

UNIVERSIDADE ESTADUAL DO SUDOESTE DA BAHIA
DEPARTAMENTO DE CIÊNCIAS EXATAS
CURSO DE CIÊNCIA DA COMPUTAÇÃO

SAULO EDUARDO DE MENEZES BARROS

**SINEAD – Sistema de Informação para Educação a Distância -
UESB**

Vitória da Conquista - BA

2010

SAULO EDUARDO DE MENEZES BARROS

**SINEAD – Sistema de Informação para Educação a Distância -
UESB**

Monografia apresentada ao
Departamento de Ciências Exatas – DCE –
UESB, como requisito para obtenção do
título de Bacharel em Ciência da
Computação.

Orientador: Adilson de Lima Pereira

Co orientadora: Maria Silva Santos Barbosa

Vitória da Conquista - BA

2010

SAULO EDUARDO DE MENEZES BARROS

**SINEAD – Sistema de Informação para Educação a Distância -
UESB**

Monografia apresentada ao Departamento de Ciências Exatas – DCE –
UESB, como requisito para obtenção do título de Bacharel em Ciência da
Computação.

Banca examinadora:

Maria Silva Santos Barbosa – Co-Orientadora
Universidade Estadual do Sudoeste da Bahia – UESB

Fabício de Sousa Pinto
Universidade Estadual do Sudoeste da Bahia – UESB

Jornandes Jesus Correia
Universidade Estadual do Sudoeste da Bahia – UESB

Dedico esse trabalho a todos que acreditaram no meu potencial e na minha persistência, em especial minha família e para a minha namorada Leilane.

AGRADECIMENTOS

Agradeço muito a...

Deus, por ter me dado força para que eu pudesse lutar nas horas mais difíceis.

Minha família, em especial aos meus pais Gislane Andrade e Francisco Lamartine pelo alicerce que me foi dado, confiança em mim depositada e acima de tudo amor, sinto tanto a falta de vocês...

Meus irmãos Victor e Nathália, pela força e preocupação com meu futuro.

Minha namorada Leilane, pelo carinho, compreensão, demonstração de força e fé, por ter me dado coragem nas horas difíceis da jornada, pelo amor que vem compartilhando comigo durante todo esse tempo e principalmente por me mostrar que basta acreditar e lutar que a recompensa será sempre certa. Você foi mais que fundamental para realização desse sonho!

Gorette e Renival, tio Renaldo e tia Nair, Luciano e Ângela, toda a família Andrade que sempre estão do meu lado no momento que preciso.

Meu brother Vinição, por todos os momentos de alegria que vivenciamos juntos.

Minha Co-orientadora Maria Silva, por ter me dado abertura de espaço e suporte para realização das tarefas.

Todos que me ajudaram de forma direta e indiretamente a conquistar essa vitória.

Muito obrigado!

RESUMO

O trabalho aqui proposto apresenta o desenvolvimento de um sistema web para o setor de Educação à Distância da Universidade Estadual do Sudoeste da Bahia - UESB. Esse sistema oferece maior facilidade ao processo de gerenciamento de dados, que até então demanda muito tempo e espaço no setor. A EAD-UESB possui hoje um setor em pleno crescimento dentro da universidade e, para acompanhar tal crescimento faz-se necessário a criação de um sistema que facilite a administração dando suporte também a armazenamento de dados. O setor de EAD-UESB cadastra por ano uma média de 2.000 alunos necessitando assim que o processo de matrícula e administração seja informatizado, já que esses processos são de importância vital para a organização do setor. O sistema SINEAD fará o gerenciamento de informações sobre os alunos, os tutores, os coordenadores, os ciclos, os cursos, as ofertas e turmas que existem no programa de EAD-UESB, já que todos estarão interligados dentro do mesmo processo. Dessa forma o sistema será de grande utilidade para o setor, além do que desenvolvimento foi feito pensando na portabilidade e desempenho, podendo ser configurado e utilizado não só para a Educação a Distância da UESB em Vitória da Conquista, como também pelos outros campi da Universidade.

Palavras-chave: Sistema web. Educação a distância.

LISTA DE FIGURAS

Figura 01: Diagrama de casos de uso-----	24
Figura 02: Modelo do módulo Cadastro-----	25
Figura 03: Modelo do módulo Administração-----	26
Figura 04: Tela de Login-----	27
Figura 05: Administrador logado-----	28
Figura 06: Funcionário logado-----	28
Figura 07: Menu Administração-----	29
Figura 08: Listagem de Usuários-----	30
Figura 09: Listagem de Papéis-----	31
Figura 10: Listagem Usuário Papel-----	31
Figura 11: Listagem Permissão-----	32
Figura 12: Menu Cadastro-----	33
Figura 13: Listagem de Alunos-----	34
Figura 14: Cadastro de Alunos novos-----	34
Figura 15: Cadastro de Coordenadores-----	35
Figura 16: Listagem de Coordenadores-----	35
Figura 17: Mensagem de erro: campos obrigatórios-----	36
Figura 18: Campos Data com máscara-----	37
Figura 19: Filtro por CPF-----	38
Figura 20: Filtro por Turma-----	38

LISTA DE ABREVIATURAS E SIGLAS

CSS	Cascading Style Sheet
EAD	Educação à Distância
EUA	Estados Unidos da América
FK	Foreign Key
HTML	HyperText Markup Language
IDE	Integrated Development Environment
IHM	Interface Homem-Máquina
JEE	Java Enterprise Edition
MVC	Model-View-Controller
UESB	Universidade Estadual do Sudoeste da Bahia
UML	Unified Modeling Language
PDF	Portable Document Format
PK	Primary Key
XML	Extensible Markup Language
XP	Extreme Programming
W3C	World Wide Web Consortium

SUMÁRIO

1	INTRODUÇÃO	11
1.1	PROGRAMA EAD – UESB	11
1.2	OBJETIVO GERAL	13
1.3	OBJETIVOS ESPECÍFICOS	13
1.4	JUSTIFICATIVA	13
1.5	METODOLOGIA	14
2	FUNDAMENTOS TEÓRICOS	15
2.1	METODOLOGIAS DE DESENVOLVIMENTO	15
2.1.1	Metodologia de Desenvolvimento Ágil	15
2.1.2	Extreme Programming (XP)	16
2.2	MODELAGEM DE SISTEMAS	16
2.3	TECNOLOGIAS UTILIZADAS	18
2.3.1	Java	18
2.3.2	Next framework	18
2.3.3	Hibernate	19
2.4	PRINCÍPIOS DE INTERFACE HOMEM-MÁQUINA	19
2.4.1	Usabilidade	20
2.4.2	Acessibilidade	20
2.4.3	Comunicabilidade	21
3	DESENVOLVIMENTO DO SISTEMA	22
3.1	ESTRUTURA DO SISTEMA	22
3.2	MODELAGEM DO SISTEMA	22
3.2.1	Análise de requisitos	23
3.2.1.1	Requisitos Funcionais	23
3.2.1.2	Requisitos não funcionais	23
3.2.2	Diagrama de casos de uso	24
3.2.3	Modelo Entidade Relacionamento	24
4	APRESENTAÇÃO DO SISTEMA SINEAD	27
4.1	MENU ADMINISTRAÇÃO	29
4.1.2	Submenu Usuário	30
4.1.3	Submenu Papel	30
4.1.4	Submenu UsuárioPapel	31

4.1.5 Submenu Permissão-----	32
4.2 MENU CADASTRO-----	32
4.2.1 Submenu Aluno-----	33
4.2.2 Submenu Coordenador-----	35
4.3 MENSAGENS DE ERRO-----	36
4.4 MÁSCARA PARA CAMPOS DO FORMULÁRIO-----	36
4.5 FILTROS DE LISTAGEM-----	37
4.5.1 Filtro CPF -----	37
4.5.2 Filtro Turma-----	38
5 CONCLUSÃO E TRABALHOS FUTUROS-----	39
6 REFERÊNCIAS-----	40
APÊNDICE 1 - Descrição do Diagrama de Casos de Uso-----	41

1 INTRODUÇÃO

O trabalho aqui proposto apresenta o desenvolvimento de um sistema web que facilitará o processo de gerenciamento de dados do setor de Educação à Distância da UESB - Universidade do Sudoeste da Bahia, que até então demanda muito tempo dos funcionários e espaço físico do setor.

A EAD-UESB possui hoje um setor em pleno crescimento dentro da universidade. E para acompanhar esse crescimento faz-se necessário a criação de um sistema que facilite a administração, o controle e o armazenamento de dados. Esse sistema proporcionará ao setor de Educação à Distância da UESB gerenciar as informações de todos os sujeitos participantes do programa EAD-UESB,

E como hoje, é indiscutível a participação da Internet na vida das pessoas, o ideal é que este sistema fosse Web, possibilitando o acesso a partir de qualquer computador que esteja ligado à Internet.

Segundo Litto e Formiga (2009), com recursos destinados a Lei de Incentivo Audiovisual, ou a Lei Rouanet de amparo à cultura, seria possível estruturar um sistema que trataria de assuntos relacionados à Educação a Distância.

Na produção de softwares, para atender à demanda que surgiria poderíamos pensar em como os programas de educação à distância seriam gerenciados por todo o país. Hoje já existem softwares como o TELEDUC (ambiente em desenvolvimento no Núcleo de Informática Aplicada à Educação (NIED) da Universidade Estadual de Campinas) que administra, cria e propõe participações via web para participantes de cursos a distância.

Pode-se pensar também em softwares que auxiliariam os setores responsáveis pela EAD individualmente, e foi pensando neste sentido que este trabalho foi desenvolvido.

1.1 PROGRAMA EAD – UESB

O programa de Educação a Distância da Universidade Estadual do Sudoeste da Bahia (neste trabalho será sempre citado como EAD-UESB) está, hoje, instalado

no módulo Glauber Rocha que fica no campus de Vitória da Conquista. O programa conta com uma estrutura física modesta, mas em acelerado crescimento.

