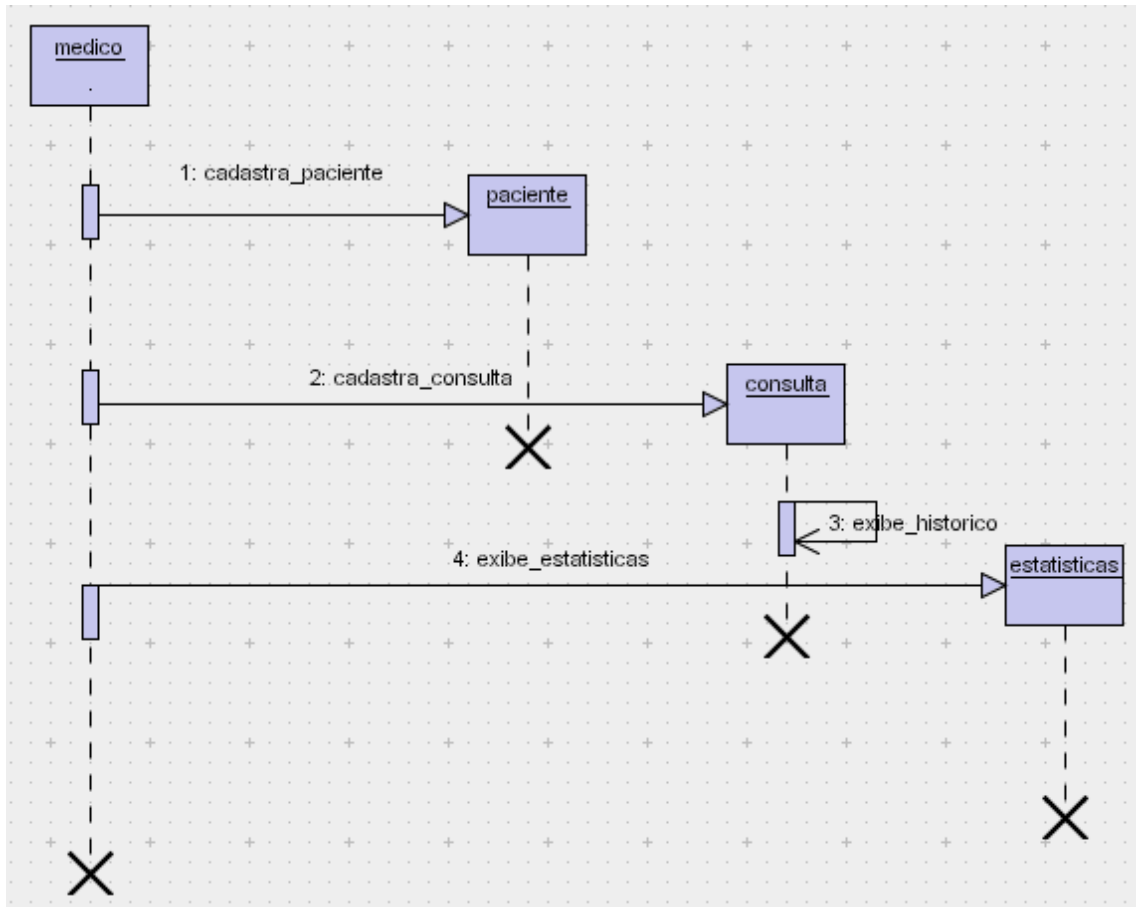
### 3.4 – DIAGRAMA DE SEQÜÊNCIA

Os diagramas de sequencia são utilizados para a modelagem dos aspectos dinamicos do sistema. Um diagrama de interação mostra uma interação, formada por um conjunto de objetos e seus relacionamentos, incluindo as mensagens que poderão ser enviadas entre eles. Um diagrama de sequencias é um diagrama de interação que dá ênfase à ordenação temporal das mensagens [10].



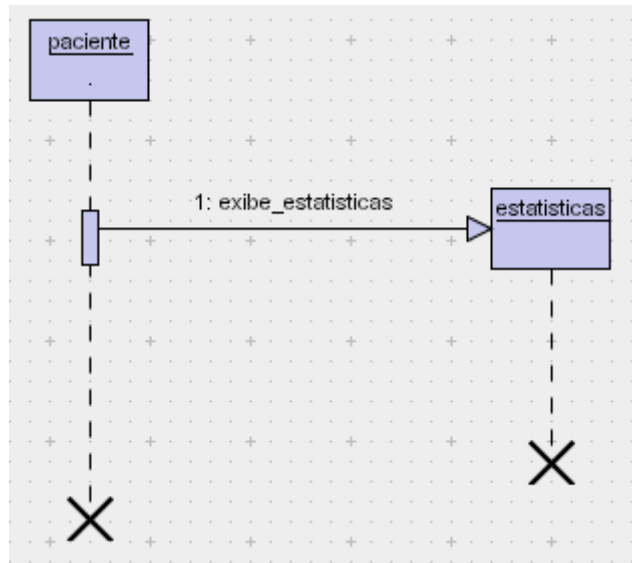
**Figura 5: Diagrama de Seqüência – Administrador**

A figura 5 demonstra a sequencia ser descrita pelo administrador do sistema, este terá opções para adicionar clinica, cadastrar especialidades de médicos, cadastrar anamneses e cadastrar médicos, nesta ordem.



**Figura 6: Diagrama de Seqüência – Médico**

A figura 6 mostra o diagrama de sequencia a ser descrito pelo médico. O médico deve inicialmente cadastrar o paciente, depois cadastrar a consulta referente ao paciente. Após cadastrada a consulta ele tem a opções para exibir histórico do paciente e por último ele poderá acessar as estatísticas do sistema.



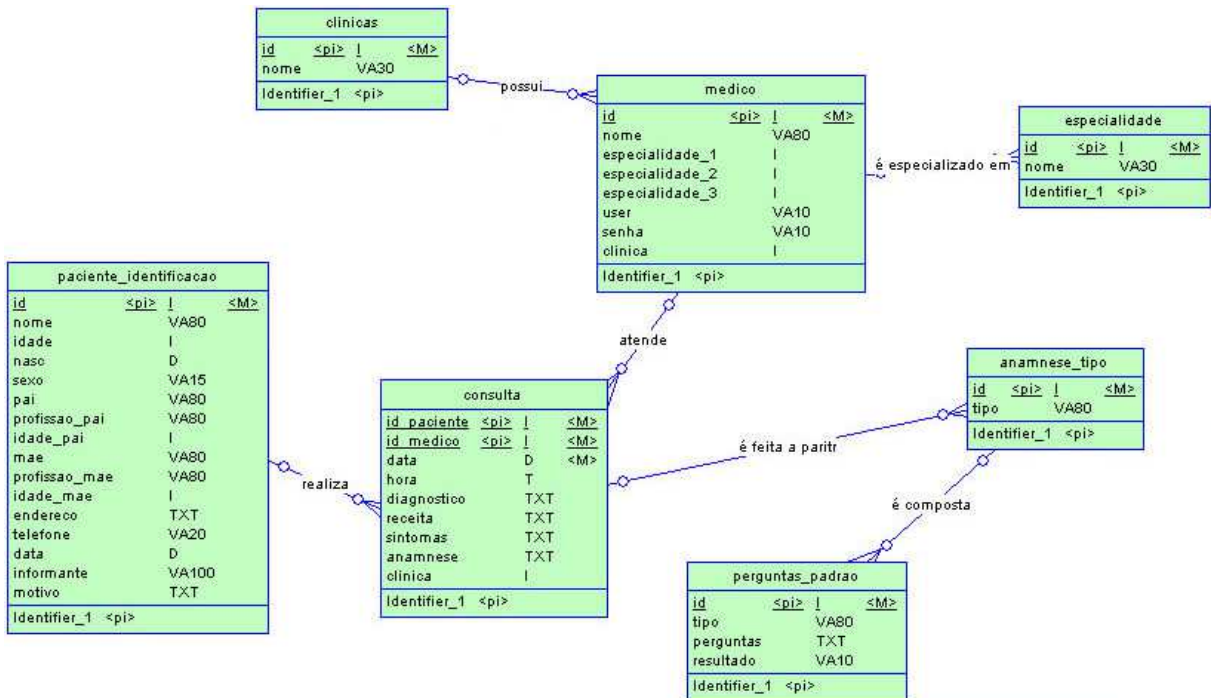
**Figura 7: Diagrama de Seqüência - Paciente**

A figura 7 mostra o diagrama de sequencia a ser descrito pelo paciente. A única ação disponível para o paciente é visualizar as estatisticas do site. Esta ação está disponível para todos os visitantes do SISClin.

### 3.5 – MODELAGEM DO BANCO DE DADOS

A modelagem de dados é a criação de uma estrutura que representa um conjunto de informações. Esta estrutura permite ao usuário recuperar dados de forma eficiente e rápida.





**Figura 8: Modelagem do banco de dados**

### 3.7 – DIAGRAMA DE NAVEGABILIDADE

O diagrama de navegabilidade consiste no caminho que o usuário do sistema deverá fazer para chegar a um determinado ponto do Sisclin. A figura 9 mostra os caminhos a serem seguidos pelos usuários do sistema. A partir de qualquer ponto da página o usuário terá acesso aos botões para ir para Página inicial, Sisclin, Login, Desenvolvimento, Contato e Estatísticas.

