

Universidade Federal de Campina Grande – UFCG
Departamento de Sistemas e Computação – DSC
Centro de Ciências e Tecnologia – CCT
Disciplina : Estágio Supervisionado
Aluno : Sérgio Augusto Freitas Pinto
Orientador : Marcus Salerno de Aquino

Relatório Final

Campina Grande 08/04/2003



Biblioteca Setorial do CDSA. Maio de 2021.

Sumé - PB

Índice

1. Apresentação	3
2. Objetivos do Estágio	4
3. Ambiente de Estágio	4
4. Descrição do Problema	5
5. Solução Adotada	5
6. Atividades Desenvolvidas	7
7. Conclusões	10
8. Anexos	11

1. Apresentação

A disciplina Estágio Supervisionado surge como uma nova oportunidade de familiarizar o aluno em formação com a atual conjuntura do mercado de informática, pois com a experiência de desenvolver um projeto em toda sua magnitude fica, por responsabilidade do estagiário, trazendo assim uma maior visibilidade a dinâmica do cotidiano no mercado. Foi a função do orientador com um acompanhamento minucioso sobre o problema e o auxílio a uma solução do mesmo.

O sistema desenvolvido é chamado de ColetivaWeb, tem como principal funcionalidade, viabilizar uma entrevista coletiva via internet . Essa ferramenta proporciona a interatividade entre diversos jornalistas e um entrevistado, de maneira e eficiente, bastando somente de um conhecimento básico de navegação na internet.

O ColetivaWeb também trará uma redução significativa com os gastos gerados numa entrevista. Numa entrevista existe despesas, como deslocamento de jornalista, hospedagem, credenciamento, etc. Com a utilização da aplicação a diminuição desses gastos apareceram nitidamente.

O relatório está constituído em duas partes. Inicialmente com uma narrativa do problema a ser resolvido e sua solução, bem como as técnicas utilizadas para desenvolvê-lo, e em seguida apresetam-se os anexos, cujo primeiro, é a tela de autenticação do usuário , em seguida a tela de cadastro e o ambiente onde acontece a entrevista. O restante dos anexos é a documentação do sistema com os diagramas UML, a arquitetura onde o sistema foi desenvolvido, a descrição das ferramentas e o plano de estágio.

2. Objetivos do Estágio

O objetivo deste estágio partiu da necessidade da empresa Pacto4 entrar no mundo de sistemas distribuídos na Web. Atualmente esse mercado é o mais promissor no desenvolvimento de softwares para grandes e médias corporações, pois trás para o cliente a oportunidade de atender todo e qualquer possível usuário do sistema, apenas com um conhecimento básico em internet.

3. Ambiente de Estágio

O estágio foi realizado junto a Pacto4, empresa incubada ao Projeto Poligene – Softex, localizado na Universidade Federal de Campina Grande (UFCG) no 1º andar do Departamento de Sistemas e Computação.

O projeto Poligene – Softex é uma programa que proporciona a empreendedores, subsídios para a estruturação de sua empresa a ele vinculada. Como consultoria em marketing, instalações físicas de qualidade, equipamentos, cursos etc.

Atualmente a Pacto4, inicia um processo de desenvolvimento de aplicações para Web. Utilizando tecnologia baseada em Java, adentra num nicho de mercado absolutamente rentável e promissor.

O ambiente em que foi desenvolvido o estágio dispõe do da seguinte estrutura :

- Um servidor IBM Netfinity
- Sete máquinas Compaq de 300Mhz , com HD 10Gb todas ligadas em Rede
- Uma impressora Laser
- Linha Telefônica
- Ar Condicionado

A estrutura montada para o desenvolvimento do projeto foi de certa forma satisfatória, as máquinas apresentaram um rendimento sustentável para cumprir o cronograma estimado para o projeto, os horários de trabalho eram flexíveis para que o estagiário não se ausentasse de suas outras tarefas acadêmicas, foi importantíssimo o acesso a livros, já que a biblioteca se encontrava no mesmo prédio e andar. Em contrapartida, o espaço físico da sala, bem como a falta de manutenção na impressora e no ar condicionado, gerou alguns transtornos ao andamento do projeto.

4. Descrição do Problema

A empresa Deléon Comunicações realiza entrevista coletiva com convidados em sua sede na grande São Paulo, porém alguns de seus jornalistas credenciados não podiam sempre comparecer a entrevista, inviabilizando assim um nível maior de interação entre entrevistado e entrevistadores

Este problema era o foco principal do desenvolvimento do projeto, porém durante a construção da ferramenta a empresa contratante identificou a necessidade de documentar uma entrevista e este novo requisito foi também incluso dentro do cronograma, como a segunda etapa a ser resolvida pela ferramenta.

Diante destes problemas foi desenvolvido um sistema chamado ColetivaWeb que viabilizou o desenrolar de uma entrevista através da internet.

5. Solução Adotada

O ColetivaWeb foi construído sobre uma arquitetura em três camadas (Anexo 1). A escolha dessa estrutura para desenvolvimento partiu do fato de que, em se tratando de um sistema Web esta é a estrutura que demonstrou sucesso em outros projetos realizados como Projeto Cenisa, portal de e-commerce agropecuários, o Atualizare, sistema gestor de sites dinâmicos.

A estrutura de três camadas proporciona uma escalabilidade maior ao sistema, suportando a utilização do mesmo por centenas de usuários e a diminuição dos custos de instalação e manutenção do sistema, pois para a sua utilização basta apenas o uso de um browser (Internet Explorer ou Netscape).

A tecnologia escolhida para o desenvolvimento do sistema foi centrado em Java, pois foi amplamente estudada e difundida durante todo o curso em diversas disciplinas. Além dessa facilidade a tecnologia trouxe a desvinculação do problema de plataforma, ou seja, o sistema funcionará tanto para Windows como para Linux.

A análise do sistema será baseada em UML (Anexo 2), linguagem de documentação já conhecida e aplicada sistematicamente na universidade e a mais utilizada para a arquitetura em que se apresenta o projeto. A UML proporcionou a descrição de cada modelo, especificação de suas classes e construção dos diagramas, que realmente foram necessários a melhor compreensão da ferramenta. Entretanto a não utilização de todos os diagramas de UML, como diagrama de colaboração, diagrama de gráficos de estados, diagrama de componentes, etc. Deveu-se ao tamanho do projeto e a redundância das informações que traria na confecção de todos esses os documentos e diagramas.

O sistema proporciona um cadastro de usuários, basicamente repórteres e entrevistados, onde os mesmos podem se cadastrar dispõem de um Login e Senha.

