



# **SISTEMA DE CONTROLE DE ALUNOS EM ESTÁGIO CURRICULAR**

**Trabalho de Conclusão de Curso**

**Engenharia da Computação**

**Anthony Cavalcanti de Cerqueira**

**Orientador: Prof. Sérgio Castelo Branco Soares**

**Recife, junho de 2008**



**UNIVERSIDADE  
DE PERNAMBUCO**



# SISTEMA DE CONTROLE DE ALUNOS EM ESTÁGIO CURRICULAR

**Trabalho de Conclusão de Curso**

**Engenharia da Computação**

Este Projeto é apresentado como requisito parcial para obtenção do diploma de Bacharel em Engenharia da Computação pela Escola Politécnica de Pernambuco – Universidade de Pernambuco.

**Anthony Cavalcanti de Cerqueira**

**Orientador: Prof. Sérgio Castelo Branco Soares**

**Recife, junho de 2008**



UNIVERSIDADE  
DE PERNAMBUCO

**Anthony Cavalcanti de Cerqueira**

# **SISTEMA DE CONTROLE DE ALUNOS EM ESTÁGIO CURRICULAR**

## Resumo

Atualmente, o controle de alunos em Estágio Curricular feito pela Coordenação de Estágio do Departamento de Sistemas e Computação (DSC) da Escola Politécnica de Pernambuco (POLI) apresenta diversas dificuldades, como a falta de organização de documentos produzidos de forma manual, alunos que negligenciam a disciplina ao não comparecerem às reuniões de acompanhamento e estagiários trabalhando em atividades não recomendadas pelo departamento. Este projeto propõe o desenvolvimento de uma aplicação *web* de gerenciamento de Estágio Curricular dos alunos, tornando mais fácil o trabalho do Coordenador de Estágio, evitando matriculas em estágios de baixa qualidade e aumentando o controle sobre os estagiários e a participação dos mesmos em relação à disciplina. Exemplos de tecnologias utilizadas neste trabalho são C#, ASP.NET e SQL Server.



# Abstract

*Nowadays, the control of students curricular stage in the Department of Computing and Systems (DSC) of the Polytechnic School of Pernambuco (POLI) show many difficulties, such as lack of organization of manual producing documents, students that neglect the associated course as they miss meetings, and working in companies not recommended by the Department. This project proposes the development of a web application for controlling students in curricular stage, aiming at making easier the stage coordinator's work, avoiding matriculation in low quality stages and increasing the control of students and their participation in the discipline. Examples of used technologies are C#, ASP.NET and SQL Server.*

# Sumário

|   |           |
|---|-----------|
| <b>1. Introdução</b>                                | <b>11</b> |
| 1.1. Motivação                                      | 11        |
| 1.2. Objetivos e Metas                              | 11        |
| <b>2. Estado da Arte</b>                            | <b>13</b> |
| 2.1. Tecnologias                                    | 13        |
| 2.2. Processos                                      | 16        |
| 2.3. Justificativa da configuração utilizada        | 17        |
| 2.4. Conclusão                                      | 17        |
| <b>3. O Trabalho</b>                                | <b>18</b> |
| 3.1. O Sistema                                      | 18        |
| 3.2. Arquitetura                                    | 27        |
| 3.3. Metodologia                                    | 27        |
| 3.4. Lições aprendidas no desenvolvimento           | 29        |
| 3.5. Recomendações                                  | 29        |
| 3.6. Conclusão                                      | 29        |
| <b>4. Conclusões e Trabalhos Futuros</b>            | <b>30</b> |
| 4.1. Contribuições                                  | 30        |
| 4.2. Trabalhos Futuros                              | 30        |
| <b>5. Bibliografia</b>                              | <b>32</b> |
| <b>6. Apêndice A: Caso de Negócio</b>               | <b>33</b> |
| <b>7. Apêndice B: Análise de Riscos</b>             | <b>36</b> |
| <b>8. Apêndice C: Análise de Casos de uso</b>       | <b>41</b> |
| <b>9. Apêndice D: Especificação de Casos de uso</b> | <b>49</b> |
| <b>10. Apêndice E: Diagramas</b>                    | <b>60</b> |