## Diagrama de navegabilidade

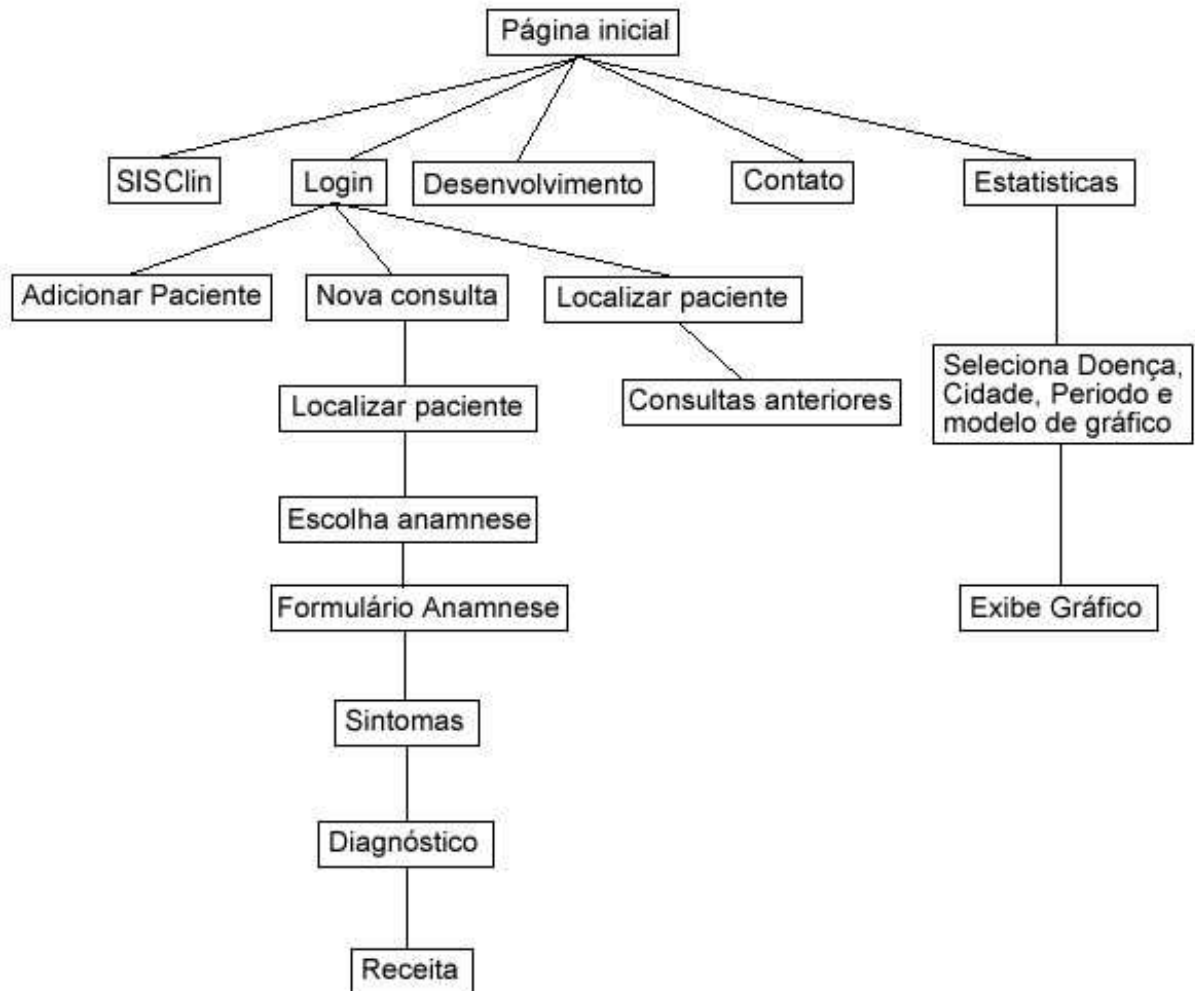


Figura 9: Diagrama de Navegabilidade

## 4 – MODELAGENS DA HIPERMÍDIA

O projeto de uma aplicação hipermídia deve cuidar da hiperbase, das estruturas de acesso e de uma interface com o usuário, de forma diferenciada, mas coordenada [7]. São quatro, os passos que fazem parte do processo de construção de uma aplicação hipermídia [7]: modelagem conceitual; modelagem navegacional; design abstrato da interface e implementação específica.

A escolha do modelo OOHDM foi feita levando em consideração a modelagem clara e simples e a facilidade para criar o projeto de interface abstrata.

### 4.1 – MODELO CONCEITUAL

Esta etapa da modelagem consiste na construção do domínio da aplicação e produz esquema de classes e relacionamentos [12].

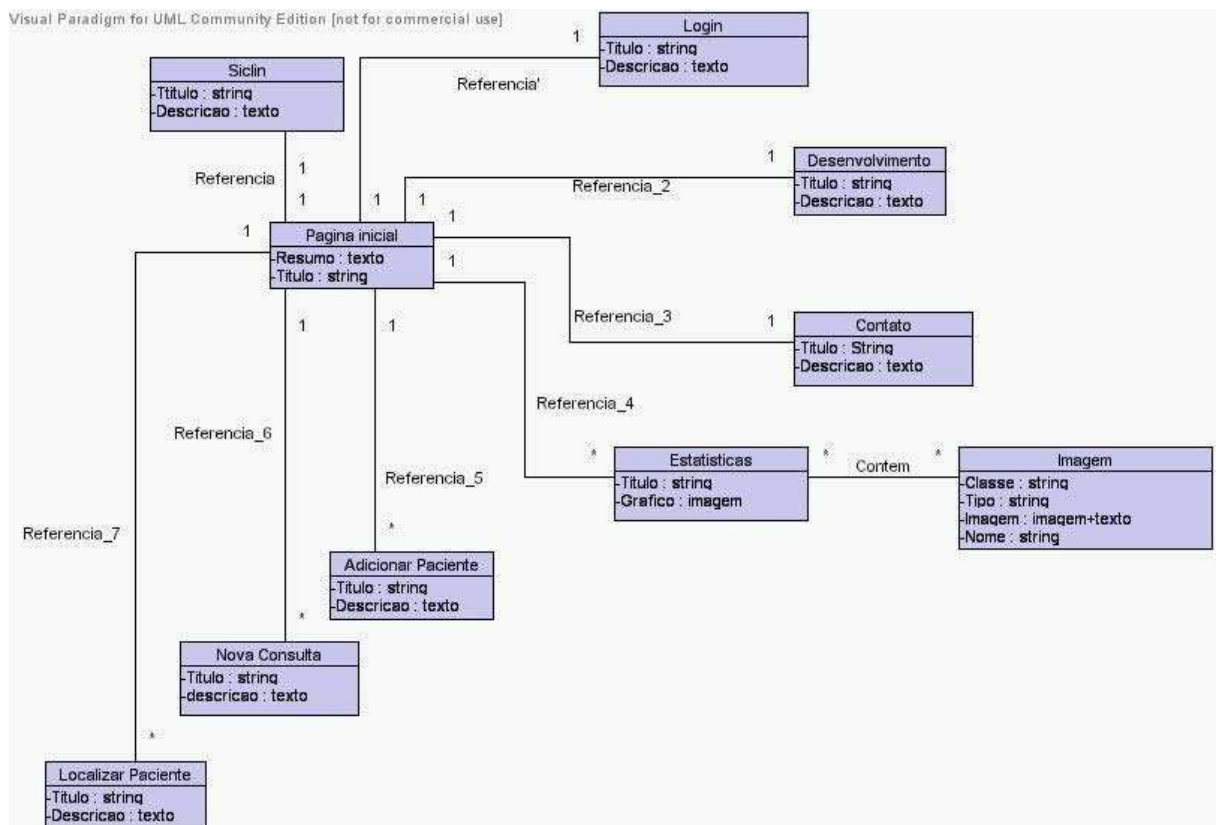


Figura 10: Modelo Conceitual do SISClin

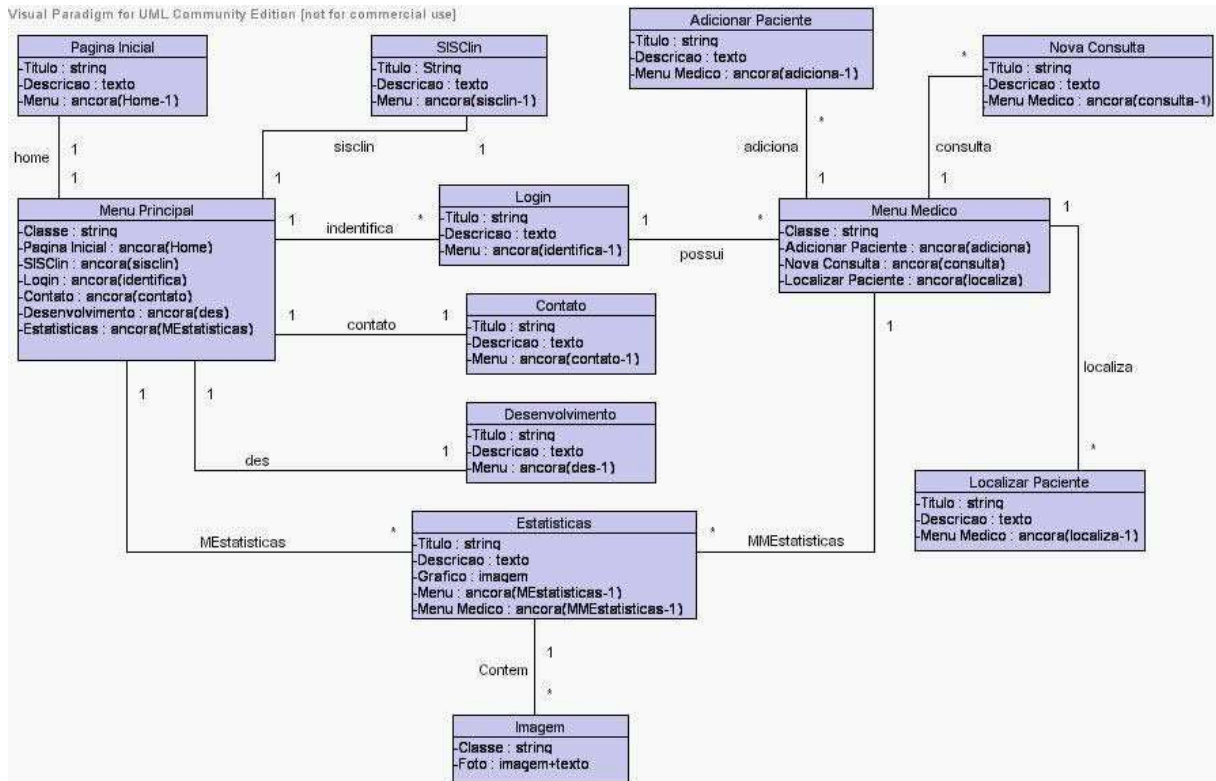
A figura 10 apresenta o modelo conceitual utilizado pelo Sisclin, a classe Sisclin irá conter uma descrição do sistema, seu objetivo e suas funcionalidades. A classe página inicial contém uma breve apresentação do sistema e é o ponto de partida para a navegação na hipermídia.