Uma vez já cadastrado no sistema, automaticamente o usuário é inserido em uma "Sala" (Anexo 3), estando, conseqüentemente habilitado a participar da entrevista. Entende-se por "Sala", o ambiente ou tela onde se processa a entrevista.

Dentro da "Sala" há um local onde os repórteres elaboram suas perguntas num campo de texto em destaque, o qual o entrevistado e todos os demais participantes da entrevista têm acesso visualmente. As perguntas elaboradas ficam visíveis para todos os usuários, contudo o entrevistado escolhe qual a pergunta a qual ele quer responder, podendo "descartar" a que ele julgar desinteressante. O usuário entrevistado responde, ainda na mesma "Sala" às perguntas e estas respostas também ficam visíveis a todos os usuários ativos na "Sala". Cada usuário dentro da "Sala" sabe quem são os participantes da entrevista, pois existe uma lista ativa de todos os usuários que estão participando da entrevista.

Uma entrevista encerrará quando o Entrevistado se ausentar da "Sala".

6. Atividades Desenvolvidas

	1º semana	2º semana	3º semana	4º semana
Outubro	Estudo do problema e definição da tecnologia a ser utilizada	Modelagem do sistema e diagramação UML	Desenvolvimento do Banco de Dados e reuniões com o orientador	Modularização do sistema e reuniões com o cliente
Novembro	Implementação do 1º módulo (Cadastro de Usuário, inserção e alteração)	Teste de aceitação do Módulo desenvolvido	Implementação do 2º módulo (Autenticação)	Testes de aceitação do Módulo Autenticação
Dezembro	Implementação Modulo de Sincronização de Perguntas - Respostas	Implementação Modulo de Sincronização de Perguntas - Respostas	Implementação Modulo de Sincronização de Perguntas - Respostas e Reunião com Orientador	***** *****
Janeiro	Implementação Modulo de Sincronização de Perguntas - Respostas	Testes de aceitação do Módulo Perguntas e Respostas e Reunião com Orientador	Implementação do Módulo Lista Ativa	Implementação do Módulo Lista Ativa
Fevereiro	Testes de Aceitação do Modulo Lista Ativa e reunião com o Orientador	Módulo Imprimir Entrevista	Testes Funcionais	***** *****
Março	*****	Relatório	Relatório	Relatório

obs : os campos marcados com * são feriados. Cada semana são 20 horas dedicadas ao projeto, divididas entre estudos sobre as novas tecnologias e implementação.

6.1 Modelagem do sistema e diagramação UML

Como dito anteriormente o sistema foi baseado em UML, entretanto foi confeccionado o diagrama de Use Case (Anexo 4), que representa a quem pertence cada ação do sistema.

O diagrama de classes (Anexo 5), retratando os relacionamentos entre as classes Java, e por último o diagrama de sequência (Anexo 6), refletindo o caminhamento de uma determinada ação dentro do sistema.

6.2 Modulo de Sincronização de Perguntas e Respostas

Este módulo é onde está toda a regra de negócio do sistema, ele é o gestor da funcionalidade essencial do projeto, que é o processo de uma entrevista.

Como primeiro desafio do projeto, fez-se necessário um estudo de Thread Java. Este pacote foi fundamental para a finalização do módulo pois existia um problema de concorrência entre os processos. O problema se iniciava no momento em que vários usuários (repórteres) tentavam enviar perguntas, enquanto o entrevistado estava respondendo a uma outra pergunta e de um outro usuário. Nesse caso o consumo de memória virtual da máquina aumentava vertiginosamente tornando assim o sistema lento.

Após o estudo do pacote Thread Java percebi que já se encontrava implementado na biblioteca, métodos que sincronizavam varias requisições a um mesmo processo. Com o estudo do pacote e o entendimento do funcionamento dos métodos, ficou mais claro a solução do módulo.

Contudo, o tempo decorrido para a implementação deste módulo foi estendido várias vezes dentro do cronograma. Fez-se necessário também um estudo adicional sobre o funcionamento de threads na visão de sistemas operacionais, para entender melhor a concorrência entre os processos.

A integração das páginas JSP e dos Beans Java, demandou um tempo considerável para o andamento do módulo, tendo em vista a falta de um Webdesigner para a definição e construção das telas do sistema.

6.3 Testes de Aceitação

Durante os Testes de Aceitação do Módulo Sincronização de Perguntas e Respostas, foi desenvolvido uma classe simuladora para avaliar a quantidade aceitável de participantes em uma entrevista, verificando dessa forma, a carga de dados em circulação suportada pelo servidor.

A classe utilizada, ativava a métodos que criavam as perguntas e respostas no escopo de um laço. Em seguida a quantidade de repetições do laço era verificada e comparada com a quantidade de recursos que aquele processo estava tomando num determinado intervalo de tempo. Através da aferição desse número, chegou-se a conclusão de que os participantes de uma entrevista necessitariam ser limitados, pois poderia tornar o processo de responder as perguntas bastante lento.

Os Testes Funcionais foram realizados quando cada módulo estava finalizado, com páginas JSP's e Bean Java integrados e compilados. Testava-se a funcionalidade com entradas incorretas, tempo de ociosidade de uma ação dentro de uma sessão, problemas de interface quando se carregava o JSP em diferentes browsers.

Esses testes foram importantes para aferir a operacionalidade e a robustez do Módulo de Sincronização, pois comprovaram que a arquitetura e a tecnologia utilizada foi uma escolha certa

6.4 Módulo Lista Ativa

O Módulo Lista Ativa é utilizado para deixar exposto os usuários que estão participando da entrevista. Seu funcionamento é automático e acionado no momento em que um usuário entra numa "Sala".

Estando esse usuário ativo, todos os participantes saberão de quem se trata , pois essa informação será colocada numa lista com o login do respectivo usuário e o veículo de comunicação a quem ele está vinculado.

Finalizando o Módulo Lista Ativa, foi implementado a funcionalidade que encerra uma entrevista, após o usuário entrevistado se ausentar da "Sala".

6.5 Módulo Imprimir Entrevista

O desenvolvimento deste módulo foi concluído através da biblioteca java iText, a qual proporcionou a montagem do documento que acabara de ser produzido ao final de uma entrevista. O documento fica exposto na Internet no formato pdf e txt para download de um usuário logado no sistema.

7. Conclusões

O projeto foi desenvolvido sem maiores problemas com relação ao conhecimento técnico utilizado no desenvolvimento do projeto. A pratica de engenharia de software já realizada em disciplinas do curso supriram de forma satisfatória o projeto como um todo. entretanto a tecnologia JSP foi um desafio, já que ela não foi abordada de nenhuma disciplina da estrutura curricular disponível. Desta forma surgere-se a inclusão de uma disciplina na grade curricular do Curso de computação para a abordagem de sistemas desenvolvidos para a Web.