No que diz respeito aos recursos humanos o programa possui quatro grupos de colaboradores: o Coordenador que pode ser Coordenador Geral, Coordenador de Curso, Coordenador de Tutor e Coordenador de Ciclo; o Tutor que pode ser Tutor Presencial e Tutor Virtual (à distância), Professor que pode ser Professor Pesquisador, Professor Conteudista e Professor Executor; e o Aluno. No que diz respeito à estrutura dos cursos, o projeto de educação à distância é organizado por Ofertas, cada Oferta possui três Ciclos (básico, intermediário e avançado) e dentro dos Ciclos tem-se os Módulos, que são os conteúdos ministrados pelos professores.

O programa oferece o curso de Formação Continuada Mídias na Educação e é destinado aos professores da Educação Básica, Educação Infantil, Educação Especial, Educação de Jovens e Adultos e de Profissionais Graduados de áreas ligadas ao Magistério e à Gestão Educacional. Tendo como principal objetivo proporcionar a formação continuada para o uso pedagógico das diferentes tecnologias da informação e da comunicação – TV, vídeo, informática, rádio e impressos – de forma integrada ao processo de ensino e aprendizagem, contribuindo para a formação de um leitor crítico e criativo.

Hoje a EAD-UESB tem mais de 5000 alunos e conta com uma equipe selecionada, e em constante crescimento. O programa usando as diversas mídias de início com o Módulo Básico, que teve sua primeira oferta com 700 (setecentas) vagas em 2006, em seguida o Ciclo Básico do Programa de Formação Continuada em Mídias na Educação, foi oferecido, numa segunda oferta com 1200 (mil e duzentas) vagas, e o ciclo intermediário, com 435 cursistas cuja clientela foi, também, clientela do Módulo Avançado.

Paralelo a oferta do Ciclo Avançado, foi oferecida a segunda oferta do Ciclo Intermediário, e ainda o ciclo Básico, em sua terceira oferta, com 3623 alunos. Todo esse processo de preparação existe com o foco voltado para a atuação dos professores em sala de aula, e chega ao seu quarto ciclo pretendendo completar o conteúdo dos ciclos anteriores, a fim de promover as ações que realmente possam aumentar a performance do docente.

1.2 OBJETIVO GERAL

Desenvolver um sistema Web que permite o gerenciamento de matrículas de alunos e acesso de informações on-line dos sujeitos envolvidos no programa EAD-UESB.

1.3 OBJETIVOS ESPECÍFICOS

- Gerenciar informações do setor de EAD-UESB;
- Agilizar o processo de matrícula;
- Oferecer satisfação, facilidade e versatilidade para o usuário;
- Permitir interface natural e intuitiva, utilizando os princípios de IHM;
- Disponibilizar a ferramenta para utilização via Web para que os funcionários da EAD-UESB possam utilizá-la de qualquer computador conectado à internet.

1.4 JUSTIFICATIVA

A EAD-UESB possui hoje um setor em pleno crescimento dentro da Universidade. A parte administrativa é feita totalmente dentro do setor EAD-UESB e os funcionários precisam fazer uso de uma planilha impressa para cada aluno, tutor e coordenadores que participam desse projeto, ou seja, para cada um desses sujeitos citados que seja participante do programa de educação a distância, terá um registro feito manualmente em um formulário de papel e guardado em um armário de arquivos. Isso torna o processo demorado e altamente custoso.

Para acompanhar tal crescimento faz-se necessário a criação de um sistema que facilite a administração, o controle e o armazenamento de dados. O setor cadastra por ano uma média de 2.000 (dois mil) alunos, necessitando assim que o processo de matrícula e administração seja informatizado, já que tais processos são de importância vital para a organização do setor.

Neste sentido o desenvolvimento do SINEAD ajudará de forma muito significativa, pois ele foi desenvolvido para gerenciar essas informações e agilizar o processo de matrícula e facilidade de acesso às informações nele contidas.

1.5 METODOLOGIA

Para o desenvolvimento deste trabalho foram feitas pesquisas de caráter bibliográfico e exploratório, com objetivos de colher informações suficientes sobre todos os processos existentes na EAD-UESB e escolher as melhores tecnologias para o desenvolvimento do sistema.

Utilizando a metodologia de programação *Extreme Programming*, que traduz uma das formas de se aplicar os métodos ágeis de desenvolvimento, foram feitos questionamentos diretamente aos funcionários e futuros usuários da ferramenta na EAD-UESB, a fim de levantar as informações necessárias, bem como, de que forma esse sistema poderia ser útil ao setor. Essas informações foram imprescindíveis ao desenvolvimento do projeto.

Com esses questionamentos, e com uma leitura detalhada sobre o projeto foi adquirido conhecimento sobre todos os processos existentes na EAD-UESB, o que serviu de base para o desenvolvimento do sistema.

A metodologia de programação *Extreme Programming* exige que o desenvolvedor esteja em contato com o cliente, e ao mesmo tempo, escreva o código, tire dúvidas e execute os códigos escritos, fazendo integração entre as tarefas propostas para que o objetivo desejado seja alcançado com maior eficácia.

Essa metodologia é baseada na aquisição de resultados práticos, rápidos e com código de fácil entendimento e manutenção. Isso facilita bastante o desenvolvimento do sistema, pois diminui a quantidade de possíveis erros e aumenta a produtividade.

Posteriormente, no próximo capítulo, serão apresentadas as ferramentas e tecnologias escolhidas para o desenvolvimento do sistema.

2 FUNDAMENTOS TEÓRICOS

Conhecimentos utilizados como base para a produção do sistema.

2.1 METODOLOGIAS DE DESENVOLVIMENTO

A metodologia escolhida para o desenvolvimento do sistema foi Metodologia de Desenvolvimento Ágil, *Extreme Programming (XP)*. Ela foi escolhida levando em consideração a forma como é aplicada e a objetividade.

2.1.1 Metodologia de Desenvolvimento Ágil

A programação cotidiana inicia-se com uma tarefa conectada de forma clara a uma funcionalidade desejada, seguida de testes, passando pela implementação e pelo projeto, terminando na integração entre essas tarefas. Um pouco de cada uma das atividades do desenvolvimento de software está incluída em cada episódio. (Beck, 2000, p. 25).

O desenvolvimento de software está sofrendo mudanças em um ritmo cada vez mais acelerado, e condicionando a tratar os problemas no desenvolvimento de forma mais rápida e ágil. Dessa forma é que está surgindo uma forte tendência de utilização dos Métodos Ágeis de Desenvolvimento.

Segundo Scott W. Ambler (2004), os métodos ágeis estão sendo cada vez mais utilizados pelas equipes de desenvolvimento de software, pois utilizam uma abordagem simples e prática do processo. Ele ratifica também que os métodos ágeis não podem ser confundidos com falta de interesse, organização ou falta de controle da equipe de desenvolvimento, pois, os métodos ágeis exigem que a equipe seja disciplinada e muito bem organizada.

Ainda por Scott W. Ambler (2004), as metodologias ágeis descrevem um estilo de modelagem que ao serem usadas corretamente resultam em softwares de melhor qualidade e de desenvolvimento mais rápido, ao mesmo tempo em que evita simplificações excessivas e expectativas não realistas. A metodologia ágil não é uma receita de bolo, ela é um suplemento aos métodos preexistentes.

2.1.2 *Extreme Programming (XP)*

O método de desenvolvimento *XP* possui uma objetividade bastante acentuada e está focada em obter resultados em curto prazo denominados *testes*, em que o código e o teste são escritos e colocados para “rodar” (executando para obter resultados). Caso sejam executado com sucesso estes códigos e testes são então salvos em um repositório compartilhado para que possam ser acessados por qualquer integrante do grupo de desenvolvimento. Isso permite que qualquer pessoa da equipe saiba o que está acontecendo, podendo então alterar os dados à medida que for necessário.

Esse método foca a simplicidade e a prática de refatoração, resultando em um código muito claro e limpo. Se o código já foi fácil de entender, porque investir muito tempo escrevendo documentação para ajudar a entendê-lo? Ambler (2004)

Como toda metodologia ágil, o *XP* diminui a carga de documentação que é gerada e aumenta o contato com o cliente, pois a equipe de desenvolvimento e de clientes trabalha simultaneamente (em contato constante), para que possam determinar o que precisam, criando um ambiente de retorno rápido e confiável para ambas as partes.

Um dos maiores equívocos relacionados ao *XP* é achar que a carga de trabalho será menor pelo fato da diminuição da documentação. Nada poderia estar mais longe da verdade. Em um projeto que utiliza *XP* a decisão de criar ou não documentação é questão de negócio e não de técnica. Se houver a necessidade de um documento, o cliente deve solicitá-lo da mesma forma que solicitaria uma característica do programa. A partir desse ponto a equipe estimará o custo da produção dessa documentação e agendará com o cliente a entrega.

2.2 MODELAGEM DE SISTEMAS

Segundo Martim Fowler (2004) para que se inicie o desenvolvimento de sistema faz-se necessário um planejamento. O embasamento para esse

planejamento normalmente está ligado a um conceito de modelo, amparado por um modelo é que se dá características essenciais para o sistema.

Por esse motivo, sentiu-se a necessidade de criar uma modelagem padrão para que pudesse ser utilizada e aceita amplamente. Em 1996 surgiu então a UML (*Unified Modeling Language*).

É através da utilização de modelos que poderemos identificar os custos; podendo diminuí-los, identificar erros e fazer correções ainda na fase inicial do sistema. Por meio deles poderemos também gerenciar melhor o sistema, prever comportamentos normais e anormais que possam ocorrer e ainda temos a facilidade de interação entre as pessoas envolvidas no projeto de desenvolvimento.

Nesse contexto de desenvolvimento do software temos os diagramas que correspondem à representação de gráficos seguindo algum padrão lógico; se pode também dizer que um diagrama é uma apresentação de uma coleção de elementos gráficos que possuem um significado predefinido e são normalmente construídos de acordo com regras de notação bem definidas; cada forma utilizada em um diagrama de modelagem tem um significado específico.

Segundo Ricardo Argenton Ramos (2002), diagramas permitem a construção de uma representação concisa de um sistema a ser construído. “Uma figura vale por mil palavras”, no entanto, modelos também são compostos de informações textuais.