As classes Localizar Paciente, Nova consulta e Adicionar Paciente estarão disponíveis apenas para o médico e será necessário o login para sua visualização. O login pode ser feito a partir da classe login.

A classe Estatísticas contém os gráficos gerados pelo sistema. A classe desenvolvimento trata o desenvolvimento do sisclin. A classe contato será o local onde o visitante poderá entrar em contato com o administrador do site.

## **4.2 – MODELO NAVEGACIONAL**

Esta etapa consiste em um projeto de como o domínio será percorrido pelo leitor. Nele, as instâncias de classes tornam-se nós e os relacionamentos tornam-se elos. O sentido do esquema navegacional é o mesmo do esquema conceitual. O navegacional possui algumas classes a mais que o conceitual, onde estão, em sua maior parte, classes de menus, sub-menus e índices, o que permite mais facilmente, entender como navegar no sistema [12].



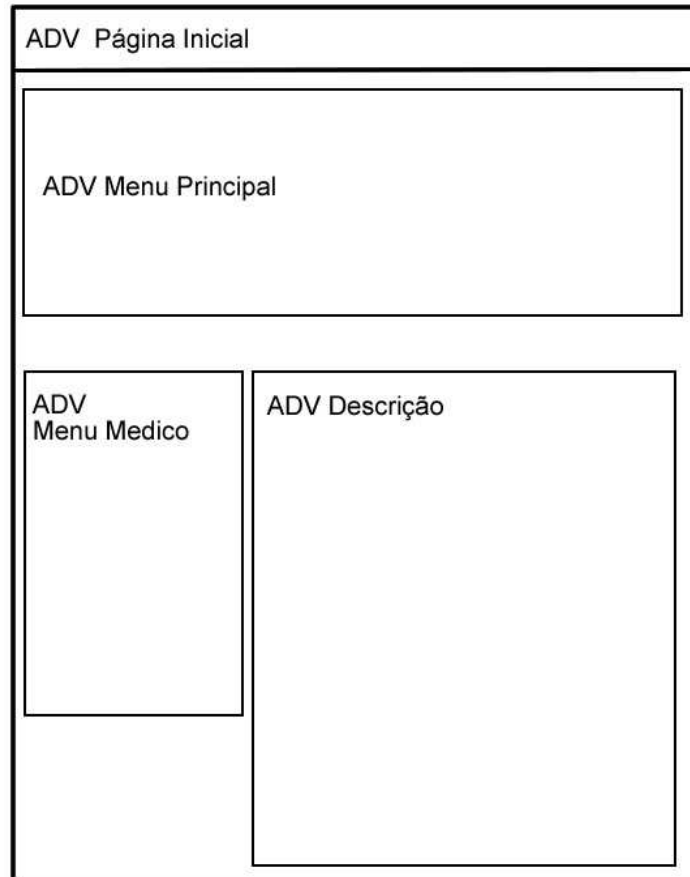
**Figura 11: Modelo navegacional do sisclin**

A figura 11 mostra o modelo navegacional do sisclin, nele foram adicionadas além das classes do modelo conceitual, os nós Menu principal e Menu Medico. Através destes nós o visitante poderá navegar através da hipermídia.

### 4.3 – PROJETO DE INTERFACE ABSTRATA

O Projeto de interface abstrata define como a navegação será mostrada ao visitante do site através de cartões ADV (*abstract data view*) e a criação de seus diagramas de configuração.

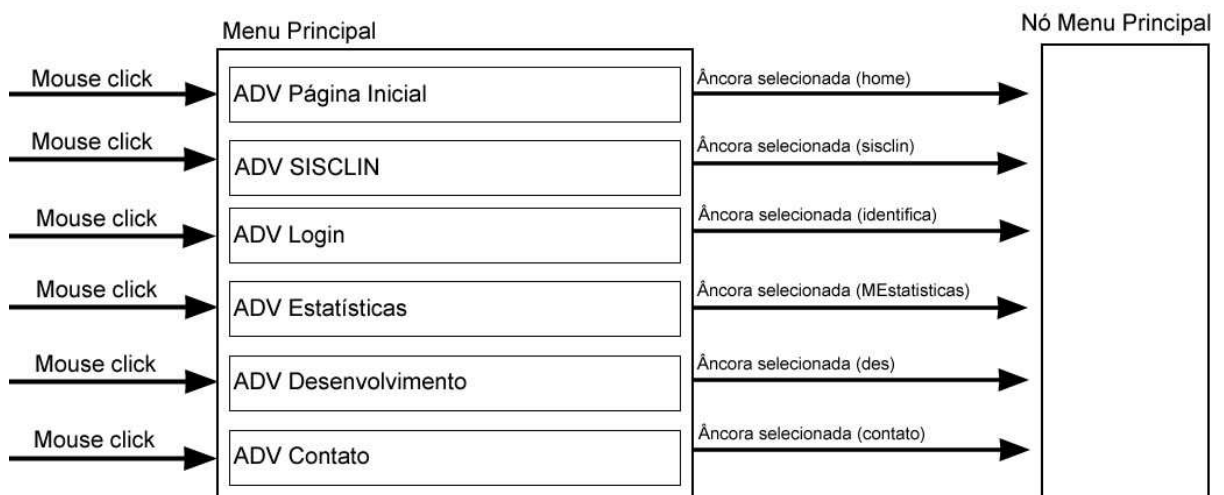
A figura 12 apresenta o cartão ADV das páginas do sisclin. Todas as páginas seguirão o mesmo modelo apresentado.



**Figura 12: Cartão ADV Sisclin**

No diagrama de configuração do Cartão ADV são representadas as ações que ocorrem no ADV. O diagrama de configuração do ADV e o cartão ADV descrevem a ação que o sistema executará para cada ação do usuário [12].

As figuras 13 e 14 apresentam o cartão de configuração do ADV Menu Principal e Menu Medico respectivamente.



**Figura 13: Cartão de configuração - ADV menu principal**



**Figura 14: Cartão de Configuração - ADV Menu Medico**

## 5 – IMPLEMENTAÇÃO

Este capítulo trata a implementação do sisclin, na seção 5.1 será visto como foi realizada a implementação e algumas telas do sistema.

### 5.1 – IMPLEMENTAÇÃO

Esta seção tem por objetivo mostrar tópicos sobre a implementação do Sisclin. O Sisclin foi desenvolvido a partir das ações do Administrador, logo depois desenvolvido as ações do médico e por último a criação das estatísticas.