# Índice de Figuras

|  |    |
|--|----|
| <b>Figura 1.</b> Visão geral do .NET Framework.                                  | 14 |
| <b>Figura 2.</b> Página de login do sistema.                                     | 18 |
| <b>Figura 3.</b> Página principal do sistema, vista pelo Coordenador de Estágio. | 19 |
| <b>Figura 4.</b> Alteração de dados de aluno.                                    | 20 |
| <b>Figura 5.</b> Inclusão de Estágio Curricular.                                 | 21 |
| <b>Figura 6.</b> Consulta de Estágios Curriculares.                              | 22 |
| <b>Figura 7.</b> Visualização de Estágio Curricular por um aluno.                | 23 |
| <b>Figura 8.</b> Marcação de reuniões de acompanhamento.                         | 23 |
| <b>Figura 9.</b> Exemplo de relatório administrativo.                            | 24 |
| <b>Figura 10.</b> Fórum.   | 24 |
| <b>Figura 11.</b> Inclusão de depoimento no fórum.                               | 25 |
| <b>Figura 12.</b> Envio de Relatório de Estágio.                                 | 26 |
| <b>Figura 13.</b> Arquitetura do sistema.  | 27 |
| <b>Figura 14.</b> Diagrama de Classes.   | 60 |
| <b>Figura 15.</b> Diagrama de Casos de Uso de Aluno.                             | 61 |
| <b>Figura 16.</b> Diagrama de Casos de Uso de Professor.                         | 61 |
| <b>Figura 17.</b> Diagrama de Casos de Uso de Coordenador.                       | 61 |
| <b>Figura 18.</b> Diagrama do Banco de Dados.                                    | 62 |

# Índice de Tabelas

|  |    |
|--|----|
| <b>Tabela 1.</b> Linguagens suportadas pelo .NET Framework | 13 |
| <b>Tabela 2.</b> Cronograma das Atividades                 | 28 |
| <b>Tabela 3.</b> Cronograma Inicial                        | 34 |
| <b>Tabela 4.</b> Descrição da classe Usuário               | 41 |
| <b>Tabela 5.</b> Descrição da classe DAOUsuario            | 41 |
| <b>Tabela 6.</b> Descrição da classe Acadêmico             | 42 |
| <b>Tabela 7.</b> Descrição da classe DAOAluno              | 42 |
| <b>Tabela 8.</b> Descrição da classe Empresa               | 43 |
| <b>Tabela 9.</b> Descrição da classe DAOEmpresa            | 43 |
| <b>Tabela 10.</b> Descrição da classe DAOProfessor         | 44 |
| <b>Tabela 11.</b> Descrição da classe Estágio              | 45 |
| <b>Tabela 12.</b> Descrição da classe DAOEstagio           | 45 |
| <b>Tabela 13.</b> Descrição da classe Depoimento           | 46 |
| <b>Tabela 14.</b> Descrição da classe Assunto              | 47 |
| <b>Tabela 15.</b> Descrição da classe DAODEpoimento        | 47 |
| <b>Tabela 16.</b> Descrição da classe DAOAssunto           | 47 |



# Tabelas de Siglas

As siglas estão dispostas por ordem de aparição no texto.