Dado um modelo de uma das perspectivas de um sistema, diz-se que o seu diagrama, juntamente com a informação textual associada, forma a documentação deste modelo.

2.3 TECNOLOGIAS UTILIZADAS NO DESENVOLVIMENTO DO SISTEMA

2.3.1 Java

Essa linguagem foi criada pela empresa *Sun Microsystems (EUA)* como parte de um projeto denominado “Projeto Green”, em 1991. A idéia era desenvolver uma linguagem que pudesse ser útil para o controle de eletrodomésticos. Inicialmente ela era utilizada apenas como ferramenta para os programadores, mas mostrou-se bem mais poderosa. A linguagem JAVA possui algumas características básicas como:

- É orientada a objeto;
- É implementada gerando um código intermediário que posteriormente é executado por uma máquina virtual;
- É portátil;
- É neutra com relação à arquitetura computacional;
- É simples;
- Executa multitarefa;
- Produz solução mais elegante multitarefa.

Segundo Mário Leite (2008), Java hoje é utilizada em aplicações que podem ser locais ou em rede, tem uma capacidade extremamente grande podendo ser empregada em diversos ramos da Tecnologia da Informação como bancos de dados, *WEB*, interfaces gráficas, conteúdo multimídia, podendo também ser implementada através de *frameworks*, que são aplicações que servem de suporte ao desenvolvimento de software num determinado contexto.

2.3.2 Next framework

Segundo Rógel Garcia (2010), o Next é um *framework* JEE baseado em Spring e Hibernate e tem como foco principal a simplicidade no desenvolvimento de aplicações web. A idéia do Next é facilitar o desenvolvimento de aplicações resolvendo problemas, mas sem adicionar novos paradigmas ou complicadores.

A principal característica do NEXT é a simplicidade na sua configuração, pois não existe a necessidade de XML e é possível criar novas aplicações completamente configuradas em pouquíssimo tempo.

Essa configuração simplificada, aliada ao design pattern Convention Over Configuration (Convenção ao invés de configuração), permite a criação de aplicações muito mais rapidamente, com menos bugs e conseqüentemente menos manutenção. Por serem baseadas em Spring e Hibernate, todas as funcionalidades desses frameworks podem ser utilizadas sem problemas em conjunto com o Next.

2.3.3 *Hibernate*

O *Hibernate* é um *framework* de mapeamento objeto relacional para aplicações Java, ou seja, é uma ferramenta para mapear classes Java em tabelas do banco de dados e vice-versa. É bastante poderoso e dá suporte ao mapeamento de associações entre objetos, herança, polimorfismo, composição e coleções.

Como menciona Craig Larman (2005), o *Hibernate* resolve a maioria dos problemas de mapeamento objeto-relacional, desempenho, apoio a transações etc.

O *Hibernate* não apresenta apenas a função de realizar o mapeamento objeto relacional, também disponibiliza um poderoso mecanismo de consulta de dados, permitindo uma redução considerável no tempo de desenvolvimento da aplicação.

Segundo Fábio Akita 2006, o *Hibernate* é um dos *frameworks* mais consagrados e utilizados com linguagem de programação Java, porém mesmo sendo muito eficaz, em muitos casos, no cotidiano o *Hibernate* vem se mostrando um *framework* cansativo e que exige muita atenção do desenvolvedor.

2.4 PRINCÍPIOS DE INTERFACE HOMEM-MÁQUINA

IHM é o canal de comunicação entre o homem e o computador, através do qual interagem, visando atingir um objetivo comum. Um conjunto de características com o qual os usuários interagem com as máquinas, dispositivos, programas de computador ou alguma outra ferramenta complexa.

Para trabalhar com um sistema, o utilizador precisa ter domínio sobre ele. Por exemplo, quando estamos pilotando uma moto precisamos utilizar o guidão, os freios e acelerador, tudo isso para guiar a moto sem maiores problemas. A moto disponibiliza ao piloto dados de velocidade, bem como a quantidade de gasolina.

Enquanto o acelerador e freios são dispositivos de entrada, os dados do hodômetro são dispositivos de saída.

Da mesma forma para utilizar um sistema de computador o usuário precisa ter domínio sobre o equipamento. Para isso, existem diversos princípios de IHM, que facilitam a interação entre os homens e os sistemas computacionais.

2.4.1 Usabilidade

O projeto de uma interface de utilizador influencia diretamente na quantidade de esforço que ele utilizará para usar as entradas no sistema e para entender sua respectiva saída, além de quanto esforço ele precisará para aprender o procedimento. A usabilidade é uma área do design que leva em consideração a psicologia e a fisiologia dos usuários para tornar os sistemas mais efetivos, eficientes e satisfatórios. Ela está principalmente associada às características da interface do usuário, mas também pode estar associada com a funcionalidade do produto.

Segundo Antonio Andrade (2007), um sistema deve manter o usuário sempre informado sobre o que está acontecendo por meio de um *feedback* apropriado dentro de um tempo razoável. Deve ainda falar a linguagem do usuário com palavras, frases e conceitos que sejam familiares, ao invés de termos orientados ao sistema. Deve seguir convenções do mundo real, fazendo a informação aparecer em uma ordem natural lógica.

2.4.2 Acessibilidade

A acessibilidade é um tema que vem sendo massivamente discutido. Atualmente estão em curso obras e serviços de adequação do espaço urbano para a inclusão de pessoas com algum tipo de deficiência ou dificuldade. Com a informática não poderia ser de outra forma, a quantidade de pessoas que hoje fazem uso da informática é extremamente grande e um número bastante elevado também é o de pessoas que não conseguem fazer uso da informática por falta de acessibilidade.

Hoje há programas que provêm acessibilidade possuindo ferramentas ou conjuntos de ferramentas que permitem que portadores de deficiências utilizem

recursos que o computador oferece. Essas ferramentas podem constituir leitores de ecrã para deficientes visuais, teclados virtuais para portadores de deficiência motora ou com dificuldades de coordenação motora, e sintetizadores de voz para pessoas com problemas de fala.

Na Internet o termo acessibilidade refere-se também a recomendações do W3C, que visam permitir que todos possam ter acesso aos websites, independente de terem alguma deficiência ou não. As recomendações abordam desde o tipo de fonte a ser usado, bem como seu tamanho e cor, de acordo com as necessidades do usuário, até as recomendações relativas ao código (HTML e CSS, por exemplo).

2.4.3 Comunicabilidade

A comunicabilidade é a capacidade de compreensão que deve existir entre quem vê e o que está sendo visto. Esse processo é considerado metacomunicativo, pois utiliza signos e códigos para melhorar a compreensão entre as partes. Entretanto e para tal tarefa devem-se observar alguns pontos como:

Naturalidade: Seqüência de interações organizada da maneira mais natural para o usuário.

Flexibilidade: Permissão de interações para um mesmo comando.

Não-modal: Permissão ao usuário consultar outras informações durante o comando e depois retornar ao ponto onde estava.

Perdão: Permissão ao usuário corrigir ou voltar atrás.

Unidade: Procurar colocar os comandos em uma única janela. Se tiver mais de um comando na mesma janela, construir grupos separados.

O objeto de interação deve ser escolhido de forma adequada para a ação que o usuário deve fazer, sendo o objeto aquele com o qual o usuário está familiarizado. O vocabulário a ser usado pelo usuário deve ser de fácil domínio, de modo que devem ser evitados termos técnicos. Os resultados devem fazer referências às informações passadas no comando e os avisos devem ser colocados em primeiro plano para chamar a atenção do usuário.

Cores fortes e diferenciadas de outros elementos devem ser usadas para chamar atenção.

3 DESENVOLVIMENTO DO SISTEMA

Na modelagem do SINEAD foram utilizadas as ferramentas ASTAH e DBDesign. No ASTAH foram definidos os casos de uso do sistema, já o DBDesign foi utilizado para criar o modelo do banco de dados, possibilitando de maneira ágil e fácil especificar classes, atributos e relacionamentos.

Na implementação, foi utilizado a IDE (Integrated Development Environment) *Eclipse Galileo* executado sobre a plataforma do sistema operacional Linux, ligado a um servidor web *TomCat*.

A codificação foi desenvolvida com a linguagem de programação Java e com o auxílio do *framework NextFramework*.

O funcionamento do sistema se baseia na estrutura Cliente/Servidor, em que o cliente solicita um resultado e o servidor se encarrega de mandar a resposta.

3.1 ESTRUTURA DO SISTEMA

A página inicial do sistema é gerada através de uma requisição feita pelo cliente, o servidor por sua vez gera uma tela de login, de onde se pode ter acesso à página principal (caso se tenha efetuado login). Na página inicial se encontram os *links* de início, menu cadastro, menu administração e sair.

O SINEAD está dividido em dois módulos que são o CADASTRO e ADMINISTRAÇÃO, como se pode ver na página inicial. No módulo CADASTRO tem todos os tipos de cadastro que serão necessários para o setor EAD-UESB, que são: Aluno, Tutor, Coordenador, Turma, Ciclo, Oferta, Curso, Ciclo-Módulo, Módulo.

No módulo ADMINISTRAÇÃO, faz-se o controle de permissão do sistema, nele é possível fazer os seguintes cadastros: Usuário, Papel, Papel-Usuário e Permissão.

3.2 MODELAGEM DO SISTEMA

A modelagem do sistema foi feita em três etapas: Levantamento de requisitos, definição dos casos de uso e desenvolvimento do modelo entidade relacionamento do banco de dados. Serão mostradas cada uma das etapas na seqüência.

3.2.1 Levantamento de requisitos

3.2.1.1 Requisitos Funcionais

Os requisitos funcionais especificam ações que um sistema que deve ser capaz de executar, sem levar em consideração restrições físicas (geralmente, isso é melhor descrito em um modelo de casos de uso), eles especificam, portanto, o comportamento de entrada e saída de um sistema. Inicialmente, o SINEAD tem como principais funcionalidades:

- Permitir acesso apenas a pessoas autorizadas;
- Manter cadastro de Aluno, Tutor, Coordenador;
- Manter cadastro de Curso, Oferta, Turma, Ciclo, Ciclo-Módulo, Modulo;
- Manter o cadastro de Permissão, Usuário, Papéis, UsuáriosPapéis

3.2.1.2 Requisitos não funcionais

Os requisitos não funcionais definem as qualidades do sistema como: desempenho, usabilidade, confiabilidade, segurança, disponibilidade e manutenibilidade.