Atualmente, o sistema se encontra com novas funcionalidades, proporcionando além das descritas neste relatório, opções para acesso a diversas "Salas" de entrevistas, realização de conferências via internet, foi resolvida a limitação usuários utilizando a ferramenta ao mesmo tempo, a interface tornou-se mais amigável e intuitiva, existe hoje também um cadastro de entrevistas realizadas, agendamento de entrevistas dentre outras operacionalidades já existentes. o novo

sistema pode ser acessado no endereço é <http://pacto4.com.br/servelet/>

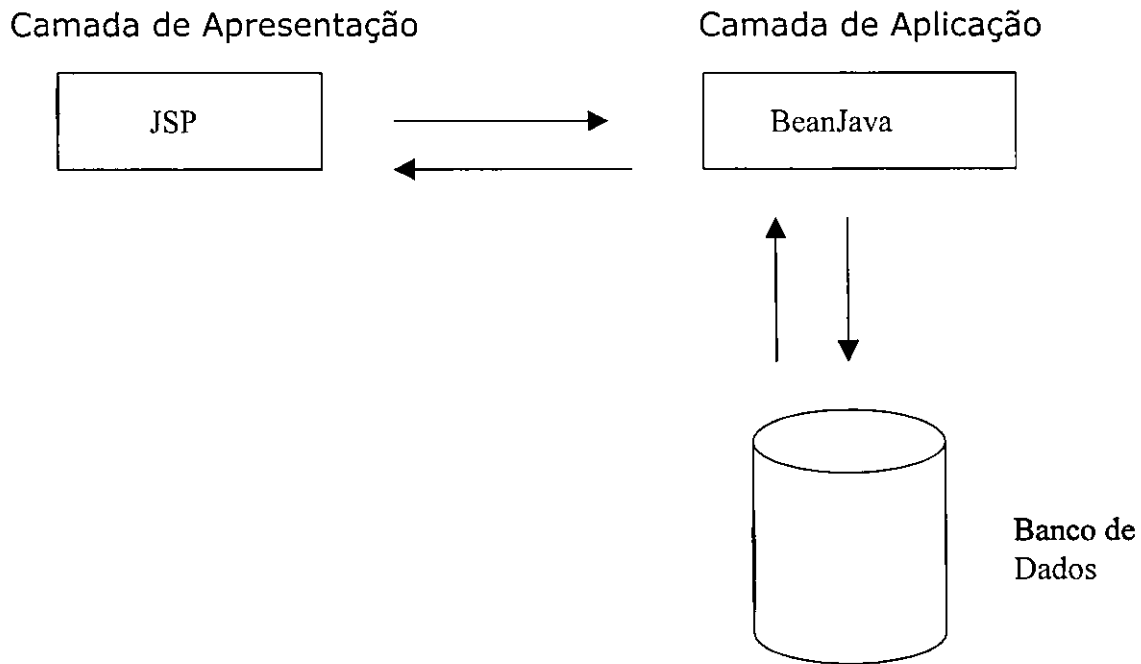
Como sugestão de projetos futuros, fica o desafio de implementar o sistema numa arquitetura corporativa chamada J2EE, possibilitando o uso do sistema por milhares de usuários, inclusão de testes de unidades no código Java através do framework Junit desenvolvido pela Sun e a utilização da ferramenta Cacthos para testes de páginas JSP's.

Enfim, o aprendizado obtido no projeto foi bastante produtivo, pois possibilitou a aquisição de entrada no mercado de trabalho, antes mesmo da conclusão do curso.

ANEXOS

ANEXO 1

1. Arquitetura do Sistema



Camada de Apresentação

Nesta camada se exibe os dados manipulados nas camadas anteriores, através de JSP.

JSP (Java Server Pages) é uma tecnologia para desenvolvimento de aplicações WEB semelhante ao Microsoft Active Server Pages (ASP), 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 produzir 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.

Camada de Aplicação

Nesta camada se localiza a regra de negócio do sistema. Todas as classes Java (Bean Java) e as operações sobre os dados do sistema são distribuídos nesta camada.

É a partir desta camada que se acessa a informações no banco de dados.

Camada de Dados

Na camada de dados ficam armazenados as informações inerentes para que o sistema funcione.

ANEXO 2

2. UML (Unified Modeling Language)

UML é uma linguagem padrão para elaboração da estrutura de projetos de software. A UML poderá ser empregada para a visualização, a especificação, a construção e a documentação de artefatos que façam o sistema complexos de software.

A UML é adequada para a modelagem de sistemas, cuja abrangência poderá incluir sistemas de informação corporativos a serem distribuídos a aplicação baseadas na Web e até sistemas complexos embutidos de tempo real.

Aprender a aplicar a UML de maneira efetiva tem início com a formação de um modelo conceitual da linguagem, o que pressupõe o entendimento de três principais elementos : os blocos básicos de construção da UML, as regras que determinam como esses blocos de construção devem ser combinados e alguns mecanismos básicos que se aplicam a toda a linguagem.

A UML é apenas uma linguagem e, portanto, é somente uma parte de um método para desenvolvimento de software. A UML é independente do processo, apesar de ser perfeitamente utilizada em processo orientado a caso de usos, centrado na arquitetura, interativo e incremental.

3. Diagramas

O Diagrama de *Use Cases* tem o objetivo de auxiliar a comunicação entre os analistas e o cliente.

Um diagrama de *Use Cases* descreve um cenário que mostra as funcionalidades do sistema do ponto de vista do usuário.

O cliente deve ver no diagrama de *Use Cases* as principais funcionalidades de seu sistema.

3.1 Notação

O diagrama de *Use Cases* é representado por:

- atores;
- use cases;

- relacionamentos entre estes elementos.

Estes relacionamentos podem ser:

- associações entre atores e use cases;
- generalizações entre os atores;
- generalizações, *extends* e *includes* entre os use cases.

Estes *use cases* podem opcionalmente estar envolvidos por um retângulo que representa os limites do sistema.

3.2 Em maiores detalhes:

- **Atores**



Um ator é representado por um boneco e um rótulo com o nome do ator. Um ator é um usuário do sistema, que pode ser um usuário humano ou um outro sistema computacional.

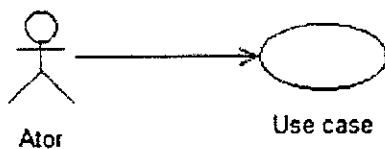
- **Use case**



Um *use case* é representado por uma elipse e um rótulo com o nome do *use case*. Um *use case* é uma funcionalidade do sistema.