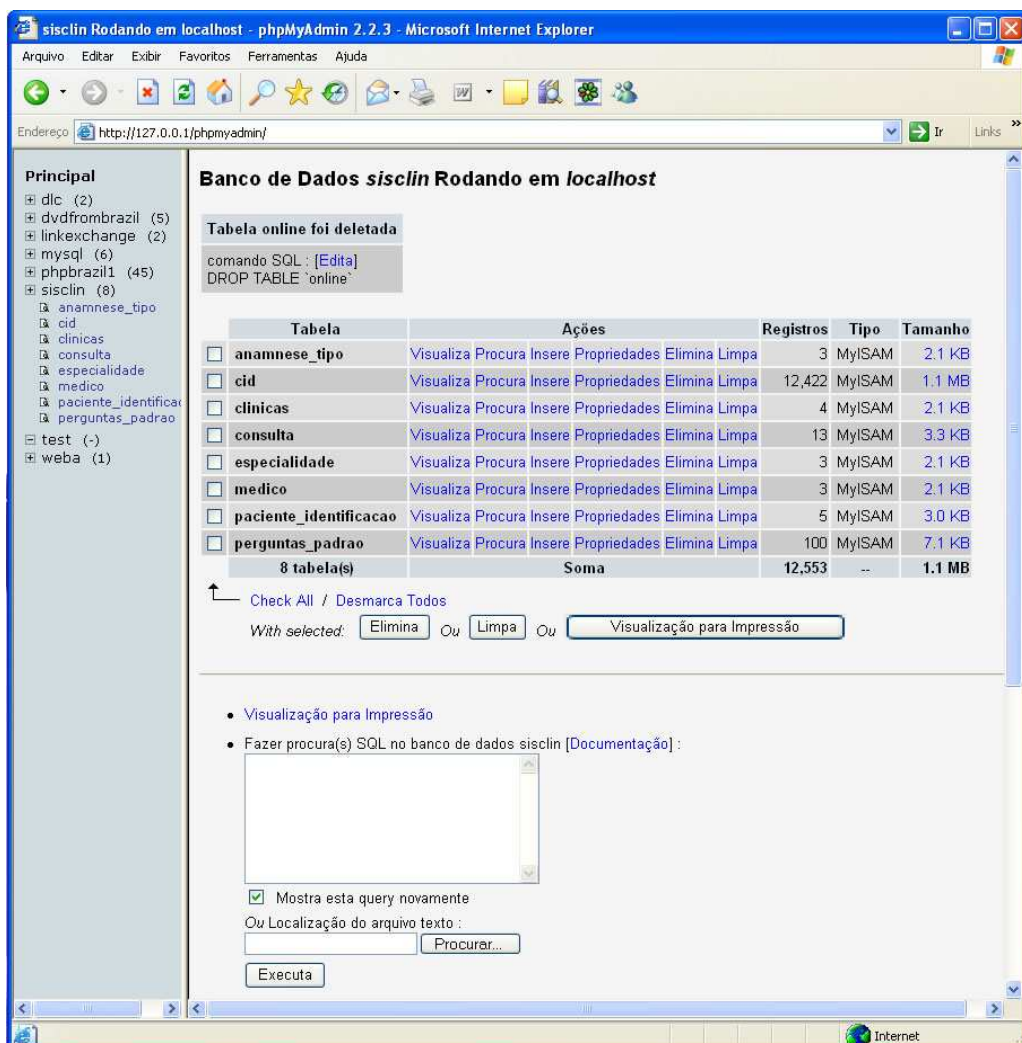


Figura 15: Tela inicial do phpmyadmin



Para o manuseio do mysql foi utilizado um script chamado *phpmyadmin*. O *phpmyadmin* tem como principal vantagem a fácil criação de tabelas, pesquisas no banco de dados e alterações em dados e tipos de dados. A figura 15 mostra a manutenção do banco de dados *sisclin* no *phpmyadmin*.

Após a criação do banco de dados no *sisclin* no *MySQL*. Foi criado um arquivo de classes e configurações do sistema. O arquivo de configuração do sistema possui as configurações para o *mysql* e nomes e senhas para administradores do sistema (figura 16).

```
8 // Configuração do MySQL //
9 $user_db = "webmaster";
10 $pass_db = "4321we";
11 $host_db = "localhost";
12 $db = "sisclin";
13
14 // Configuração da senha de administradores do sistema //
15 // Vetor com os nomes de usuários e senhas com acesso a administração. //
16 $user_login[0]="admin";
17 $pass_login[0]="12345";
18 $user_login[1]="epifanio";
19 $pass_login[1]="21epi";
```

**Figura 16: arquivo conf.php**

O nome e senha do administrador são salvos em um vetor, deste modo o *Sisclin* pode ter quantos administradores forem necessários com seus respectivos nomes de usuário e senha.

As configurações do *mysql* estão dispostas entre as linhas 8 e 13. Na linha 9 e 10 é identificado nome de usuário e senha que pode acessar o banco de dados *MySQL*. Na linha 11 é identificado o endereço do servidor onde está localizado o banco de dados. Deste modo o *Sisclin* pode estar instalado em uma máquina e acessar o banco de dados em um computador remoto, no caso da *UNIPAC* o

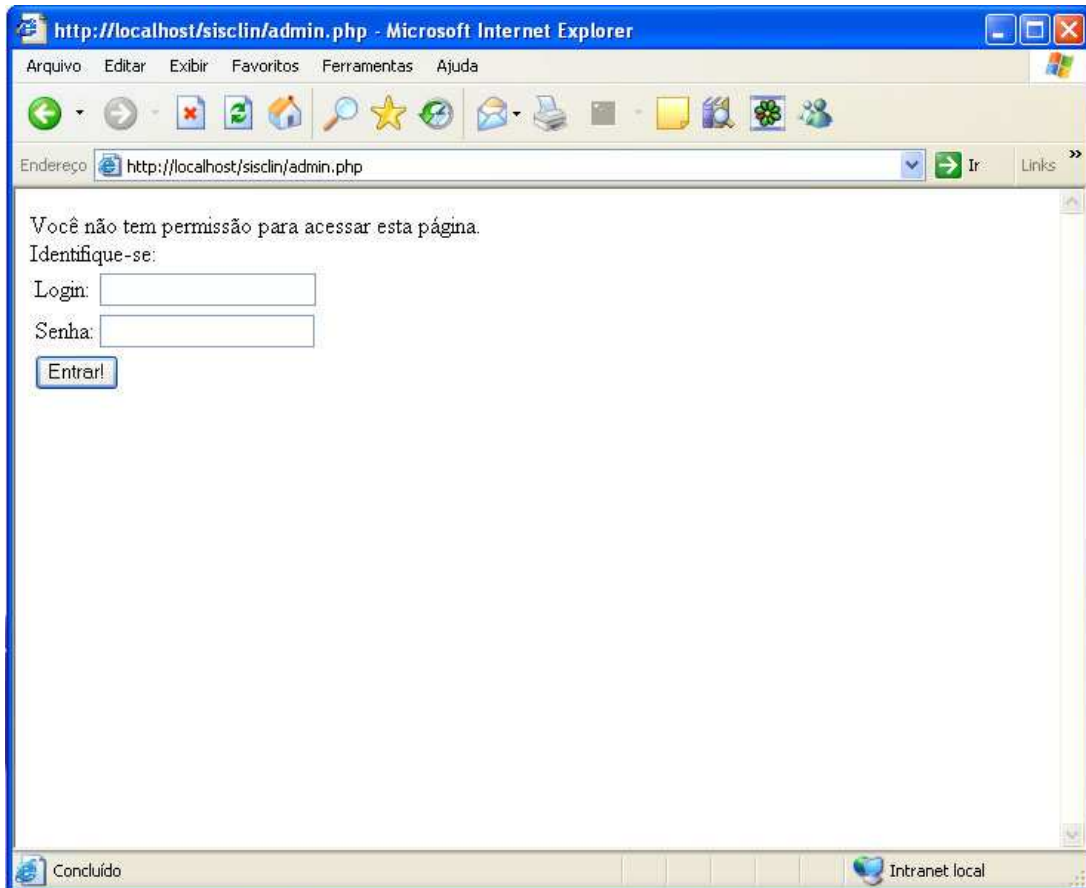
MySQL precisa estar instalado em um servidor apenas e trabalhar com várias clínicas remotas.

No arquivo class.php estão contidas as classes que o sistema utiliza. A conexão com o mysql é feita a partir da classe mysql, a figura 17 mostra o desenvolvimento da classe.

```
57 // Conexão com mysql
58 class mysql {
59     function open_conn(&$conn,$user,$pass,$host,$db){
60         $conn = mysql_connect($host,$user,$pass) or die(mysql_error());
61         $db = mysql_select_db($db);
62     }
63     function close_conn(&$conn){
64         $conn = mysql_close($conn);
65     }
66 }
```

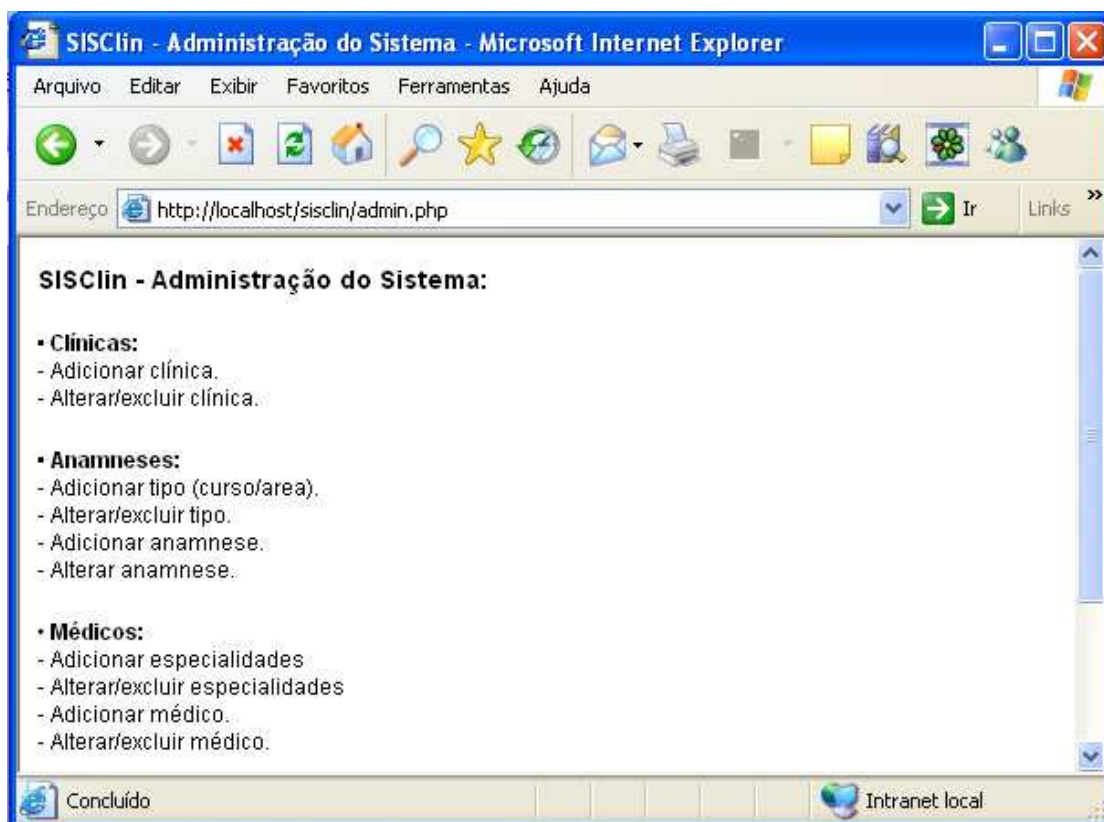
**Figura 17: Classe MySQL**

A linha 59 mostra o método de conexão com o MySQL. A conexão é feita na linha 60, a partir da função mysql\_connect, os valores das variáveis são passados pelo arquivo conf.php e o método retorna a variável \$conn que é utilizada nas consultas sql, conforme será visto mais à frente. O método close\_conn finaliza a conexão com banco de dados.