| <b>Sigla</b> | <b>Significado</b>                         |
|--------------|--|
| DSC          | Departamento de Sistemas e Computação      |
| POLI         | Escola Politécnica de Pernambuco           |
| CLR          | <i>Common Language Runtime</i>             |
| IL           | <i>Intermediate Language</i>               |
| MSIL         | <i>Microsoft Intermediate Language</i>     |
| JI           | <i>Just-In-Time</i>                        |
| CTS          | <i>Common Type System</i>                  |
| CLS          | <i>Common Language Specification</i>       |
| VB           | <i>Visual Basic</i>                        |
| ASP          | <i>Active Server Pages</i>                 |
| HTML         | <i>HyperText Markup Language</i>           |
| SQL          | <i>Structured Query Language</i>           |
| SGBD         | Sistema de Gerenciamento de Banco de Dados |
| RAM          | <i>Random Access Memory</i>                |
| HD           | Hard Disk                                  |
| IDE          | <i>Integrated Drive Electronics</i>        |
| GB           | Gigabyte                                   |
| MB           | Megabyte                                   |
| GHz          | Gigahertz                                  |
| PDF          | <i>Portable Document Format</i>            |
| FAQ          | <i>Frequently Asked Questions</i>          |
| IIS          | <i>Internet Information Services</i>       |
| FTP          | <i>File Transfer Protocol</i>              |
| LINQ         | <i>Language Integrate Query</i>            |
| WPF          | <i>Windows Presentation Foundation</i>     |
| XML          | <i>Extensible Markup Language</i>          |
| DAO          | <i>Data Access Object</i>                  |



# Agradecimentos

À Universidade de Pernambuco, por me dar a oportunidade de cursar uma universidade pública de qualidade. Ao professor Sérgio Soares Castelo Branco, pelas orientações necessárias para a confecção deste trabalho. A meus pais, por me darem a educação que se tornou a minha arma para enfrentar a vida. A minha esposa e minha filha, por me darem todo o apoio que eu pudesse precisar. A Deus, por me dar a força e a determinação necessárias para alcançar meus objetivos.

# Capítulo 1

## Introdução

### 1.1 – Motivação

Atualmente, todo o controle de alunos em Estágio Curricular do Departamento de Sistemas e Computação (DSC) da Escola Politécnica de Pernambuco (POLI) é feito de forma manual pela Coordenação de Estágio, através do preenchimento de documentos. Esta prática acarreta em acúmulo excessivo de documentos, duplicação de informações e dificuldades em se administrar os estágios em curso. Além disso, todas estas informações e tarefas ficam centralizadas no Coordenador de Estágio do DSC, o que tende a causar a sobrecarga deste profissional.

Como prática geral, os alunos tendem a se matricular primeiro no Estágio Curricular através da Divisão de Estágio da POLI, usando o SIG@ – o sistema de gestão acadêmica da instituição –, e só então procuram um professor orientador do DSC. O que ocorre é que muitas vezes o estágio em questão não é recomendado pelo departamento, pois o aluno servirá de mão-de-obra barata e não terá nenhuma experiência enriquecedora para a formação profissional.

Uma vez iniciado o Estágio Curricular, surge outro problema. Os alunos tendem a aparecerem para a primeira reunião de acompanhamento com o professor orientador e, ao invés de manterem uma regularidade nestas reuniões, passam o resto do semestre sem entrar em contato com o docente, reaparecendo apenas pouco antes da entrega do relatório do Estágio Curricular. Surge aí a dificuldade de controle e obtenção de informações sobre o status atual dos discentes.

### 1.2 – Objetivos e Metas

O objetivo geral deste projeto é o desenvolvimento de uma aplicação *web* de gerenciamento de Estágio Curricular dos alunos, tornando mais fácil o trabalho do Coordenador de Estágio, evitando matrículas em estágios de baixa qualidade e aumentando o controle sobre os estagiários e a participação dos mesmos em relação à disciplina.

No referido sistema, o Coordenador de Estágio tem acesso pleno a informações de alunos, empresas e professores orientadores. É exigido o cadastramento prévio do aluno no sistema antes que o mesmo se matricule de fato no Estágio Curricular, a fim de que o Coordenador possa fazer uma avaliação da atividade e decidir se a mesma é recomendada ou não como experiência profissional e acadêmica.

Uma vez iniciado o estágio, o professor orientador pode fazer a marcação de reuniões de acompanhamento e registrar o comparecimento do aluno às mesmas, a fim de evitar que o aluno se despenda da disciplina e deixe de fornecer informações ao Departamento da sua atividade.