A qualidade de um software também pode ser avaliada de duas maneiras: a visível para o usuário final e a interna visível em tempo de desenvolvimento.

O sistema SINEAD foi desenvolvido visando atender as necessidades de utilização do usuário, possui uma interface amigável que transmite todas as características acima citadas.

Para atender às necessidades de performance e desempenho, o sistema SINEAD foi desenvolvido com tecnologias atuais e conta com uma boa plataforma de funcionamento.

3.2.2 Diagrama de casos de uso

O diagrama de casos de uso representa de forma abstrata como os elementos externos interagem com as funcionalidades do sistema.

A Figura 01 abaixo mostra o diagrama de casos de uso do sistema SINEAD.

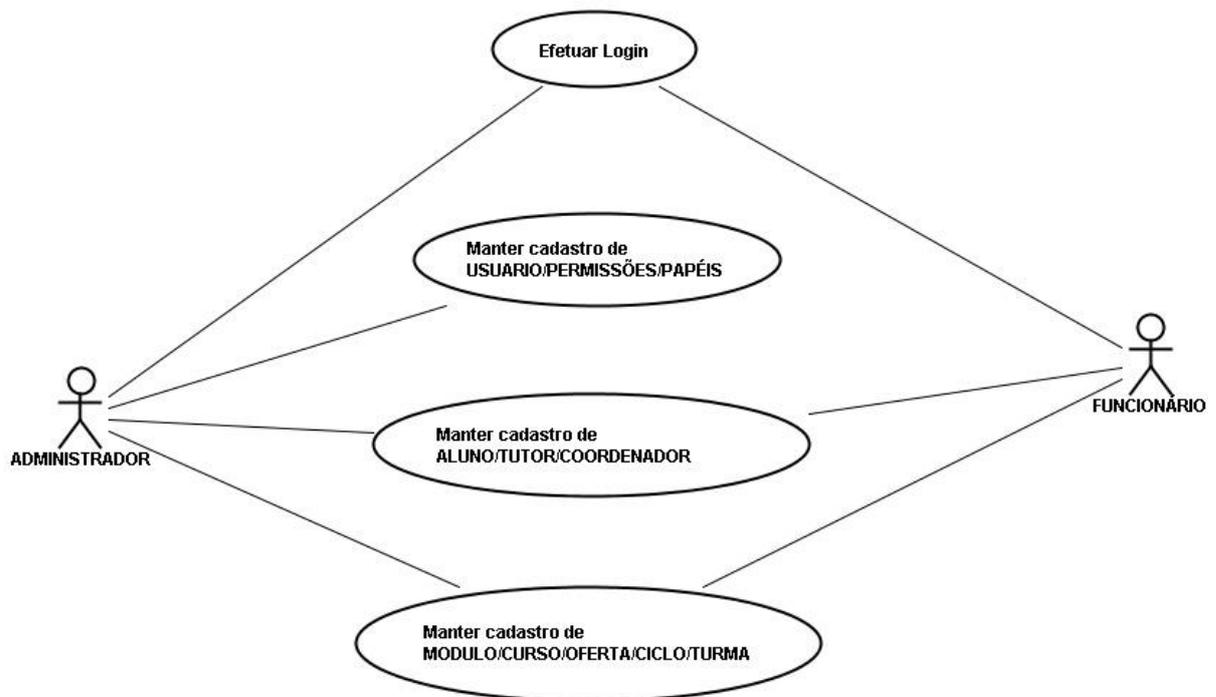


Figura 01: Diagrama de casos de uso.

3.2.3 Modelo Entidade Relacionamento

Com esse modelo é possível visualizar como as tabelas se relacionam dentro do banco de dados. Nele pode ser vistas as chaves primárias identificadoras (PK), as chaves estrangeiras (FK), os relacionamentos entre tabelas e também as interdependências que são criadas pelas tabelas. Com a criação desse modelo fica bem mais simples desenvolver o banco de dados.

A figura 02 exibe o modelo entidade relacional do módulo cadastro, nesse modelo se pode ver como as entidades se relacionam. A entidade aluno se relaciona, pertence a uma turma. A entidade Turma possui um Tutor e pertence a um Ciclo. Já um Ciclo pertence a uma Oferta e que por sua vez está ligada a um Curso.

Cada Curso possui diversos módulos, esses módulos estão ligados a um Ciclo, numa relação de muitos para muitos, ou seja, um módulo pode se relacionar com vários ciclos e um Ciclo pode estar relacionado à vários módulos.

Os módulos estão ligados ao Curso através do relacionamento Ciclo - Oferta - Curso. Já a entidade Tutor está ligada a uma turma e também a entidade Coordenador. Cada Tutor tem um Coordenador. A entidade Coordenador está ligada a uma turma, quando for Coordenador de Turma, e está relacionada com Curso, no caso de ser Coordenador de Curso.

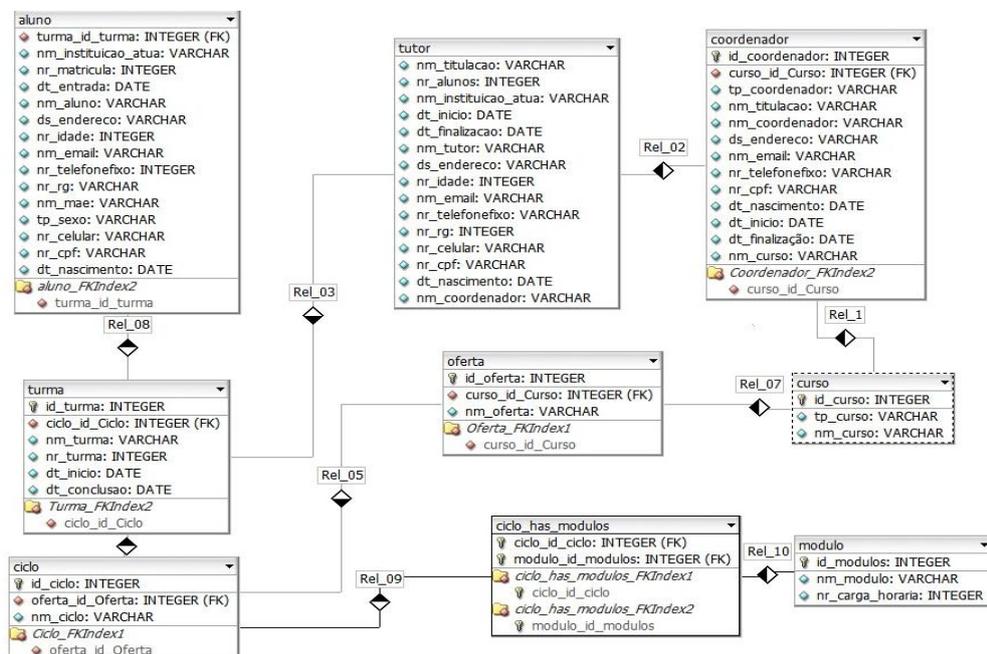


Figura 02: Modelo do módulo Cadastro

A figura 03 mostra o modelo do módulo Administração, nele vemos os relacionamentos das entidades que são responsáveis pelo controle de permissão do sistema.

Um usuário e o seu papel (Função que desempenha no sistema) estão ligados através da entidade Usuário Papel, permitindo dessa forma que um usuário esteja ligado a mais de um papel e que um papel esteja relacionado a mais de um usuário.

A entidade permissão está relacionada com a entidade Papel, definindo dessa forma as tarefas que um determinado papel pode realizar no sistema.

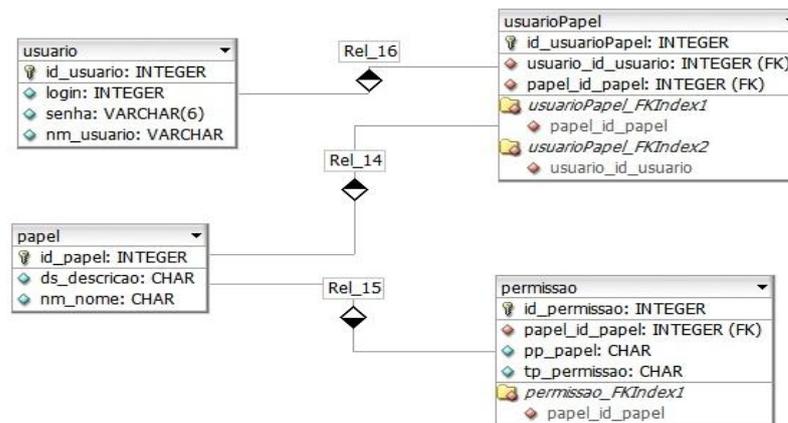


Figura 03: Modelo do módulo Administração

4 APRESENTAÇÃO DO SISTEMA SINEAD

Neste capítulo serão apresentadas algumas telas do Sistema Informatizado da EAD. Cada tela terá suas funções detalhadas para melhor entendimento do funcionamento e utilização do sistema.

Na figura apresentada abaixo, pode-se observar a tela de login. Para ter acesso ao sistema, o usuário deve estar previamente cadastrado no SINEAD.