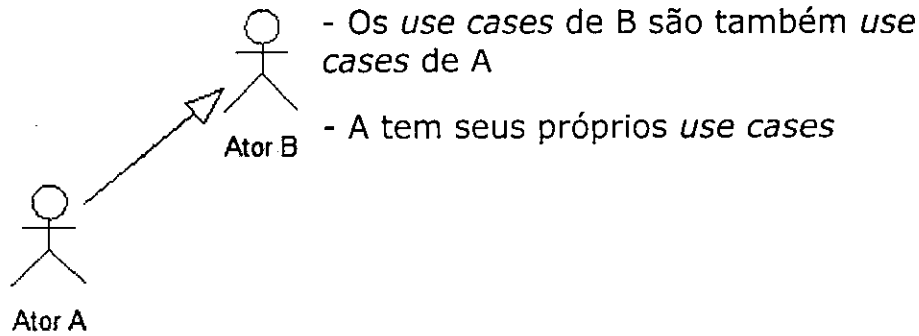
- **Relacionamentos**

- Ajudam a descrever os *use cases*
- Entre um ator e um *use case*
 - Associação



Define uma funcionalidade do sistema do ponto de vista do usuário.

- Entre atores
 - Generalização



4. Diagrama de Seqüência

Consiste em um diagrama que tem o objetivo de mostrar como as mensagens entre os objetos são trocadas no decorrer do *tempo* para a realização de uma operação.

Em um diagrama de seqüência, os seguintes elementos podem ser encontrados:

- Linhas verticais representando o tempo de vida de um objeto (*lifeline*);
- Estas linhas verticais são preenchidas por barras verticais que indicam exatamente quando um objeto passou a existir. Quando um objeto desaparece, existe um "X" na parte inferior da barra;
- Linhas horizontais ou diagonais representando mensagens trocadas entre objetos. Estas linhas são acompanhadas de um rótulo que contém o nome da mensagem e, opcionalmente, os parâmetros da mesma. Observe que também podem existir mensagens enviadas para o mesmo objeto, representando uma iteração;
- Uma condição é representada por uma mensagem cujo rótulo é envolvido por colchetes;
- Mensagens de retorno são representadas por linhas horizontais tracejadas. Este tipo de mensagem não é freqüentemente representada nos diagramas, muitas vezes porque sua utilização leva a um grande número de setas no diagrama, atrapalhando o entendimento do mesmo. Este tipo de mensagem só deve ser mostrada quando fundamental para a clareza do diagrama.

5. Diagrama de Classes

O diagrama de Classes são os encontrados com maior frequência na modelagem de sistemas orientados a objetos. Um diagrama de classe mostra um conjunto de classes, interfaces e colaborações e seus relacionamentos.

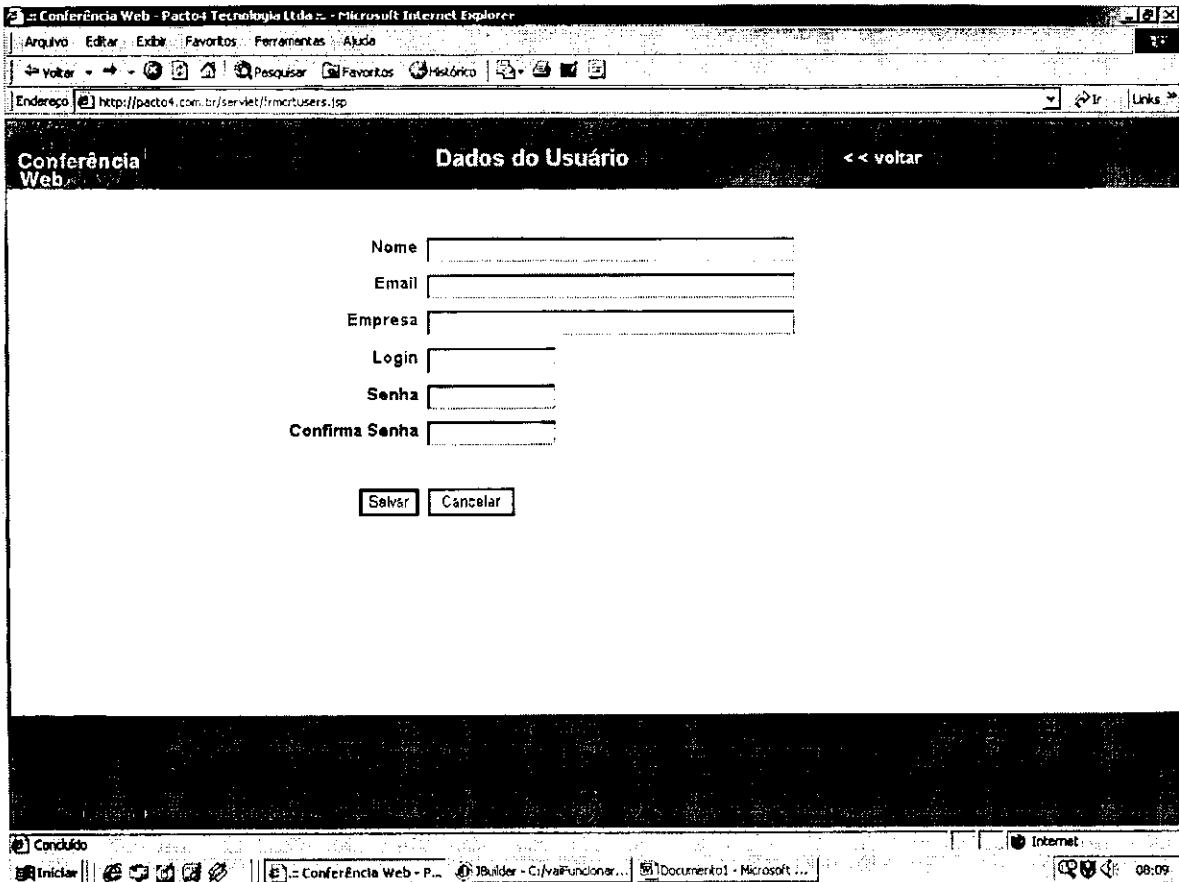
Usa-se os diagramas de classe para fazer a modelagem da visão estática do projeto de um sistema. Na maioria dos casos, isso envolve a modelagem do vocabulário do sistema, a modelagem de colaborações ou a modelagem de esquemas. Os diagramas de classe também são a base para uma par de diagramas relacionados: os diagramas de componentes e diagramas e os diagramas de implantação.