O aluno, durante o exercício do seu estágio, pode acompanhar a marcação de reuniões de acompanhamento feitas por seu professor orientador e postar em um fórum depoimentos sobre exercício da sua função. Ao fim do estágio, deve enviar seu Relatório de Estágio para livre consulta por outros usuários.

Ao final desse projeto, deseja-se:

- ✓ Permitir o gerenciamento de forma automatizada e mais eficiente dos alunos em Estágio Curricular.
- ✓ Evitar a sobrecarga de serviços sobre o Coordenador de Estágio.
- ✓ Prover uma análise prévia a respeito da qualidade do estágio proposto.
- ✓ Proporcionar a troca de experiências profissionais entre alunos através do fórum de depoimentos.
- ✓ Disponibilizar uma vasta biblioteca de Relatórios de Estágio.
- ✓ Melhorar o controle em relação às reuniões de acompanhamento entre alunos e professores orientadores.
- ✓ Manter constante a obtenção de informações sobre o status atual dos estágios.
- ✓ Aumentar a participação dos alunos em relação ao Estágio Curricular, uma vez que os mesmos poderão acompanhar sua situação através do sistema.

# Capítulo 2

## Estado da Arte

### 2.1 – Tecnologias Utilizadas

#### 2.1.1 – A Plataforma .NET

O Microsoft .NET<sup>7</sup> é uma iniciativa da Microsoft em que visa uma plataforma única para desenvolvimento e execução de sistemas e aplicações. Todo e qualquer código gerado para .NET pode ser executado em qualquer dispositivo ou plataforma que possua o .NET Framework instalado. Com idéia semelhante à plataforma Java, o programador deixa de escrever código para um sistema ou dispositivo específico e passa a escrever para uma plataforma.

A plataforma .NET que tem o objetivo de facilitar a vida do programador: ao programar em uma linguagem que o *framework* tem suporte (vide **Tabela 2.1**), a plataforma faz com que o código escrito seja rapidamente traduzido em qualquer uma das outras. Assim, um programador pode desenvolver um programa em VB.NET utilizando classes implementadas em C++, por exemplo. Além disso, o *framework* comunica-se diretamente com o sistema operacional, abstraindo operações complexas, como manipulação de ponteiros e rotinas em Assembly.

Desta forma, A plataforma .NET faz uma integração considerável entre linguagens e sistemas, diminui custos em treinamento de pessoal e agilizada a implantação de sistemas.

**Tabela 1:** Linguagens suportadas pelo .NET Framework.

|             |                   |                 |
|-------------|-------------------|-----------------|
| APL         | Pascal            | RPG             |
| Oberon      | Component Pascal  | Fortran         |
| C#          | Perl              | Scheme          |
| Oz          | Curriculum        | Haskell         |
| COBOL       | Eiffel            | SmallTalk       |
| J#          | JScript .NET      | Mercury         |
| Standard ML | Visual Basic .NET | Visual C++ .NET |

A plataforma .NET é composta por dois componentes: um ambiente de execução – a CLR –, uma grande biblioteca de classes – a *Framework Class Library*.

A CLR (*Common Language Runtime*) é o ambiente de execução para todos os programas no .NET Framework. Ela age como um sistema operacional, ou seja, tudo que é executado dentro dela é gerenciado por ela mesma, como requisições de acesso à memória e de *hardware*. Ele encarrega-se de aprimorar o desempenho de aplicativos, minimizando processamento desnecessário e realizando *garbage collection* (destruição de objetos inúteis) e tratamento de exceções.



**Figura 1.** Visão geral do .NET Framework.

A CLR interpreta e executa dinamicamente código na IL (*Intermediate Language*), também conhecida como MSIL (*Microsoft Intermediate Language*). O compilador de cada linguagem do *framework* compila seu código para a IL. Em tempo de execução, o compilador JIT (*Just-In-Time*) da CLR compila o código em IL para código nativo, o qual é interpretado e executado. Desta forma, a CLR é acessível a qualquer compilador que produza código em IL.