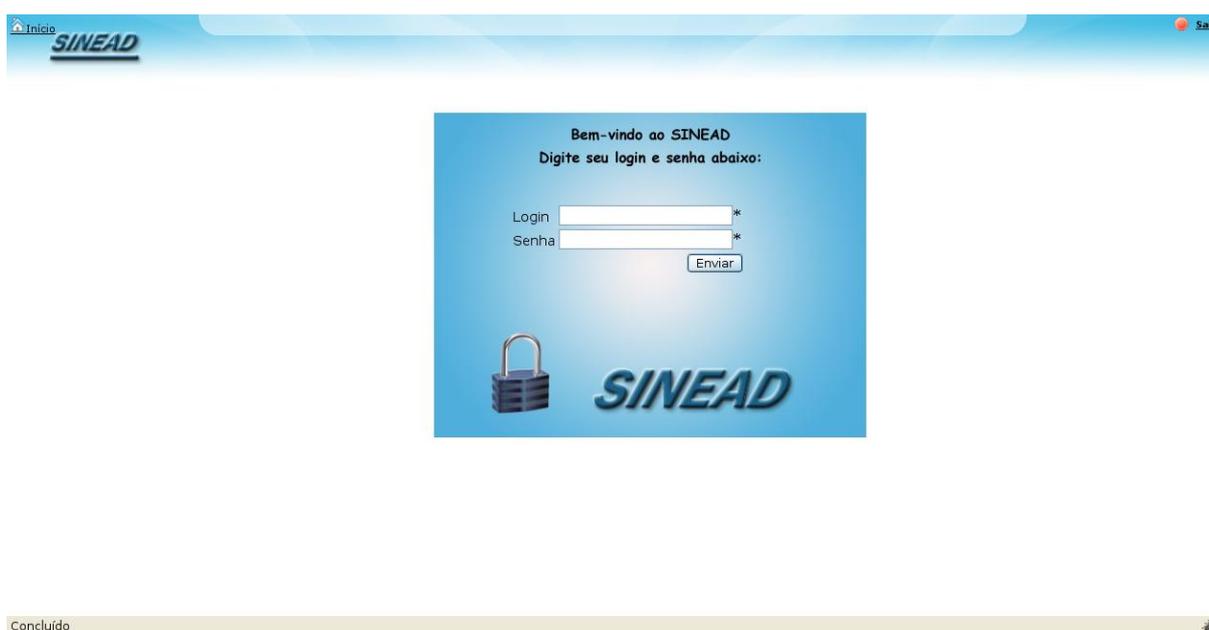


Figura 04: Tela de login.

O SINEAD possui um controle de permissões que possibilita o acesso dos funcionários e do administrador.

O usuário administrador tem acesso a dois módulos, CADASTRO e ADMINISTRAÇÃO, onde ele pode efetuar qualquer tipo de operação nos dados. O usuário do tipo funcionário tem acesso apenas ao módulo CADASTRO, não podendo, dessa forma, alterar permissões do sistema.

Na Figura 05 vê-se a página inicial com o usuário do tipo administrador logado, pode ser visto ainda o link de *Início* que baseado nas regras da Interface Homem Máquina foi criado para que o usuário possa voltar para a página inicial de qualquer lugar que estiver dentro do sistema.

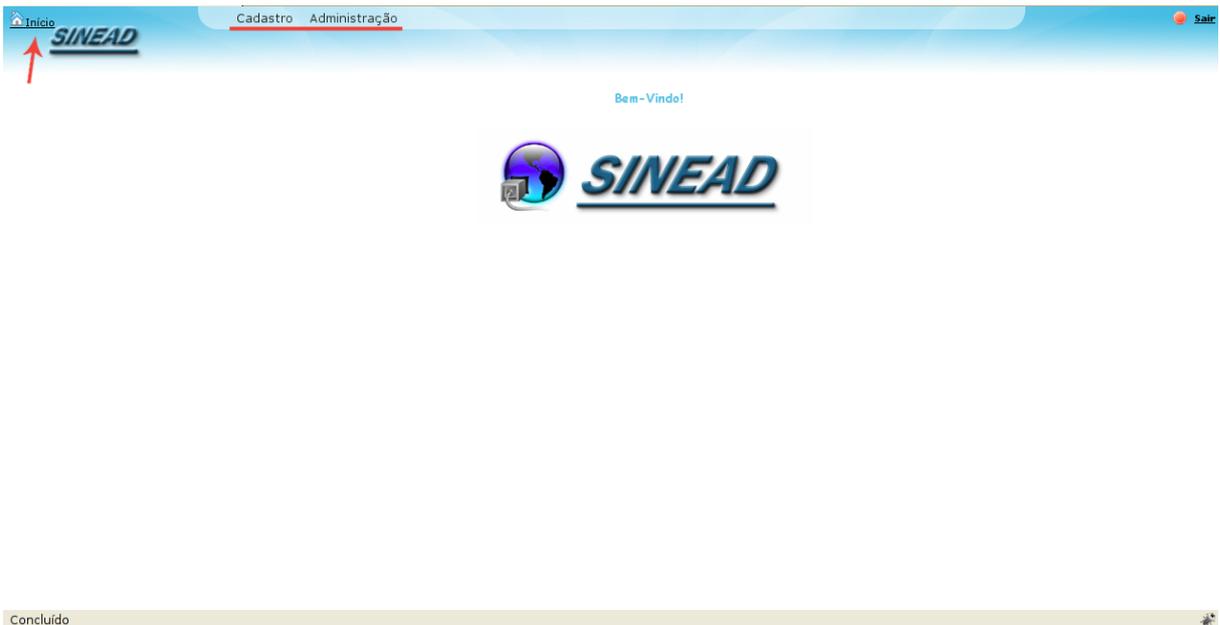


Figura 05: Administrador logado.

A Figura 06 abaixo mostra a tela inicial com o usuário do tipo funcionário logado no sistema. Observa-se que ele tem acesso restrito, podendo apenas ver e utilizar o menu Cadastro.

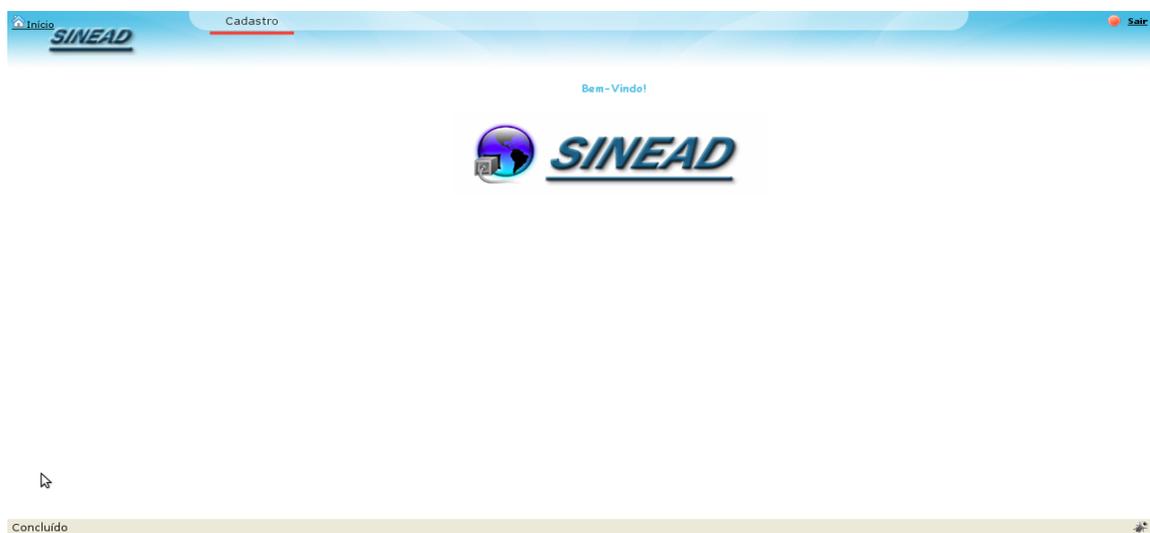


Figura 06: Funcionário logado.

4.1 MENU ADMINISTRAÇÃO

No módulo administração é feito o controle de permissão do sistema. São realizados os cadastros de usuário, papéis e permissão, definindo quem terá acesso ao sistema. Na imagem abaixo pode-se observar o menu Administração que está disponível somente para o administrador.

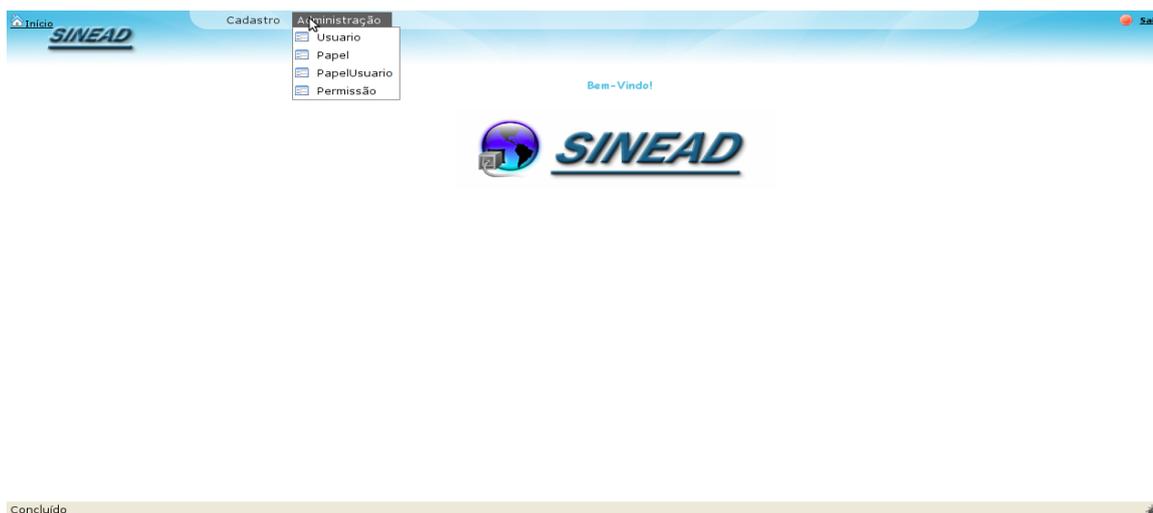


Figura 07: Menu Administração

A segurança de uma aplicação pode ser dividida em duas partes: autenticação e autorização. A autenticação é o processo de identificação do usuário. Geralmente, através de uma tela onde o usuário fornece seu login e senha. Como vimos na figura 04, o SINEAD possui uma tela de login para acessar o sistema.

E a autorização é o processo onde o sistema verifica se determinado usuário possui permissão para acessar alguma funcionalidade do sistema. Para o processo de autorização do SINEAD foi necessário implementar algumas interfaces, são elas:

- User: representa um usuário no sistema.
- Role: representa um papel (ou nível) no sistema.
- Permission: representa a permissão de determinado papel para determinado controller do sistema.