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 e reversa

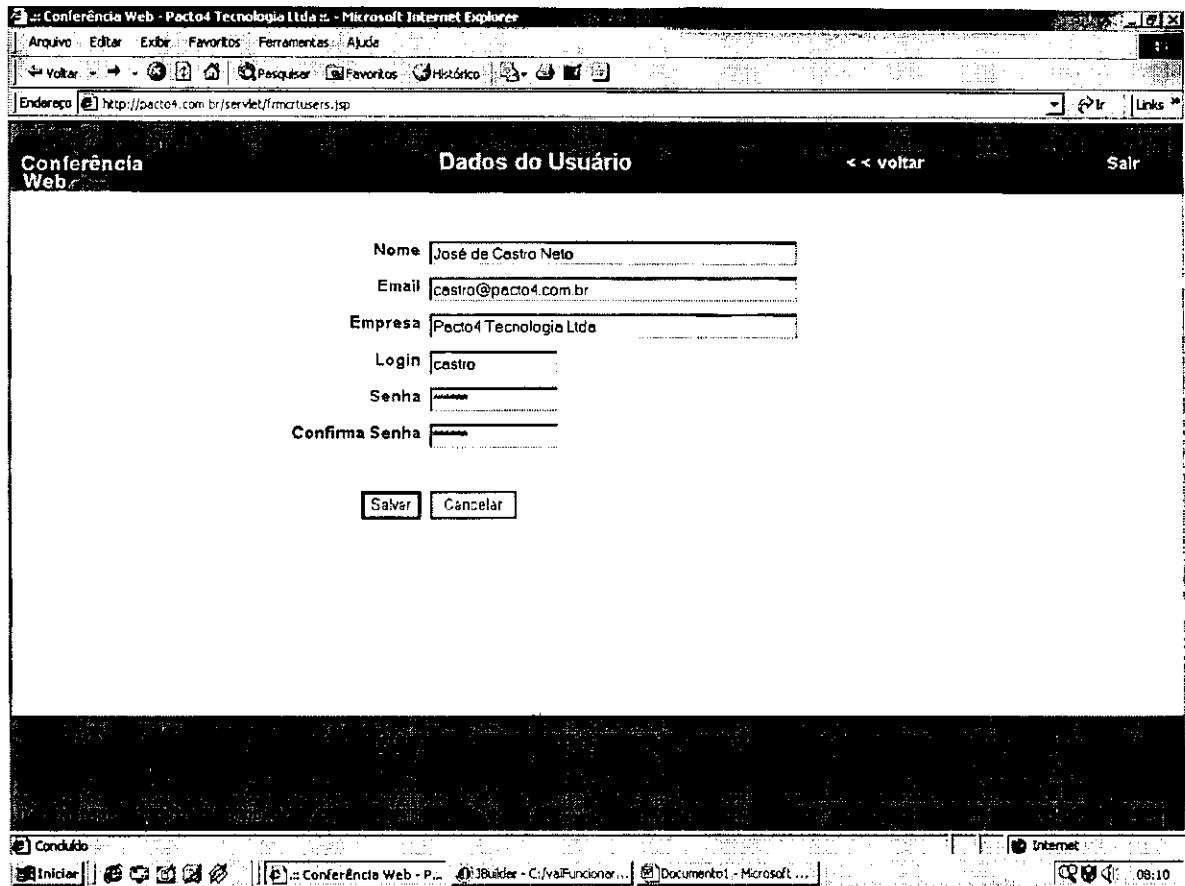
ANEXO 3

Telas do Sistema

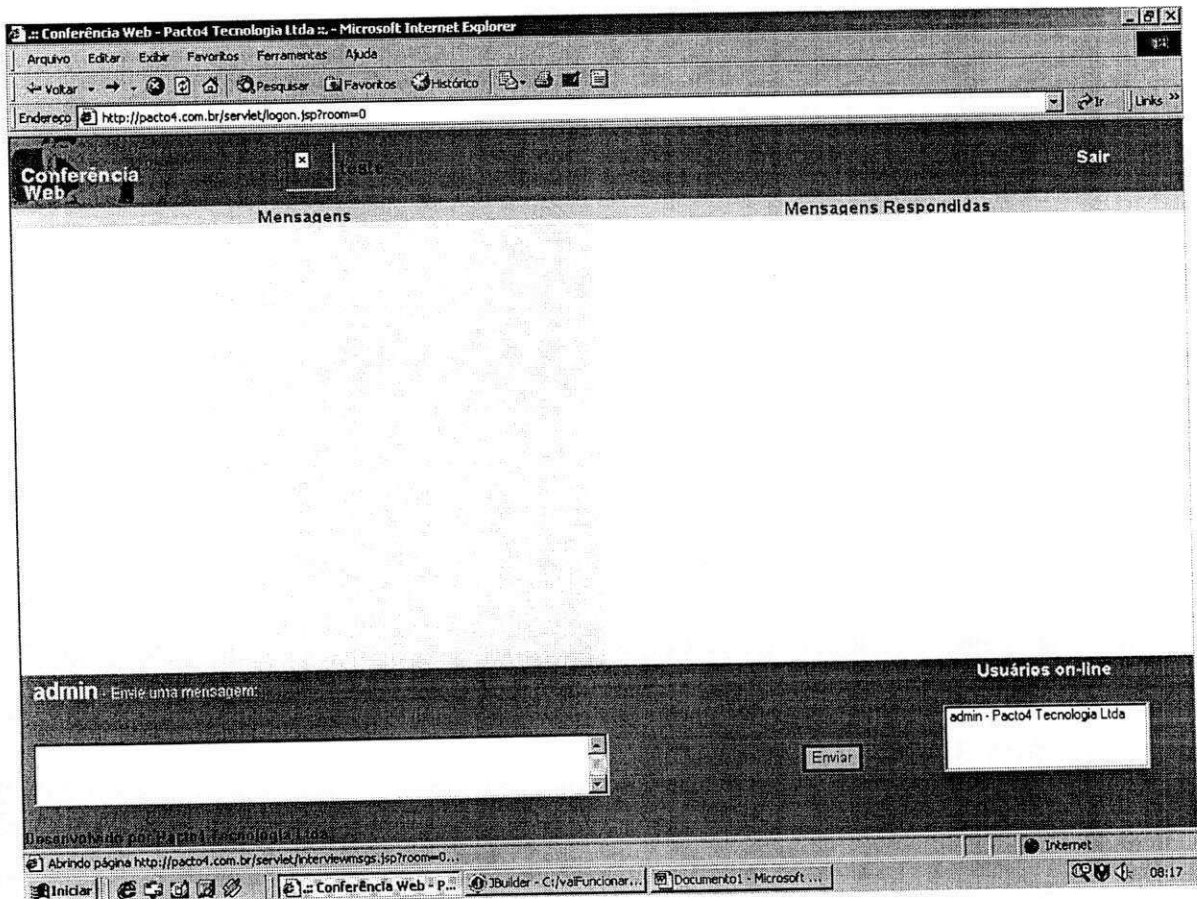
1. Tela de Cadastro



2. Tela de cadastro de Usuários

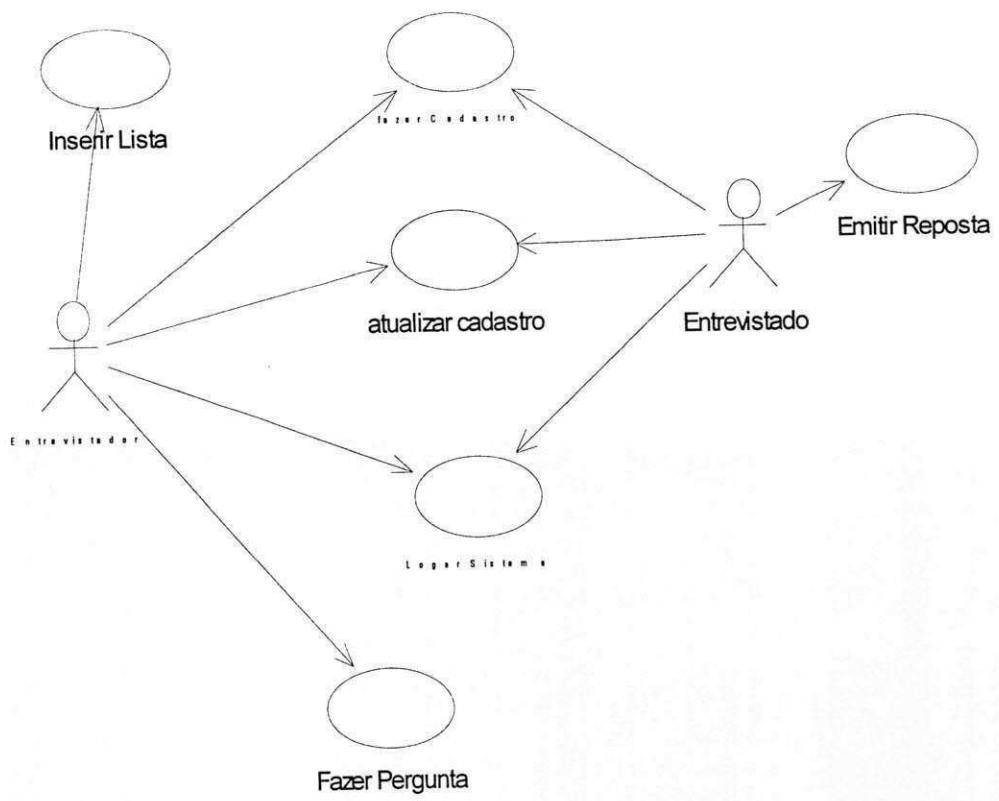


3. Tela donde se processa a entrevista (Sala)



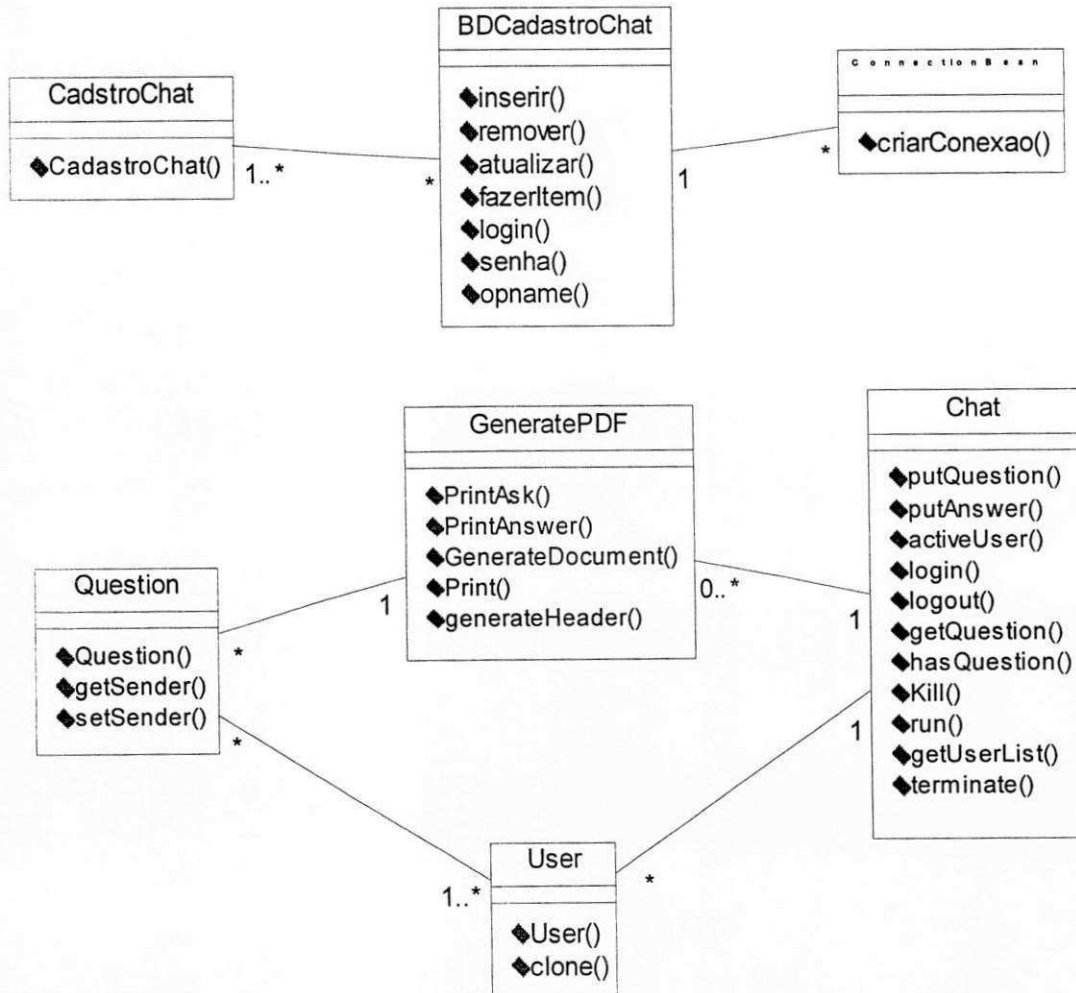
ANEXO 4

5. Diagrama de Use Case



ANEXO 5

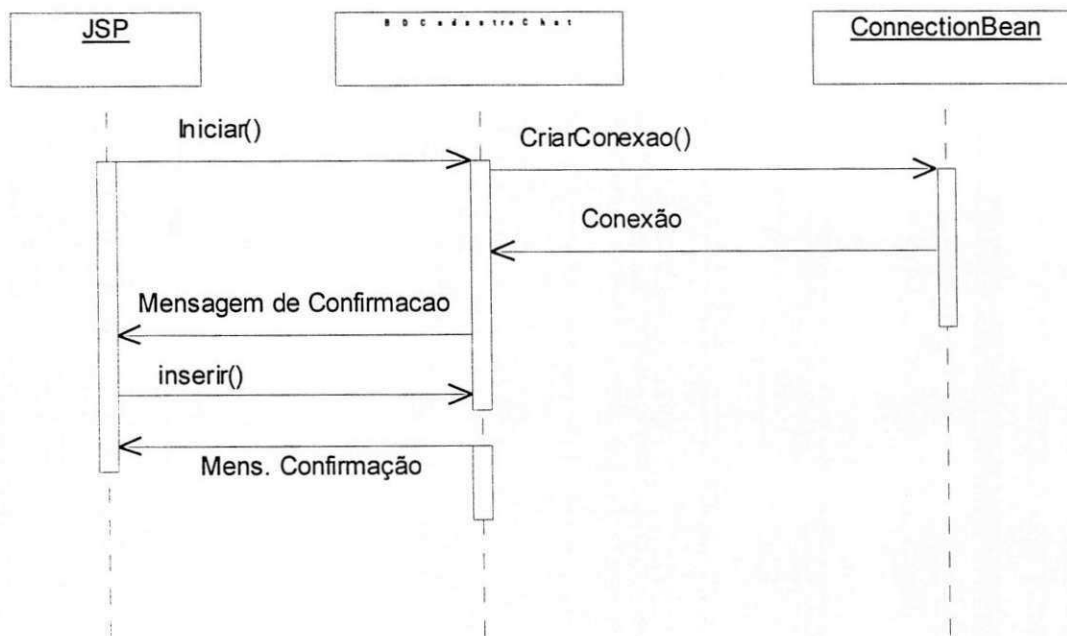
6. Diagrama de Classes



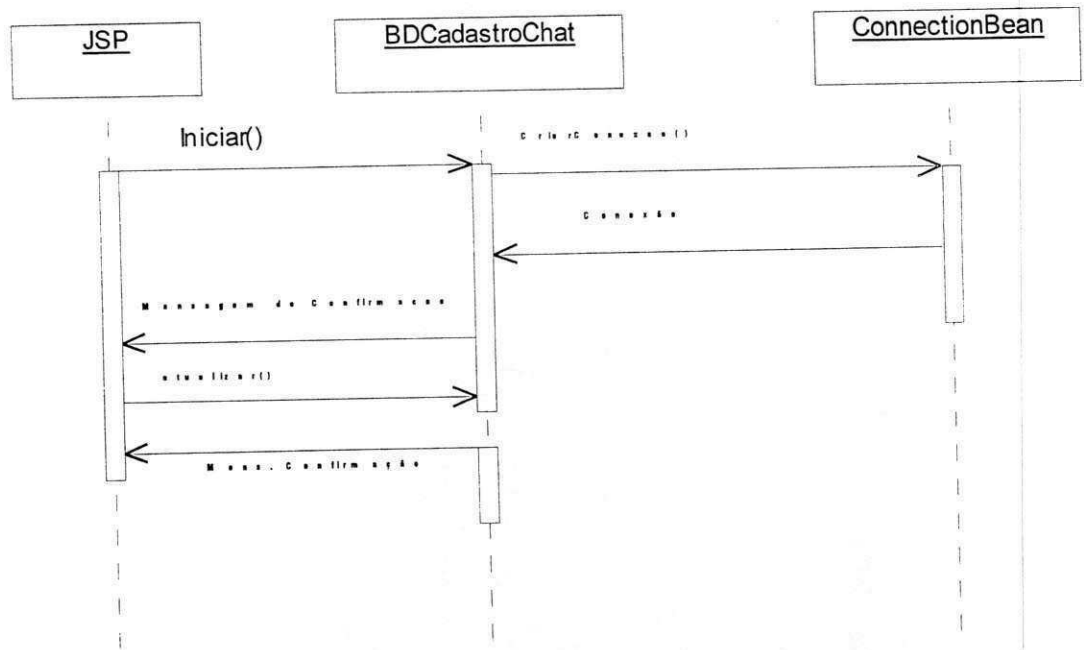
ANEXO 6

7. Diagrama de Sequência

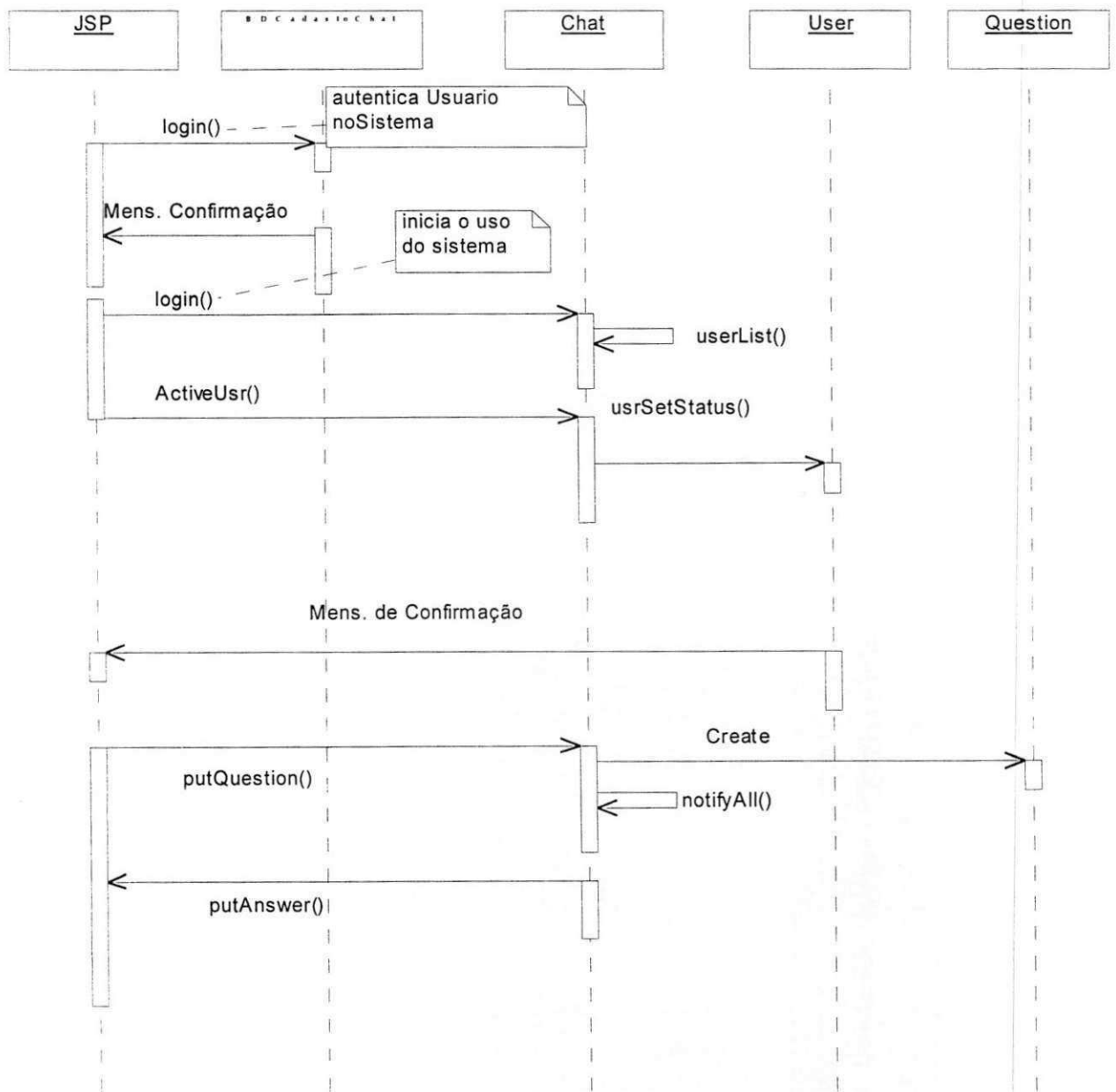
- Inserir Usuário



- Atualizar Usuário



- Fluxo de atividades quando se realiza uma entrevista



ANEXO 7

8. Ferramentas Utilizadas

- JCREATOR – Freeware de fácil manuseio e amplamente usado para desenvolver programas em Java.
- DREAMWEAVER 4.0 – Ferramenta utilizada para desenvolver páginas HTML
- HOMESITE 5.0 - Ferramenta utilizada para implementar código JSP. Facilita a visualização das Tags, diferenciando o HTML do código JSP
- JAKARTA 3.2.2 – servidor de aplicação bastante utilizado e desenvolvido pela *Sun*, empresa inventora da linguagem java. Funciona independentemente da plataforma a qual o sistema está instalado.
- MYSQL - Banco de Dados com suporte a linguagem SQL e também independente de plataforma em que esteja instalado o sistema.