Fazem parte ainda da CLR a CTS (*Common Type System*), que descreve o conjunto de tipos suportados pela mesma, e a CLS (*Common Language Specification*), que determina a forma como as linguagens do *framework* compartilham suas bibliotecas e tem o propósito de prover a interoperabilidade entre as linguagens do .NET.

## 2.1.2 – ASP.NET

ASP.NET<sup>8</sup> é a plataforma da Microsoft para o desenvolvimento de aplicações *Web* e é o sucessor da tecnologia ASP. É um componente do servidor *Web* IIS que permite criar páginas dinâmicas através de uma linguagem de programação integrada no .NET Framework. Não é nem uma linguagem de programação, nem um servidor *Web*.

O ASP.NET é baseado no .NET Framework herdando todas as suas características. Por isso, como qualquer aplicação .NET, as aplicações para essa plataforma podem ser escritas em várias linguagens, como C# e Visual Basic .NET. Uma aplicação para *Web* desenvolvida em ASP.NET pode reutilizar código de qualquer outro projeto escrito para a plataforma .NET, mesmo que em linguagem diferente. Uma página ASP.NET escrita em VB.NET pode chamar componentes escritos em C, por exemplo.

O ASP.NET é a evolução do ASP<sup>11</sup>, tecnologia usada anteriormente pela Microsoft para criar aplicações *Web*. Algumas das suas vantagens em relação ao ASP são:

1. No ASP, quando uma página é solicitada no servidor, a biblioteca do ASP lê cada uma das linhas passo a passo, interpreta os códigos, monta um HTML virtual e envia para o browser que o solicitou. Já no ASP.NET, as páginas são compiladas; por isso, são cerca de 25% mais rápidas que no ASP.
2. No VBScript, a linguagem de programação usada no ASP, a declaração de uma variável não é obrigatória. Além disso, as variáveis não podem ser tipadas (ou seja, ter seu tipo de

conteúdo definido); são do tipo *Variant* – podem receber qualquer tipo de conteúdo. No ASP.NET, todas as variáveis devem ser declaradas e ter o respectivo tipo definido, o que é fundamental no tratamento do código.

3. Nas páginas em ASP, o código VBScript é inserido dentro do código HTML, retalhando toda a página e dificultando sua manutenção. Em ASP.NET, usa-se o *Code Behind*: uma página é dividida em dois arquivos vinculados, um de layout, escrito em HTML, e outro com o código de programação.
4. O ASP.NET disponibiliza uma vasta biblioteca de controles *Web* pré-desenvolvidos, como tabelas, caixas de seleção e calendários, o que facilita a criação das páginas, principalmente quando é preciso publicar o conteúdo de um banco de dados. Além disso, existem os controles de validação, usados para validar os dados inseridos pelo usuário.
5. Em ASP, a melhor forma de *debugar* erros é imprimindo o conteúdo de variáveis na tela do *browser*. No ASP.NET, utilizando a ferramenta de desenvolvimento adequada (Visual Studio .NET ou Visual Web Developer), é possível definir *breakpoints* (pontos de interrupção) no código, provocando uma pausa temporária durante a execução da página no *browser*. Assim, pode-se checar o conteúdo das variáveis e acompanhar o processamento passo a passo.
6. No ASP.NET, o tratamento de erros, bastante deficiente no ASP, foi totalmente reestruturado com o uso de exceções, sendo possível identificar qualquer erro que ocorra na página e personalizar mensagens.
7. Normalmente, em ASP.NET, é usada a programação orientada a eventos, que é bem mais intuitiva e muito similar à usada em aplicações locais.