Serão mostradas nos tópicos seguintes, as imagens das telas de cadastros das entidades implementadas para fazer o controle de permissão do sistema.

4.1.2 Submenu Usuário

Utilizando o submenu *Usuário* se pode fazer inclusão, edição e exclusão de usuários. Ao clicar no submenu pode ser visto uma listagem de todos os usuários que foram cadastrados até o momento no sistema. Um usuário possui nome, login e uma senha. Por questão de segurança a senha não aparece na listagem. Para criar um novo usuário, basta clicar no botão *Novo*.

As telas de cadastro de novas entidades serão detalhadas posteriormente, no tópico 4.2.



The screenshot shows the SINEAD system interface. At the top, there is a navigation bar with 'Início', 'Cadastro', and 'Administração' menus, and a 'Sair' button. Below the navigation bar, the 'Usuário' submenu is active. A table lists existing users, and a 'Novo' button is visible for adding a new user. The table has columns for 'Nome', 'Login', and 'Ação'.

Nome	Login	Ação
saulo	saulo	editar excluir
vana	vana	editar excluir
Meire	meire	editar excluir

Página 1

Figura 08: Listagem de Usuários

4.1.3 Submenu Papel

Neste submenu pode-se fazer inclusão, edição e exclusão de papéis. Ao clicar no submenu pode ser visto uma listagem dos papéis disponíveis para o sistema SINEAD. Até o momento, os papéis imprescindíveis para o sistema são: Administrador e Funcionário.

Nome	Descrição	Ação
Administrador	Administrador do Sistema	editar excluir
Funcionario	Funcionario EAD	editar excluir

Página 1

Concluído

Figura 09: Listagem de Papéis

4.1.4 Submenu Usuário Papel

A entidade Usuário Papel foi criada para que o administrador pudesse especificar o tipo de papel que cada usuário tem dentro do sistema.

Um usuário pode desempenhar mais de um papel no sistema. E vários usuários podem desempenhar o mesmo papel.

Usuário	Papel que desenvolve	Ação
saulo	Administrador	editar excluir
vana	Funcionario	editar excluir
Meire	Funcionario	editar excluir

Página 1

Concluído

Figura 10: Listagem Usuário Papel

4.1.5 Submenu Permissão

No submenu *Permissão* se pode fazer inclusão, edição e exclusão de permissões.

Nesta figura, é exibida a lista de permissões cadastradas no sistema. Nesse submenu é possível especificar as permissões que cada Papel tem dentro do sistema.

Papel	Permissão	Tela	Ação
Administrador	update=true;delete=true;read=true;create=true	/sinead/crud/Aluno	editar excluir
Administrador	update=true;delete=true;read=true;create=true	/sinead/crud/Tutor	editar excluir
Administrador	update=true;delete=true;read=true;create=true	/sinead/crud/Coordenador	editar excluir
Administrador	update=true;delete=true;read=true;create=true	/sinead/crud/Turma	editar excluir
Administrador	update=true;delete=true;read=true;create=true	/sinead/crud/Oferta	editar excluir
Administrador	update=true;delete=true;read=true;create=true	/sinead/crud/Curso	editar excluir
Administrador	update=true;delete=true;read=true;create=true	/sinead/crud/Ciclo	editar excluir
Administrador	update=true;delete=true;read=true;create=true	/sinead/crud/CicloModulo	editar excluir
Administrador	update=true;delete=true;read=true;create=true	/sinead/crud/Modulo	editar excluir
Administrador	update=true;delete=true;read=true;create=true	/sinead/crud/Usuario	editar excluir
Administrador	update=true;delete=true;read=true;create=true	/sinead/crud/Papel	editar excluir
Administrador	update=true;delete=true;read=true;create=true	/sinead/crud/UsuarioPapel	editar excluir
Administrador	update=true;delete=true;read=true;create=true	/sinead/crud/Permissao	editar excluir
Funcionario	update=false;delete=false;read=false;create=false	/sinead/crud/Usuario	editar excluir
Funcionario	update=true;delete=true;read=true;create=true	/sinead/crud/Aluno	editar excluir
Funcionario	update=true;delete=true;read=true;create=true	/sinead/crud/Tutor	editar excluir
Funcionario	update=true;delete=true;read=true;create=true	/sinead/crud/Coordenador	editar excluir
Funcionario	update=true;delete=true;read=true;create=true	/sinead/crud/Turma	editar excluir
Funcionario	update=true;delete=true;read=true;create=true	/sinead/crud/Oferta	editar excluir
Funcionario	update=true;delete=true;read=true;create=true	/sinead/crud/Curso	editar excluir
Funcionario	update=true;delete=true;read=true;create=true	/sinead/crud/Ciclo	editar excluir
Funcionario	update=true;delete=true;read=true;create=true	/sinead/crud/CicloModulo	editar excluir

Figura 11: Listagem Permissão

4.2 MENU CADASTRO

No módulo CADASTRO existe a opção de fazer operações com as seguintes entidades: *Aluno*, *Tutor*, *Coordenador*, *Turma*, *Ciclo*, *Oferta*, *Curso*, *Ciclo-Módulo* e *Módulo*.

No submenu *Aluno* pode cadastrar todos os alunos que são filiados ao setor da EAD-UESB. Em *Tutor* são cadastrados todos os tutores que fazem parte do setor. Em *Coordenador* pode ser efetuado o cadastro dos coordenadores, que podem ser Coordenadores de curso, Coordenadores gerais e Coordenadores de outros coordenadores.

Em *Turma* são cadastradas todas as turma que serão disponibilizadas pela EAD-UESB no momento. Cada turma pertence a um Ciclo. Em *Oferta* serão

cadastradas as ofertas do tipo: primeira oferta, segunda oferta e terceira oferta. Cada oferta pertence a um curso. No submenu *Curso* podem ser cadastrados todos os cursos que serão oferecidos pela EAD-UESB.

Em *Ciclo* serão cadastrados os seguintes tipos de ciclo: básico, intermediário e avançado. Para cada ciclo será necessário escolher a que oferta ele pertence.

Em *Módulo* podem ser cadastrados os tipos de módulo disponíveis na EAD-UEAB e em *Ciclo-Módulo* será feita a ligação entre o módulo que já foi cadastrado com um ciclo que foi também cadastrado.

Na Figura 12 em seguida se pode observar o menu *CADASTRO* que está disponível tanto para o Funcionário quanto para Administrador.

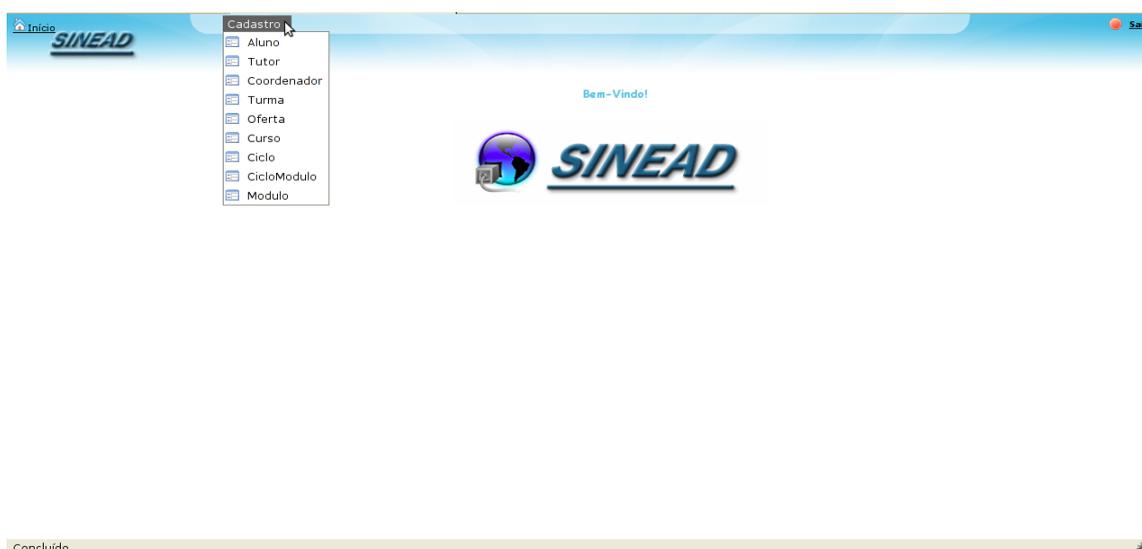


Figura 12: Menu Cadastro

Para demonstrar algumas funcionalidades do sistema foram escolhidos os submenus *Aluno* e *Coordenador*.

4.2.1 Submenu Aluno

Utilizando o submenu *Aluno* o usuário pode fazer inclusão, edição e exclusão de alunos. Ao clicar no submenu pode ser visto uma listagem de todos os alunos que já foram cadastrados no sistema.

Na listagem há a opção de fazer uma pesquisa para localizar dados rapidamente, essa busca, no caso de Aluno pode ser feita por nome, CPF e/ou turma. Para inserir um novo Aluno, basta clicar em Novo.

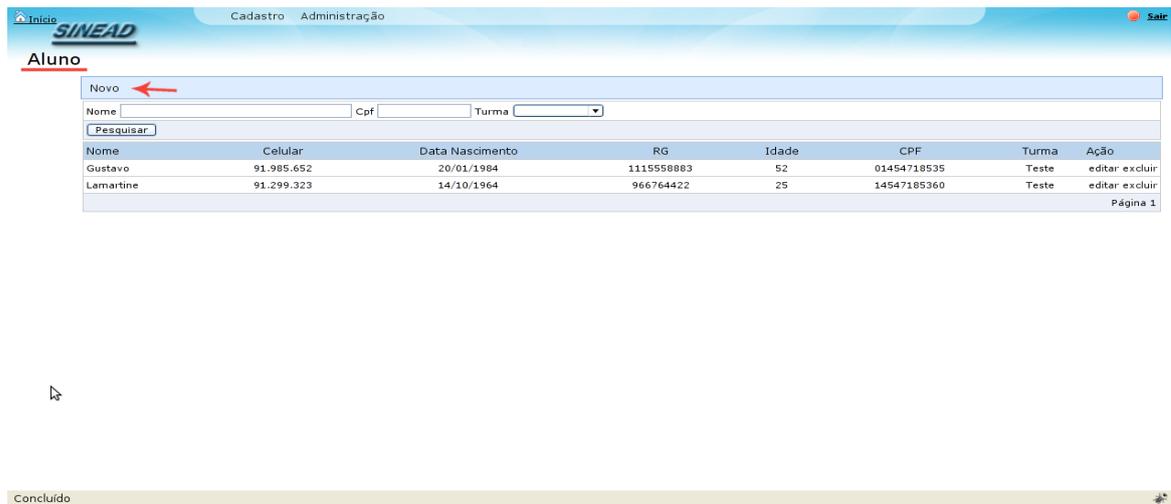


Figura 13: Listagem de Alunos.