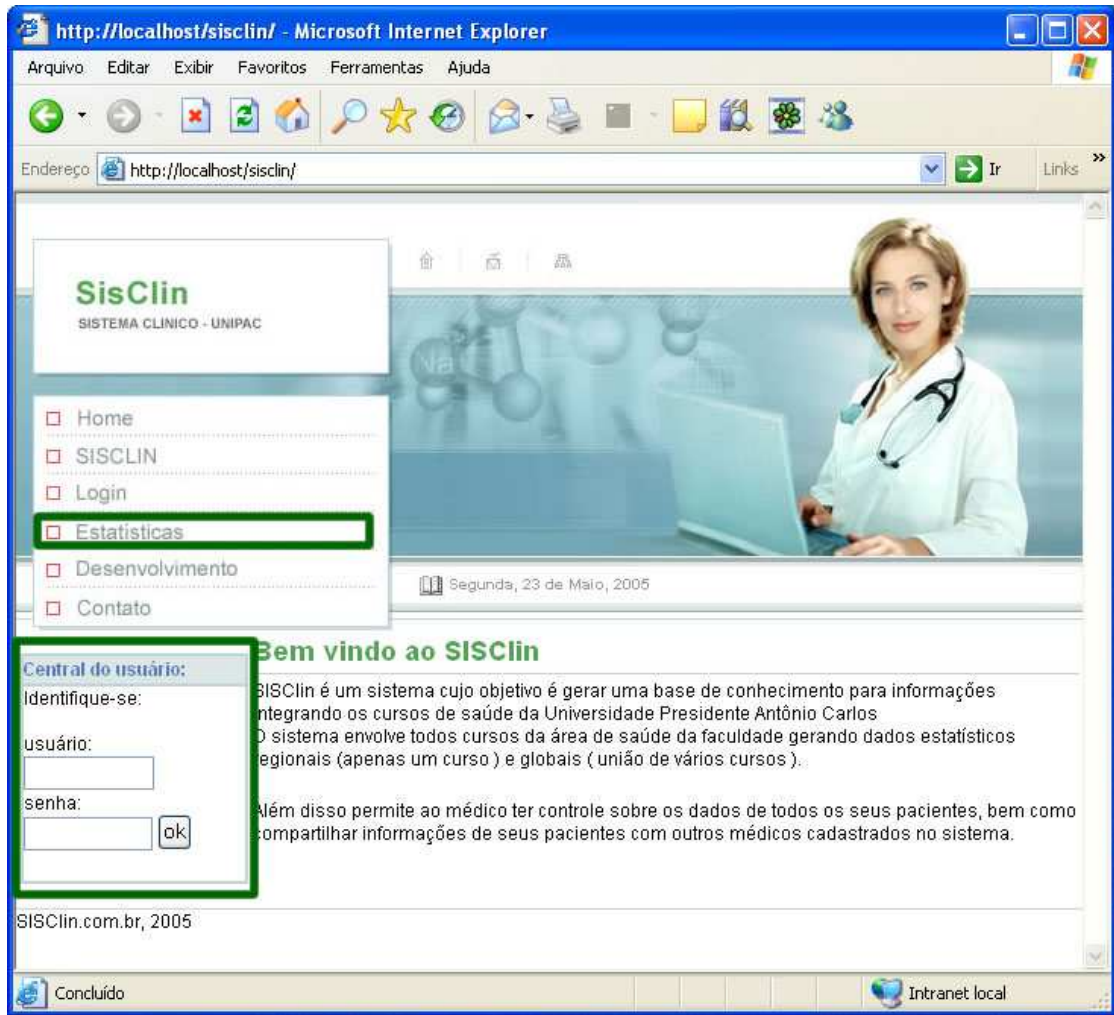
**Figura 18: página de login do administrador**

O administrador do sistema deve acessar o arquivo admin.php para ter acesso às opções de administração como mostra a figura 18. o login é realizado com os dados contidos no arquivo conf.php (figura 16). Após o usuário entrar com seu nome e senha de administrador ele terá acesso ao menu de opções do administrador (figura 19) onde terá opções para adicionar e alterar clínicas, adicionar e alterar anamneses, adicionar e alterar nomes de especialidade e adicionar e alterar dados de médicos.



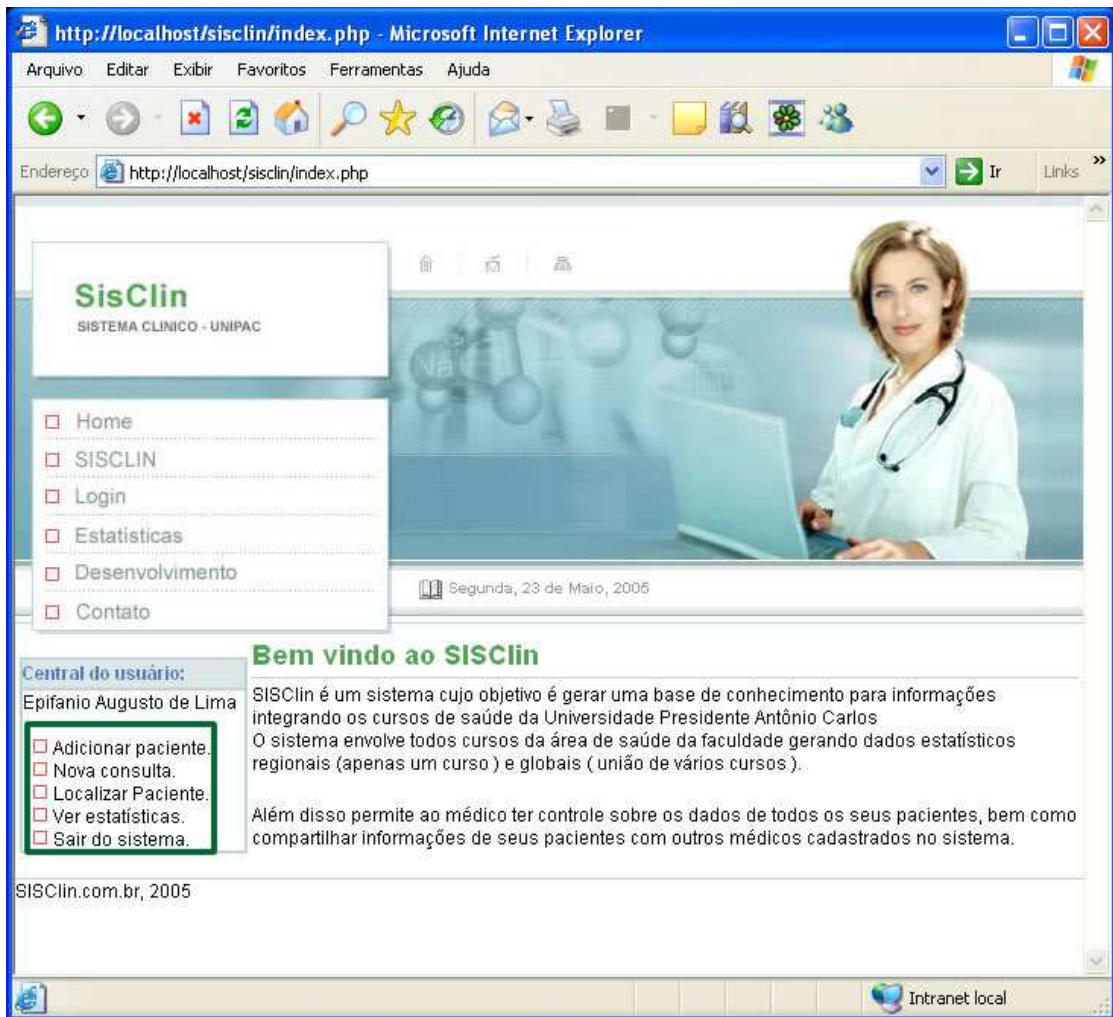
**Figura 19: Tela de administração do sistema**

A tela inicial do sistema mostrada na figura 20 pode ser acessada pelo médico e pelos outros usuários do Sisclin e pode ser acessada também no endereço <http://www.sisclin.com.br>. Na figura estão destacados o botão Estatísticas e o formulário de login do médico. As estatísticas estarão disponíveis a todos os usuários.



**Figura 20: Página inicial do Sisclin**

A página inicial assim como as páginas secundárias possui os botões de acesso *Home*, para voltar à página inicial, *SISCLIN*, uma breve descrição sobre o sistema, *Login*, onde o médico poderá se identificar, *Estatísticas*, onde o usuário terá acesso às estatísticas do sistema, *Desenvolvimento*, uma breve descrição sobre o desenvolvimento do sistema e *contato*, um formulário para o usuário entrar em contato com o administrador através do e-mail.