- J2SDK – Interpretador da linguagem Java, desenvolvido pela *Sun*, bastante utilizado em sistemas distribuídos na web.

ANEXO 8

Plano de Estágio

1. Ambiente de Estágio

O estágio será realizado junto a Engebit empresa encubada ao Projeto Poligene – Softex, localizado na Universidade Federal de Campina Grande (UFCG) no 1º andar do Departamento de Sistemas e Computação, o qual dispõe do ambiente de desenvolvimento composto da seguinte estrutura.

- Um servidor IBM Netfinity .
- Sete máquinas Compaq de 300Mhz , com HD 10Gb todas ligadas em Rede.
- Uma impressora Laser.
- Linha Telefônica
- Ar Condicionado

Com o apoio do Poligene a empresa se encontra em estruturação, a Engebit, inicia um processo de desenvolvimento de aplicações para Web. Utilizando tecnologia baseada em Java adentra num nicho de mercado absolutamente rentável e promissor.

2. Supervisão

Supervisor Técnico :

- José de Castro Neto

Supervisor Acadêmico :

- Professor Marcus Salerno de Aquino

3. Resumo do Problema

A empresa Deléon Comunicações realiza entrevista coletiva com convidados em sua sede na grande São Paulo, porém alguns de seus jornalistas credenciados não podem sempre comparecer a entrevista, inviabilizando assim um nível maior de interação entre entrevistado e entrevistadores. Diante deste problema iremos desenvolver um sistema chamado ColetivaWeb que viabilizará o desenrolar de uma entrevista através da internet.