### 2.1.3 – Visual C# .NET

A linguagem C#<sup>8</sup> foi criada junto com a arquitetura .NET. Embora existam várias outras linguagens que suportam essa tecnologia (como VB.NET, C++ e J#), C# é considerada a linguagem símbolo do .NET pelas seguintes razões:

- ✓ Foi criada praticamente do zero para funcionar na nova plataforma, sem preocupações de compatibilidade com código de legado.
- ✓ O compilador C# foi o primeiro a ser desenvolvido.
- ✓ A maior parte das classes do .NET Framework foram desenvolvidas em C#.
- ✓ Praticamente todo desenvolvimento feito na Microsoft hoje é em C#.

A C# (pronuncia-se "cí charp") é, de certa forma, a linguagem de programação que mais diretamente reflete a plataforma .NET, pois está de tal forma ligado a ela que possui o conceito de código não-gerenciado. Suas estruturas de dados primitivas são objetos que correspondem a tipos em .NET. A desalocação automática de memória por *garbage collection*, além de várias de suas abstrações, tais como classes, interfaces, delegados e exceções, são nada mais que a exposição explícita recursos do ambiente .NET.

A C# é baseada nas linguagens C e C++, porém apresenta melhorias em relação a elas, como por exemplo:

- ✓ Ponteiros e aritmética sem checagem contra sobrecarga (*overflow*) só podem ser utilizados em uma modalidade especial chamada “modo inseguro” (*unsafe mode*).
- ✓ Objetos não são liberados explicitamente, mas através de *garbage collection*, quando não há referências aos mesmos, prevenindo assim referências inválidas.
- ✓ Destrutores não existem, mas sim finalizadores. Porém, sua execução não é imediata.
- ✓ Não é permitida herança múltipla, mas uma classe pode implementar várias interfaces. O objetivo principal é simplificar a implementação do ambiente de execução.
- ✓ C# é mais seguro com tipos que C++. As únicas conversões implícitas por padrão são as seguras, tais como ampliação de inteiros e conversões de um tipo derivado para um tipo base. Não existem conversões implícitas entre inteiros e variáveis lógicas ou enumerações, assim como ponteiros nulos.
- ✓ Propriedades estão disponíveis, as quais permitem que métodos sejam chamados com a mesma sintaxe de acesso a membros de dados.

### **2.1.4 – Microsoft SQL Server 2005 Express Edition**

O Microsoft SQL Server 2005 Express Edition<sup>9</sup> (SQL Server Express) é um SGBD (sistema de gerenciamento de banco de dados) poderoso e confiável que fornece recursos robustos, proteção de dados e desempenho para clientes de aplicativos incorporados, aplicativos *Web* simples e armazenamentos de dados locais. Criado para ser de protótipo rápido e de fácil implantação, está disponível gratuitamente. O SQL Server Express foi criado para se integrar perfeitamente com outros investimentos em infra-estrutura de servidor.

Existe uma classe no .NET Framework exclusiva para acesso ao SQL Server, o que torna mais fácil utilizar toda a potência deste SGBD em uma aplicação .NET.

## **2.2 – Processos**

### **2.2.1 – Estação de Trabalho**

No desenvolvimento do sistema proposto, foi utilizado um notebook Toshiba STi com processador AMD Turion 64 1,8 GHz, 512 MB de memória RAM DDR 400, HD IDE de 60 GB.

### **2.2.2 – Ambiente de Desenvolvimento.**

Foi utilizado o sistema operacional Windows XP Home com Service Pack 2 instalado e, como ferramenta de desenvolvimento, o Visual Web Developer 2005 Express Edition com .NET Framework 2.0 e banco de dados Microsoft SQL Server 2005 Express Edition. O sistema foi feito em ASP.NET 2.0 usando Visual C# .NET 2.0 como linguagem de programação. Para testes, foi usado o ASP.NET Development Server, contido no Visual Web Developer.