Ao clicar no botão *Novo* os dados de um Aluno podem ser inseridos e logo após podem ser salvos. Para voltar para a Listagem, o usuário poderá clicar no botão *Listagem* e o sistema mostrará os alunos cadastrados até o momento.

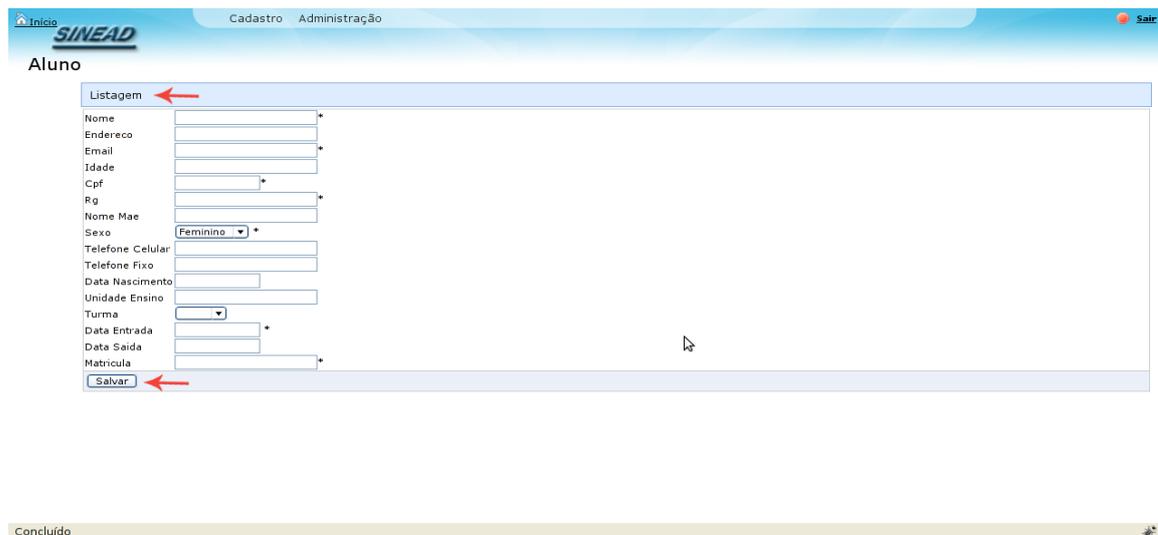


Figura 14: Cadastro de Alunos novos.

4.2.2 Submenu Coordenador

No submenu *Coordenador* é possível fazer inclusão, edição e exclusão de coordenadores. Na imagem abaixo se pode observar o cadastro de um novo coordenador.

The screenshot shows the 'Cadastro' page for a 'Coordenador' in the SINEAD system. The form contains the following fields and values:

- Nome: usuario
- Endereço: Rua União Operária 444
- Email: saulobarrs@gmail.com
- Cpf: 01454718536
- Rg: 966764423
- Data Nascimento: 20/01/1984
- Telefone Celular: 7791353375
- Telefone Fixo: 7734242447
- Data Inicio: 14/10/2009
- Data Finalizacao: 03/10/1986
- Titulacao: Especialização
- Tipo Coordenador: Coordenador de Curso
- Curso: Pedagogia

A 'Salvar' button is located at the bottom of the form.

Figura 15: Cadastro de Coordenadores

Na figura 16, se pode observar a listagem dos coordenadores que estão cadastrados no banco de dados do SINEAD. Nessa listagem é possível efetuar buscas através do nome ou do CPF.

The screenshot shows the 'Listagem' page for 'Coordenadores' in the SINEAD system. It includes a search bar with 'Nome' and 'Cpf' fields and a 'Pesquisar' button. Below is a table with the following data:

Coordenador	Tipo	Curso	E-mail	CPF	RG	Celular	Tel - Fixo	Titulação	Ação
ivanete	Coordenador de Tutor	Pedagogia	edivandasantosp@yahoo.com.br	88888888899	966.764.421			Mestrado	editar excluir
saulo	Coordenador de Curso	Mídias na Educação	saulobarrs@gmail.com	14547185366	966.764.420	7791353375	30832388	Mestrado	editar excluir

The page number 'Página 1' is visible at the bottom right of the table area.

Figura 16: Listagem de Coordenadores

4.3 MENSAGENS DE ERRO

As mensagens de erro em um sistema são muito importantes, pois é por meio delas que pode ser percebida qual falha o usuário cometeu.

Se o usuário estiver cadastrando um Aluno e esquecer-se de um dado que seja obrigatório, ao tentar salvar o sistema automaticamente mostrará uma mensagem informando que ele deve preencher aquele campo.

A figura 17 mostra uma mensagem de erro ao tentar salvar um aluno sem inserir o seu CPF.

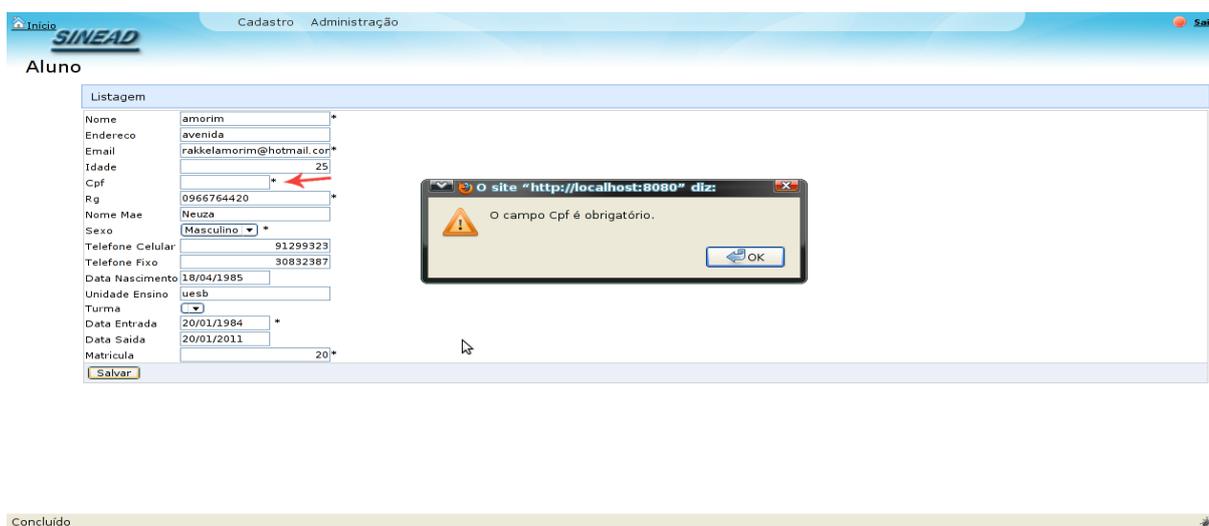


Figura 17: Mensagem de erro: campos obrigatórios.

4.4 MÁSCARA PARA CAMPOS DO FORMULÁRIO

A máscara em um campo é bastante importante, pois, evita que o usuário do sistema tente inserir dados incompatíveis com o formato desejado.

Todos os campos de *Data* possuem uma máscara para evitar que o usuário forneça dados incompatíveis com o formato de uma data. A figura 18 exibe a máscara do campo *Data de Entrada* no cadastro de *Aluno*.

The screenshot shows the SINEAD system interface for student registration. The form is titled 'Aluno' and contains the following fields:

Nome	Lamartine
Endereco	avenida Presidente Dutra
Email	midias3@yahoo.com.br
Idade	25
Cpf	14547185360
Rg	966764422
Nome Mae	Neuze
Sexo	Masculino
Telefone Celular	91299323
Telefone Fixo	91635694
Data Nascimento	14/10/1964
Unidade Ensino	uesb
Turma	Teste
Data Entrada	20/12/
Data Saida	
Matricula	

A red arrow points to the 'Data Entrada' field. A 'Salvar' button is located at the bottom left of the form.

Figura 18: Campos Data com máscara.

4.5 FILTROS DE LISTAGEM

4.5.1 Filtro CPF

Em um sistema que possua muitos dados cadastrados é de grande importância que se tenha filtros que possam retornar respostas rápidas. No submenu *Aluno* o SINEAD possui filtros que podem fazer buscas por CPF, turma e/ou nome.

A próxima figura ilustra a busca de um aluno por CPF. Basta digitar o CPF e clicar em pesquisar, que o aluno com o respectivo CPF será encontrado e aparecerá na listagem.

Novo

Nome Cpf 14547185360 Turma

Nome	Celular	Data Nascimento	RG	Idade	CPF	Turma	Ação
Lamartine	91.299.323	14/10/1964	966764422	25	14547185360	Teste	editar excluir

Página 1

Concluído

Figura 19: Filtro por CPF.

4.5.2 Filtro Turma

Nesse outro caso estamos buscando todos os alunos que pertencem a uma mesma turma. Basta selecionar uma turma e clicar em pesquisar. Logo, aparecerão na Listagem todos os alunos que estão matriculados na turma escolhida.