4. Proposta da Solução

O ColetivaWeb será desenvolvido sobre uma arquitetura em três camadas, utilizando de paginas dinâmicas construídas através da linguagem de programação Java e baseadas em Servlet e JSP. A análise do sistema será baseada em UML com descrição de cada modelo e especificação de suas classes e construção dos diagramas que realmente sejam necessários a melhor compreensão do sistema.

O sistema proporcionará um cadastro de repórteres e de entrevistados onde os mesmos poderão dispor de Login e Senha. Uma vez já cadastrado no sistema, o participante da entrevista poderá entrar na "Sala" onde se desenvolverá todo o processo de entrevista, lembrando que um repórter apenas participará da entrevista se houver logicamente um entrevistado na "Sala".

Dentro da "Sala" haverá um local onde os repórteres elaborarão suas perguntas num campo de texto em destaque o qual o entrevistado tem acesso e responderá. Cada usuário dentro da "sala" saberá quem são os participantes da entrevista, pois mostraremos todos através de uma lista ativa enquanto se processa a entrevista.

Evidentemente que uma entrevista acabará quando o usuário (Entrevistado) se ausentar da Sala, finalizando assim toda a entrevista.

5 Atividades a Serem desenvolvidas

	1º semana	2º semana	3º semana	4º semana
Outubro	Estudo do problema e definição da tecnologia a ser utilizada	Modelagem do sistema e diagramação UML	Desenvolvimento do Banco de Dados e reuniões com o orientador	Modularização do sistema e reuniões com o cliente
Novembro	Implementação do 1º módulo (Cadastro de Usuário, inserção e alteração)	Teste de aceitação do Módulo desenvolvido	Implementação do 2º módulo (Autenticação)	Testes de aceitação do Módulo Autenticação
Dezembro	Implementação Modulo de Sincronização de Perguntas - Respostas	Implementação Modulo de Sincronização de Perguntas - Respostas	Implementação Modulo de Sincronização de Perguntas - Respostas e Reunião com Orientador	***** *****
Janeiro	***** *****	Testes de aceitação do Módulo Perguntas e Respostas e Reunião com Orientador	Implementação do Módulo lista Ativa	Implementação do Módulo lista Ativa
Fevereiro	Testes de Aceitação do Modulo Lista Ativa e reunião com o Orientador	Testes Funcionais	Testes Funcionais	Relatório
Março	*****	Relatório	Defesa	

obs : os campos marcados com * são feriados.

9. Aprovação

Supervisor Técnico – José de Castro Neto

Supervisor Acadêmico – Prof Marcus Salerno de Aquino

Estagiário – Sérgio Augusto Freitas Pinto