**Figura 21: Menu do Médico**

A Figura 21 mostra as opções do médico depois de identificado. O médico terá opções para Adicionar um novo paciente, criar uma nova consulta, localizar um paciente para ver seu histórico, acessar as estatísticas e sair do sistema. O sistema de login foi criado a partir do gerenciamento de seções.

```

12  $query = mysql_query("select id,nome,clinica from medico where user='$user' and senha='$pass'", $conn)
13  if (@mysql_result($query,0,"id")== ""){
14      ?>
15      <script language="javascript">
16      alert("Nome ou senha incorreto!");
17      </script>
18      <?
19  }
20  $_SESSION['id'] = @mysql_result($query,0,"id");
21  $_SESSION['nome'] = @mysql_result($query,0,"nome");
22  $_SESSION['clinica'] = @mysql_result($query,0,"clinica");
23  echo "Usuário conectado com sucesso!<br>";

```

**Figura 22: Validação de usuário**



A Figura 22 mostra o trecho de código responsável pela validação do usuário. O usuário deve passar os valores de usuário e senha (Figura 20), a partir de então é realizada uma consulta no banco de dados um usuário com o nome e a senha passados. Se os valores retornarem um médico o valor da seção para id, nome e clinica são alterados (linhas 20, 21,22). Uma outra função checa se \$\_SESSION['ID'] foi inicializado e possui um valor diferente de nulo, se possuir significa que o usuário se conectou com sucesso.

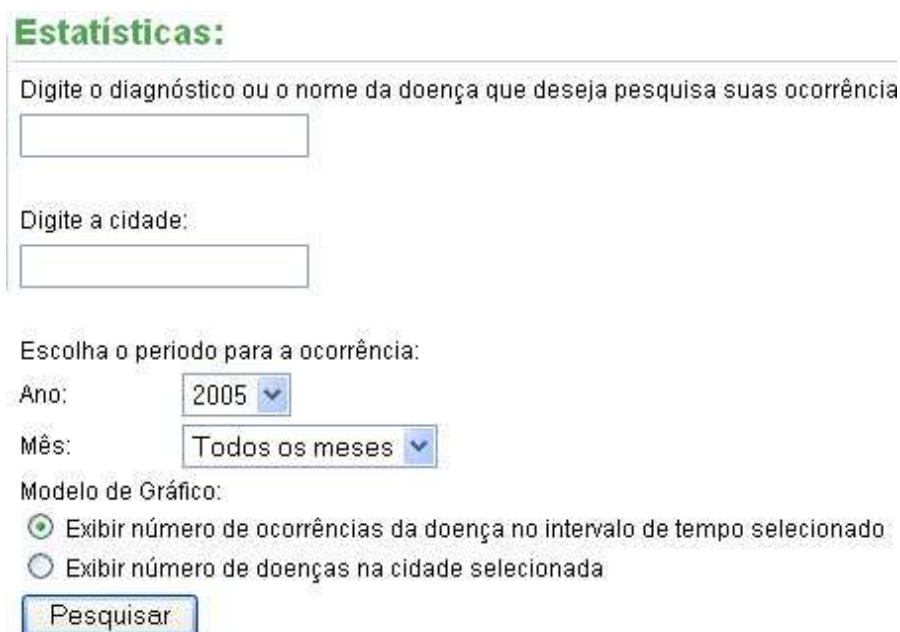
The screenshot displays the 'Adicionar paciente' (Add patient) form in the SisClin system. The browser window title is 'http://localhost/sisclin/index.php?op=add\_paciente - Microsoft Internet Explorer'. The page features a header with the 'SisClin SISTEMA CLINICO - UNIPAC' logo and a navigation menu. The main content area is titled 'Adicionar paciente:' and includes a 'Central do usuário:' sidebar for 'Epifanio Augusto de Lima'. The form is organized into sections: 'Dados do Paciente:' (Patient Data) with fields for Name, Age, Birth date, and Sex; 'Filiação:' (Affiliation) with fields for Father and Mother details; and 'Endereço:' (Address) with fields for Street, Number/Complement, Neighborhood, City, State, and CEP. The browser's status bar at the bottom indicates 'Intranet local'.

**Figura 23: Tela de adicionando paciente**

A figura 23 apresenta o formulário de cadastro de um novo paciente. Os campos nome, idade, nascimento e sexo são obrigatórios. Os demais campos são opcionais.

O sistema de estatísticas possui dois tipos de gráficos, o número de ocorrências de determinada doença em uma cidade e o número de doenças que ocorreram em uma cidade pesquisada.

A figura 24 apresenta o formulário a ser preenchido para realizar uma pesquisa.



**Estatísticas:**

Digite o diagnóstico ou o nome da doença que deseja pesquisa suas ocorrência

Digite a cidade:

Escolha o período para a ocorrência:

Ano:

Mês:

Modelo de Gráfico:

Exibir número de ocorrências da doença no intervalo de tempo selecionado

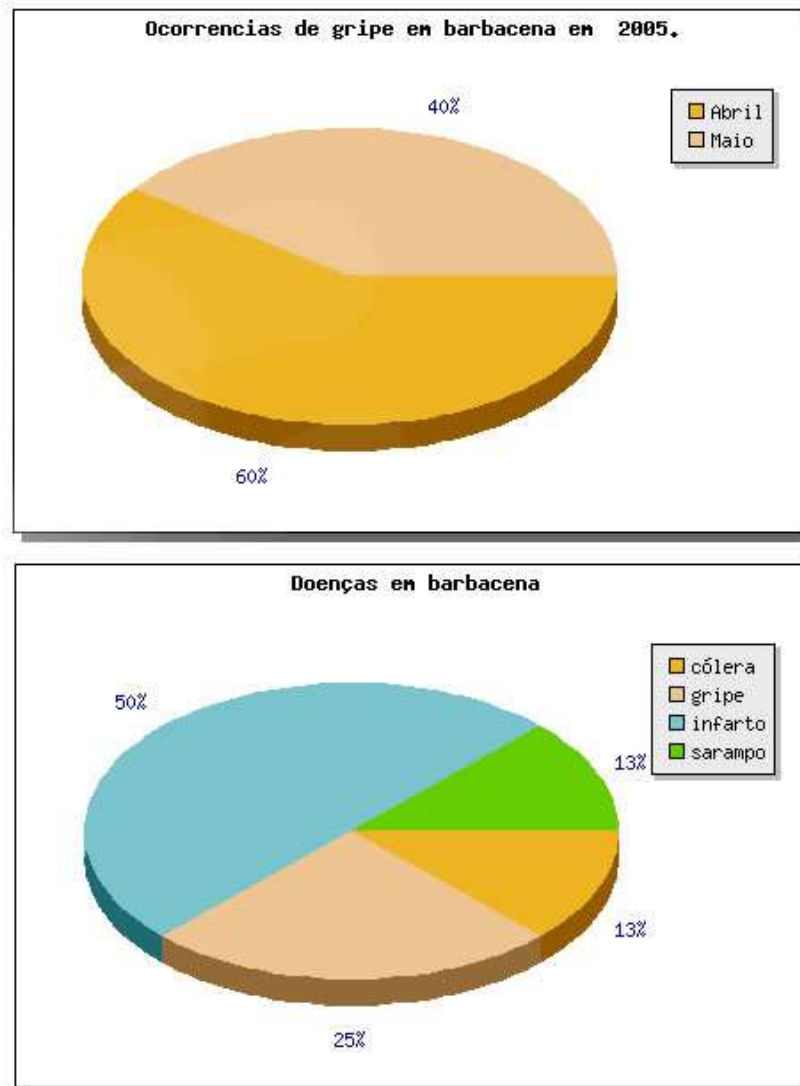
Exibir número de doenças na cidade selecionada

**Figura 24: Formulário de pesquisa**

Para realizar uma nova pesquisa o usuário deve preencher o nome da doença a ser pesquisada, o nome da cidade onde houve a doença, o ano. A escolha no campo mês é opcional. Se o usuário escolher “Todos os meses” será exibido o gráfico com os meses em que a doença ocorreu, se o usuário escolher um mês específico será exibido os dias do mês em que houve a ocorrência da doença. O Sisclin pode criar um gráfico com o número de ocorrências da doença em um intervalo de tempo selecionado ou o número de doenças na cidade selecionada. A escolha é feita a partir do campo “Modelo de Gráfico”.

A figura 25 exemplifica quais os possíveis tipos de gráficos. O primeiro gráfico mostra o número de ocorrências de gripe em Barbacena em 2005, o segundo gráfico mostra as ocorrências de doenças em Barbacena no ano de 2005.





**Figura 25: Exemplo de gráficos gerados pelo Sisclin**

Os gráficos são gerados a partir da biblioteca gratuita JpGraph .especializada na geração de gráficos. O JpGraph é uma biblioteca orientada a objetos totalmente escrita em PHP e pode ser utilizada em qualquer script PHP. Esta biblioteca possui uma interface simples e amigável na geração de gráficos simples e complexos [13].