Novo

Nome Cpf Turma

Nome	Celular	Data Nascimento	RG	Idade	CPF	Turma	Ação
Gustavo	91.985.652	20/01/1984	1115558883	52	01454718535	Teste	editar excluir
Lamartine	91.299.323	14/10/1964	966764422	25	14547185360	Teste	editar excluir

Página 1

Concluído

Figura 20: Filtro por Turma.

5 CONCLUSÃO E TRABALHOS FUTUROS

Esse trabalho apresentou o desenvolvimento de um sistema Web para gerenciar o setor de Educação à Distância da UESB.

Com esse sistema o usuário terá maior facilidade para organizar e gerenciar todos os dados relacionados ao setor da EAD-UESB. O sistema permite o cadastro de dados relacionados a alunos, tutores, professores, coordenadores e ainda outros dados relacionados à oferta de curso, ciclo do curso, módulos e turmas, o que o torna bastante completo para o gerenciamento de dados do setor.

Alguns desafios foram encontrados durante a produção do sistema, a maioria relacionada com a melhor maneira de produzir as dependências entre tabelas do banco de dados, entretanto os desafios foram vencidos com ajuda de pessoas que detinham um conhecimento mais abrangente da área, livros sobre o assunto e professores empenhados em ajudar.

O conhecimento adquirido durante a produção do sistema será levado por toda a vida e possivelmente utilizado em projetos futuros que pretendo desenvolver.

Para o desenvolvimento de trabalhos futuros pretende-se adicionar novas funcionalidades ao sistema produzido, como por exemplo, a geração de relatórios em PDF que ajudaria a manter o controle dos dados inseridos no sistema, alguns desses relatórios poderiam ser de alunos por turma, lista dos alunos reprovados e aprovados, disciplinas oferecidas, tutores e suas turmas, lista de coordenadores agrupados por tipo, entre outros.

6 REFERÊNCIAS

AKITA, Fábio. **Repensando a WEB com Rails**. Rio de Janeiro: Brasport, 2006.

AMBLER, Scott W.. **Modelagem Ágil: Práticas eficazes para programação extrema e o processo unificado**. São Paulo: Bookman, 2004.

ANDRADE, Antonio. **Usabilidade de Interfaces WEB**. Rio de Janeiro: E-papers, 2007.

BECK, Kent; FOWLER, Martin. **Planning extreme programming**. Estados Unidos da América: 2000.

FOWLER, Martin. **Uml essencial: Um breve guia para a linguagem-padrão de modelagem de objetos**. Porto Alegre: Campus, 2004.

GARCIA, Rógel. **Next framework**. São Paulo: 2010. Disponível em: <<http://www.nextframework.org>>. Acesso em: 25 jul. 2010.

LARMAN, Craig. **Utilizando UML e Padrões: Uma introdução à análise e ao projeto orientado a objetos e ao desenvolvimento iterativo**. São Paulo: Bookman, 2005.

LEITE, Mário. **Técnicas de Programação: Uma Abordagem Moderna**. São Paulo: Brasport, 2006.

LITTO, Fredric; FORMIGA, Marcos. **Educação a distância: O Estado da Arte**. São Paulo: Pearson, 2009.

RAMOS, R. Argenton. **Treinamento prático em UML**. São Paulo: Digerati, 2002.

APÊNDICE 1 - DESCRIÇÃO DO DIAGRAMA DE CASOS DE USO

CASO DE USO 01 – Efetuar Login

Sumário: O usuário efetua login no sistema para usar suas funcionalidades.

Ator primário: Usuário administrador ou funcionário

Pré-condições: Estar cadastrado no sistema

Fluxo Principal:

1. O usuário digita o seu login e senha previamente cadastrados no sistema.
2. O sistema verifica se os dados digitados estão corretos.
3. Caso os dados estejam corretos, o sistema mostra a página inicial ao usuário.
4. Caso os dados não estejam corretos, uma mensagem é mostrada ao usuário:
Login/Senha inválidos.

Pós-condições: O usuário tem acesso ao sistema.

CASO DE USO 02 - Manter cadastro (Incluir, Consultar, Alterar e Apagar) sobre as entidades Usuários, Permissões e Papéis

Sumário: O administrador usa o sistema para realizar a manutenção (CRUD) dos dados sobre as entidades.

Ator primário: Administrador do sistema

Pré-condições: O usuário está identificado pelo sistema e ser administrador

Fluxo Principal:

1. O administrador requisita a manutenção das entidades
2. O sistema apresenta todas as entidades
3. O administrador seleciona a entidade que deseja manter
4. O sistema exibe informações da entidade escolhida e apresenta as operações que podem ser realizadas: a inclusão de uma nova entidade, a alteração dos dados de uma entidade e a exclusão de uma entidade
5. O administrador indica a opção a realizar ou opta por finalizar o caso de uso
6. O administrador seleciona a opção desejada: Inclusão, alteração ou exclusão
7. Se o administrador deseja continuar com a manutenção, o caso de uso retorna ao passo 4; caso contrário, o caso de uso termina.

Fluxo Alternativo (6): Inclusão

- O administrador requisita a inclusão de uma entidade
- O sistema apresenta um formulário em branco para que os dados de uma entidade sejam incluídos
- O administrador fornece os dados da nova entidade
- O sistema verifica a validade dos dados. Se os dados forem válidos, inclui a nova entidade; caso contrário, o sistema reporta o fato, solicita novos dados e repete a verificação.

Fluxo Alternativo (6): Exclusão

O administrador seleciona uma entidade e requisita ao sistema que a exclua

O sistema pede ao administrador confirmação da exclusão

Se o administrador confirma a exclusão, o sistema exclui a entidade; caso contrário, o sistema não exclui.

Fluxo Alternativo (6): Alteração

- 3 O administrador altera um ou mais dados sobre a entidade e requisita sua atualização
- 4 O sistema verifica a validade dos dados e, se eles forem válidos, altera os dados da entidade; caso contrário, o sistema reporta o fato, solicita novos dados e repete a verificação.

Pós-condições: Uma entidade foi inserida ou removida, ou seus detalhes foram exibidos/alterados.

CASO DE USO 03 - Manter cadastro (Incluir, Consultar, Alterar e Apagar) sobre as entidades Aluno, Tutor e Coordenado

Sumário: O usuário usa o sistema para realizar a manutenção (CRUD) dos dados sobre as entidades.

Ator primário: Usuário do sistema, administrador ou funcionário

Pré-condições: O usuário está identificado pelo sistema

Fluxo Principal:

1. O usuário requisita a manutenção das entidades
2. O sistema apresenta todas as entidades
3. O usuário seleciona a entidade que deseja manter
4. O sistema exibe informações da entidade escolhida e apresenta as operações que podem ser realizadas: a inclusão de uma nova entidade, a alteração dos dados de uma entidade e a exclusão de uma entidade
5. O usuário indica a opção a realizar ou opta por finalizar o caso de uso
6. O usuário seleciona a opção desejada: Inclusão, alteração ou exclusão
7. Se o usuário deseja continuar com a manutenção, o caso de uso retorna ao passo 4; caso contrário, o caso de uso termina.

Fluxo Alternativo (6): Inclusão

- O usuário requisita a inclusão de uma entidade
- O sistema apresenta um formulário em branco para que os dados de uma entidade sejam incluídos
- O usuário fornece os dados da nova entidade
- O sistema verifica a validade dos dados. Se os dados forem válidos, inclui a nova entidade; caso contrário, o sistema reporta o fato, solicita novos dados e repete a verificação.

Fluxo Alternativo (6): Exclusão

- O usuário seleciona uma entidade e requisita ao sistema que a exclua
- O sistema pede ao administrador confirmação da exclusão

Se o usuário confirma a exclusão, o sistema exclui a entidade; caso contrário, o sistema não exclui.

Fluxo Alternativo (6): Alteração

- 5 O usuário altera um ou mais dados sobre a entidade e requisita sua atualização
- 6 O sistema verifica a validade dos dados e, se eles forem válidos, altera os dados da entidade; caso contrário, o sistema reporta o fato, solicita novos dados e repete a verificação.

Pós-condições: Uma entidade foi inserida ou removida, ou seus detalhes foram exibidos/alterados.

CASO DE USO 04 - Manter cadastro (Incluir, Consultar, Alterar e Apagar) sobre as entidades Módulo, Curso, Oferta, Ciclo e Curso

Sumário: O usuário usa o sistema para realizar a manutenção (CRUD) dos dados sobre as entidades.

Ator primário: Usuário do sistema, administrador ou funcionário

Pré-condições: O usuário está identificado pelo sistema

Fluxo Principal:

1. O usuário requisita a manutenção das entidades
2. O sistema apresenta todas as entidades
3. O usuário seleciona a entidade que deseja manter
4. O sistema exibe informações da entidade escolhida e apresenta as operações que podem ser realizadas: a inclusão de uma nova entidade, a alteração dos dados de uma entidade e a exclusão de uma entidade
5. O usuário indica a opção a realizar ou opta por finalizar o caso de uso
6. O usuário seleciona a opção desejada: Inclusão, alteração ou exclusão
7. Se o usuário deseja continuar com a manutenção, o caso de uso retorna ao passo 4; caso contrário, o caso de uso termina.

Fluxo Alternativo (6): Inclusão

- O usuário requisita a inclusão de uma entidade
- O sistema apresenta um formulário em branco para que os dados de uma entidade sejam incluídos
- O usuário fornece os dados da nova entidade
- O sistema verifica a validade dos dados. Se os dados forem válidos, inclui a nova entidade; caso contrário, o sistema reporta o fato, solicita novos dados e repete a verificação.

Fluxo Alternativo (6): Exclusão

O usuário seleciona uma entidade e requisita ao sistema que a exclua
O sistema pede ao administrador confirmação da exclusão
Se o usuário confirma a exclusão, o sistema exclui a entidade; caso contrário, o sistema não exclui.

Fluxo Alternativo (6): Alteração

- 7 O usuário altera um ou mais dados sobre a entidade e requisita sua atualização
- 8 O sistema verifica a validade dos dados e, se eles forem válidos, altera os dados da entidade; caso contrário, o sistema reporta o fato, solicita novos dados e repete a verificação.

Pós-condições: Uma entidade foi inserida ou removida, ou seus detalhes foram exibidos/alterados.