Um exemplo do código para gerar um gráfico de pizza pode ser visto abaixo:

```
<?
```

```
// incluir as classes jpgraph usadas para criação do gráfico
```

```
include ("jpgraph.php");
```

```
include ("jpgraph_pie.php");
```

```
include ("jpgraph_pie3d.php");
```

```
// definir um array com os valores de cada fatia (Número de doenças)
```

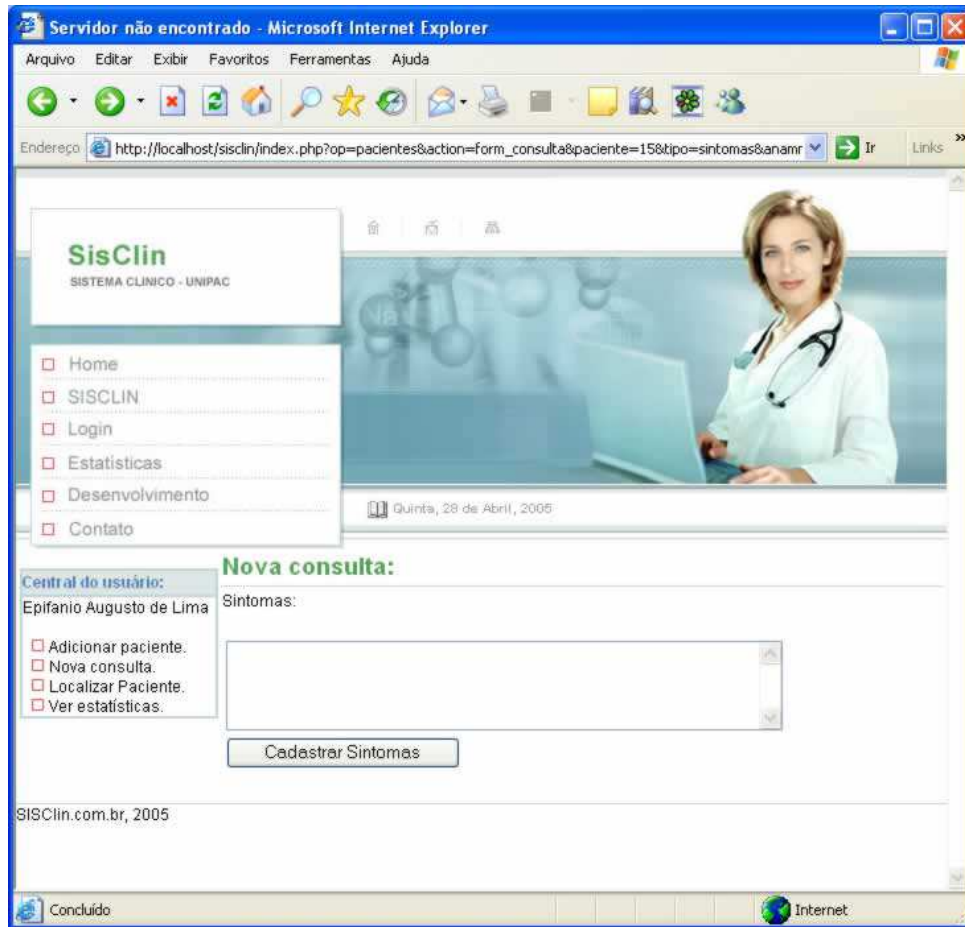
```

$numero_doencas = array(8, 26, 5, 13);
// definir um array que será usado como legenda para cada
// valor correspondente na lista acima (legenda = nomes)
$nome_doencas = array ("Cólica", "Gripe", "Infarto", "Sarampo");
// criar novo gráfico de 350x200 pixels com tipo de imagem automático
$grafico = new PieGraph(350,200,"auto");
// adicionar sombra
$grafico->SetShadow();
// título do gráfico
$grafico->title->Set("Doenças em Barbacena");
$grafico->title->SetFont(FF_FONT1,FS_BOLD);
// definir valores ao gráfico
$p1 = new PiePlot3D($numero_doencas);
// destacar o valor correspondente ao elemento (1), sendo
// que o primeiro elemento do array corresponde a (0),
// então esse é o cara cujo valor equivale a 26
$p1->ExplodeSlice(1);
// centralizar a 45% da largura
$p1->SetCenter(0.45);
// definir legendas
$p1->SetLegends($nome_doencas);
// adicionar valores ao gráfico
$grafico->Add($p1);
// gerar o gráfico
$grafico->Stroke();
?>

```

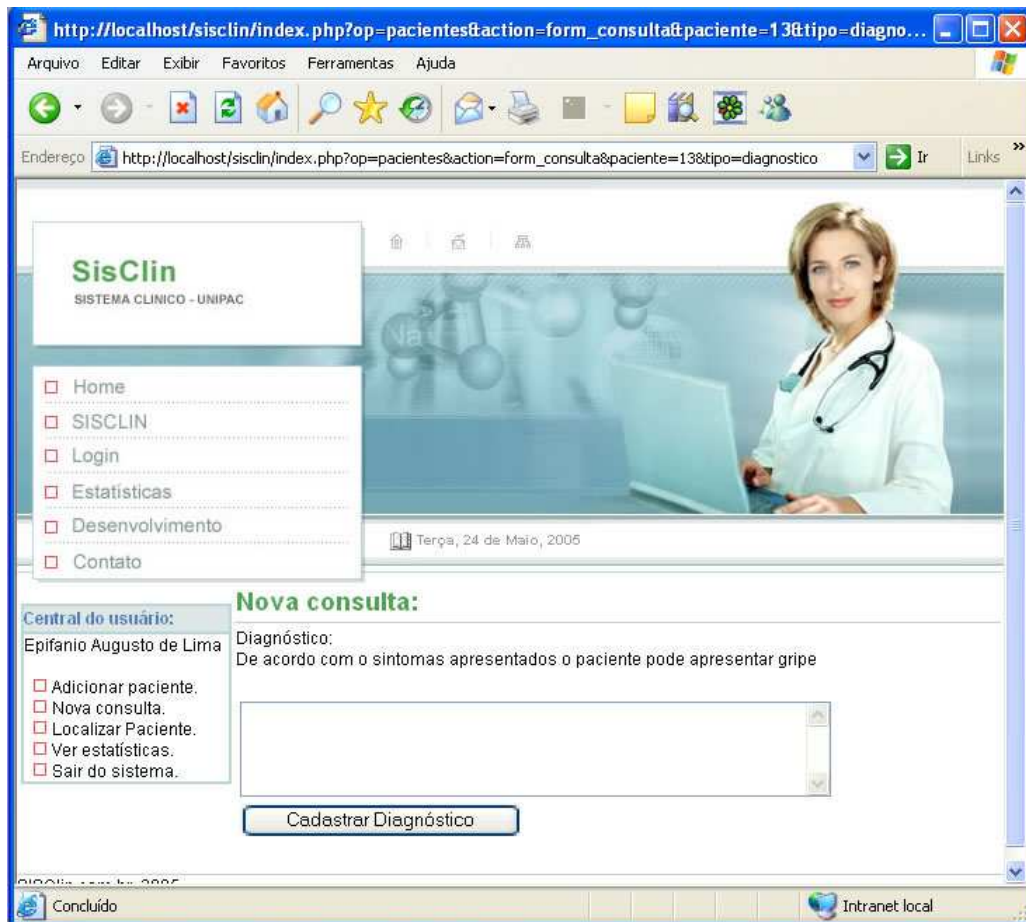
Uma nova consulta pode ser realizada clicando no link “nova consulta” (figura 21) a partir de então o usuário deve digitar o nome do paciente para pesquisar se o mesmo já está cadastrado e logo depois escolher a anamnese que será utilizada na consulta.

Depois de preenchida a anamnese o médico deverá cadastrar os sintomas do paciente (figura 26). O indicado é que o médico cadastre um sintoma por linha. A figura 26 mostra a tela onde o médico cadastra a consulta.



**Figura 26: Cadastro de sintomas**

De acordo com os sintomas especificados o sistema especialista irá indicar ao médico quais as possíveis doenças que o paciente pode ter. O sistema especialista tem como principal objetivo auxiliar o médico na escolha do diagnóstico, em nenhum momento o sistema especialista deve ser substituído pelo mesmo. A figura 27 ilustra o momento em que o sistema especialista auxilia o médico na escolha do diagnóstico.



**Figura 27: Tela de diagnóstico**

Depois de cadastrar o Diagnóstico o médico deve cadastrar a receita médica e a consulta é finalizada. O histórico do paciente pode ser visto clicando no link “Localizar Paciente” onde é exibido um campo para a pesquisa de pacientes. Ao localizar o paciente o Sisclin exibe todas as consultas realizadas em todas as clinicas por determinado paciente com a anamnese utilizada, os sintomas, os diagnósticos e as receitas cadastradas nas outras consultas.

## 6 – CONSIDERAÇÕES FINAIS

O Sisclin mostrou-se bastante útil no gerenciamento de consultas em clínicas médicas. O objetivo de gerar uma base de dados integrando várias clínicas médicas foi alcançado. O levantamento de dados estatísticos através das estatísticas se mostrou amigável e eficiente e no futuro poderá ser utilizado com o *data mining* na localização de focos de doenças e na prevenção de doenças em determinadas regiões.

O Sistema Especialista se mostra útil no auxílio ao médico em identificar doenças raras na região e que podem ter sintomas parecidos com doenças mais freqüentes. O Sistema Especialista pode reduzir significativamente o número de erros médicos.

O conhecimento adquirido permitiu a criação de um sistema onde, até o momento não foram encontradas falhas.

Este trabalho contribui para a UNIPAC no desenvolvimento de profissionais da área de saúde buscando uma forma de padronizar as consultas e criando um gerenciamento mais simples para administradores e professores acompanharem as consultas realizadas e estatísticas do sistema.

### 6.1 – TRABALHOS FUTUROS

Como sugestão para trabalhos futuros fica a sugestão para a criação de um modelo *data mining* buscando novas linhas de pesquisa para encontrar possíveis focos de doenças.

Também pode ser implementado um sistema *wap*, onde o usuário possa acessar dados referentes ao Sisclin a partir de sistemas móveis e o paciente possa receber sua receita médica em seu celular.

## 7 – REFERÊNCIAS BIBLIOGRÁFICAS

- [1] CONVERSE, Tim, PARK, Joyce, **PHP: A Bíblia**, Campus, Rio de Janeiro, 2003.
- [2] Autor Anônimo, **Segurança Máxima**, Campus, Rio de Janeiro, 2001.
- [3] SCAMBRAY, Joel, MCCLURE, Stuart, KURTZ, George, **Hackers Expostos**, 2ª Ed., Makron Books, São Paulo, 2001.
- [4] DIMOV, Jordan, **Segurança em programação PHP**, disponível em <http://br.linuxchix.org/node.php?id=149> acesso em maio de 2005.
- [5] RABELLO, Roberto dos Santos, **O que são Sistemas Especialistas e Engenheiros de Conhecimento**, Disponível em [http://www.universia.com.br/html/materia/materia\\_gdba.html](http://www.universia.com.br/html/materia/materia_gdba.html) acesso em maio de 2005
- [6] CUER, Andréia Oliveira, **SISTEMAS ESPECIALISTAS APLICADOS À MEDICINA**, Disponível em <http://www.din.uem.br/~ia/medicina/index.html> acesso em maio de 2005.
- [7] ZAMBALDE, André Luiz, **HIPERTEXTO, HIPERMÍDIA E WWW**, Disponível em <http://www.comp.ufla.br/~zambalde/cap4ihm.htm> acesso em maio de 2005.
- [8] MYSQL AB, **Um Guia para um TCO de banco de dados menor**, Disponível em [http://www.mysqlbrasil.com.br/uploads/244/mysqlbr\\_tco.pdf](http://www.mysqlbrasil.com.br/uploads/244/mysqlbr_tco.pdf) acesso em maio de 2005.
- [9] Autor Anônimo, **Modelagem**, Disponível em <http://pt.wikipedia.org/wiki/Modelagem> acesso em abril de 2005

[10] BOOCH, Grady, RUMBAUGH, James, JACOBSON, Ivar, **UML, Guia do Usuário**, Campus, Rio de Janeiro, 2000.

[11] PRESSMAN, Roger S., **Engenharia de Software**. 5ª Ed., Makron Books, São Paulo, 2001.

[12] PEREIRA, Fernando Diego Campos, **SISTEMAS HIPERMÍDIA: hipermídia sobre cidades históricas**. Barbacena: s.n, 2004. 94 p. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) – Universidade Presidente Antonio Carlos. Faculdade de Ciência da Computação e Comunicação Social

[13] JPGRAPH, **What is JpGraph?**, disponível em <http://www.aditus.nu/jpgraph/> acesso em abril de 2005

## Anexo 1 – Configurando o Servidor Apache

O desenvolvimento do SISClin contou utilizou como softwares principais o servidor Apache, o banco de dados MySQL e a linguagem de programação PHP. Este anexo visa auxiliar a instalação e a configuração destes softwares no sistema operacional Windows.

Primeiro é necessário o download dos programas a serem usados.

Quanto ao Apache, usaremos o Apache 1.3.\*, visto que na documentação do PHP existe a seguinte recomendação: "Não use Apache 2.0 e PHP em um sistema de produção, seja no Unix ou no Windows".

- Apache:

[http://mirrors.uol.com.br/pub/apache/httpd/binaries/win32/apache\\_1.3.31-win32-x86-no\\_src.exe](http://mirrors.uol.com.br/pub/apache/httpd/binaries/win32/apache_1.3.31-win32-x86-no_src.exe)

- MySQL 4.1.8: <http://dev.mysql.com/get/Downloads/MySQL-4.1/mysql-4.1.8-win-noinstall.zip> /from/<http://www.linorg.usp.br/mysql/>

- PHP 5.0.3: <http://br2.php.net/get/php-5.0.3-Win32.zip> /from/this/mirror

Depois de feito o download execute a instalação do Apache e o instale com as configurações padrões. Se quiser, pode escolher outro diretório para a instalação. Extraia o MySQL em uma pasta qualquer. Recomendo dentro da pasta onde você instalou o Apache. Ex: C:\Arquivos de Programas\Apache Group\Apache\mysql.

Extraia o PHP 5 na pasta C:\php5

O próximo passo é a configuração do PHP, vá para a pasta c:\php5 e copie o arquivo php5ts.dll para a seguinte pasta, de acordo com o seu Windows:

- c:\windows\system (em Windows 9x/Me)
- c:\windows\system32 (em WindowsXP)
- c:\winnt\system32 (para Windows NT/2000)

Copie também o arquivo c:\php5\libmysql.dll para umas das pastas ditas acima, de acordo com o Windows em uso. Esse arquivo é necessário para o funcionamento do MySQL no PHP.



Ainda no c:\php5, renomeie o arquivo "php.ini-dist" para "php.ini" e abra-o. Procure a linha `extension_dir = "./"` e a altere para `extension_dir = "c:/php5/ext/"`, é o diretório onde ficam as extensões do php (MySQL, Curl, GD, etc). Agora, localize a linha `;extension=php_mysql.dll` e tire o `;` do início dela. Se quiser também, faça o mesmo na linha `;extension=php_gd2.dll`, caso queira a biblioteca GD para a manipulação de imagens. Salve as alterações e mova o "php.ini" para a pasta:

- c:\windows (em Windows 9x/Me/XP)
- c:\winnt (para Windows NT/2000)

Vá para a pasta onde você instalou o Apache e abra o arquivo `conf/httpd.conf` em qualquer editor de texto. (Ex: Bloco de Notas).

1º) Localize a linha `#LoadModule unique_id_module modules/mod_unique_id.so` e logo abaixo dela adicione: `LoadModule php5_module "c:/php5/php5apache.dll"`

2º) Localize a linha `AddModule mod_setenvif.c` e logo abaixo, adicione: `AddModule mod_php5.c`

3º) Localize `AddType application/x-tar .tgz` e logo abaixo, adicione:

`AddType application/x-httpd-php .php`

`AddType application/x-httpd-php-source .phps`

4º) Localize:

`<IfModule mod_dir.c>`

`DirectoryIndex index.html`

`</IfModule>`

E logo ao lado do `index.html` adicione:

`index.php default.php main.php`

Não há nada para se configurar no MySQL, você só deve iniciar o mesmo. Vá para a pasta que instalou o MySQL, então abra o `bin/mysqld.exe`, iniciando o servidor do MySQL.

Lembre-se que sempre você terá de iniciar o MySQL. Se não quiser ir na pasta toda vez que iniciar o computador, crie um atalho para o `bin/mysqld.exe` e coloque no "Iniciar, Programas, Inicializar (ou Iniciar, de acordo com o windows)".

Agora, vá em "Iniciar, Programas, Apache HTTP Server, Control Apache Server, Restart", para reinicializar o Apache com as alterações feitas. Agora estão configurados PHP5, MySQL a Apache em seu Windows!.

Para efetuar um teste, crie um arquivo chamado phpinfo.php, com o conteúdo:

`<? phpinfo(); ?>` e o coloque na pasta htdocs dentro da pasta do Apache. Abra seu navegador e digite `http://localhost/phpinfo.php`. Se a página abrir com as informações do PHP, significa que tudo deu certo.

## Anexo 2 – Instalando e Configurando o phpMyAdmin

phpMyAdmin é pretendido para administração do MySQL sobre a web. Com o phpMyAdmin instalado em seu servidor você poderá criar banco de dados, copiar tabelas, criar, apagar e editar. O phpmyadmin é de fácil configuração e administração.

Para utilizar o phpMyAdmin é necessário:

MySQL: [www.mysql.com](http://www.mysql.com)

Apache: [www.apache.org](http://www.apache.org)

PHP: [www.php.net](http://www.php.net)

Para fazer o download do phpMyAdmin entre na url [www.phpwizard.net](http://www.phpwizard.net) e entre na seção phpMyAdmin, você poderá optar por duas opções do phpMyAdmin:

- Extensão \*.php3

- Extensão \*.php

É aconselhável a versão do phpMyAdmin de extensão \*.php, visto que a maioria dos servidores reconhece essa extensão.

Após fazer o download descompacte o mesmo e copie todos os arquivos para uma pasta no seu servidor ex: mysql e edite o arquivo config.inc.

```
$cfgServers[1]['host'] = 'localhost'; // MySQL hostname
```

```
$cfgServers[1]['port'] = ''; // MySQL port - leave blank for default port
```

```
$cfgServers[1]['adv_auth'] = true; // Use advanced authentication?
```

Você deve deixar true, pois assim ao acessar a url no browser ele aparecerá uma caixa de login e senha do usuário, para sua melhor segurança. Caso deixe false ele não aparecerá o login.

```
$cfgServers[1]['stduser'] = 'usuário'; // MySQL standard user (only needed with advanced auth)
```

```
$cfgServers[1]['stdpass'] = 'senha'; // MySQL standard password (only needed with advanced auth)
```

```
$cfgServers[1]['user'] = 'usuário'; // MySQL user (only needed with basic auth)
nome do usuário
```

```
$cfgServers[1]['password'] = 'senha'; // MySQL user (only needed with basic auth)
nome do usuário
```

Pronto, a configuração básica do phpMyAdmin já esta pronta.

Agora é só acessar via browser seu phpMyAdmin ex:  
[www.seusite.com.br/mysql/](http://www.seusite.com.br/mysql/)

### **Anexo 3 – Sisclin**

Sistema em CD-ROM anexo.