## 2.3 – Justificativa da Configuração Utilizada

- ✓ Uso da linguagem C# por ser uma linguagem robusta, amplamente difundida e que implementa eficientemente o paradigma de orientação a objetos.
- ✓ Uso da plataforma .NET por conta da sua alta portabilidade, uma vez que os componentes desenvolvidos do referido sistema podem ser aproveitados em outro mesmo, caso este utilize outra linguagem do .NET Framework.
- ✓ Uso da plataforma ASP.NET devido à familiaridade com ASP (tecnologia usada anteriormente pelo desenvolvedor do projeto), dos componentes *Web* pré-desenvolvidos e do uso de programação orientada a eventos.
- ✓ Uso do SQL Server por conta da facilidade de gerenciamento do Banco de Dados, do uso de acesso remoto, da definição de relacionamentos entre tabelas de forma simplificada e da segurança dos dados, além do fato de existir uma classe .NET exclusiva para acesso ao SQL Server.
- ✓ Gratuidade das ferramentas propostas.

## 2.4 – Conclusão

Diante as características e vantagens apresentadas pelas citadas tecnologias, um sistema *Web* desenvolvido em ASP.NET e usando banco de dados SQL Server 2005 Express Edition pareceu a escolha mais confiável para desenvolver uma aplicação de alto desempenho, segura em relação aos dados, de fácil manutenção e de baixo custo de produção.

# Capítulo 3

## O Trabalho

### 3.1 – O Sistema

A seguir, são apresentadas todas as funcionalidades do sistema que foram implementadas até então.

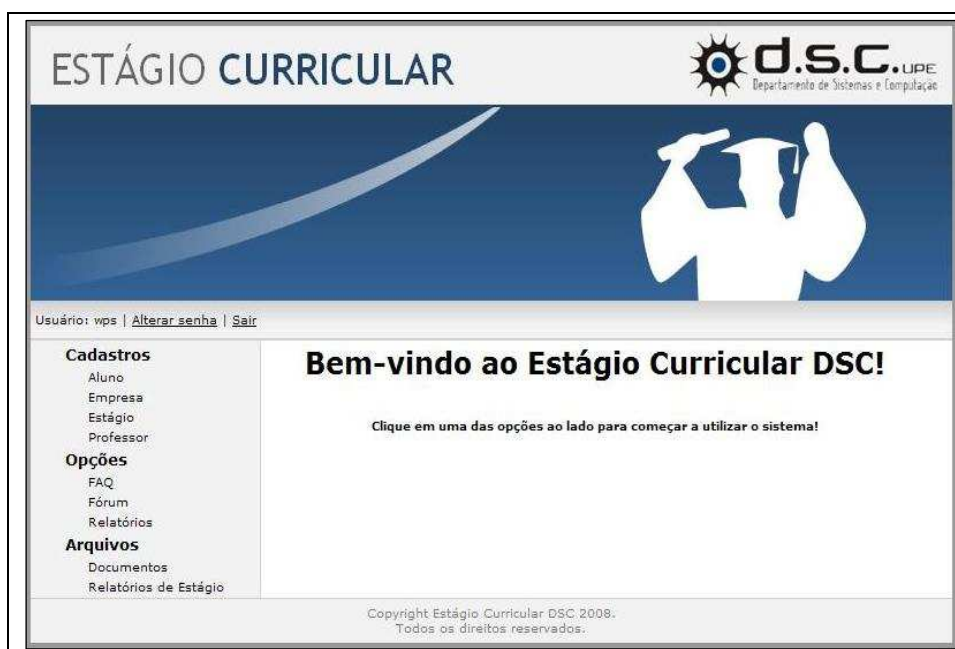
#### 3.1.1 – Login

Inicialmente, o usuário do sistema precisará efetuar *login* para poder ter acesso ao mesmo. Na página inicial, digita seu *login* e sua senha e clica no botão Entrar. Caso *login* e senha estejam corretos, o usuário passa para a página principal.



Figura 2. Página de *login* do sistema.

Na página principal, o usuário encontra ao lado esquerdo encontrará o menu de funcionalidades do sistema, dividido em Cadastros, Opções e Arquivos; ele é configurado de acordo com o tipo do usuário que efetuou *login*: aluno, professor orientador ou Coordenador de Estágio (vide **Anexo I**). Acima do mesmo, está a barra do usuário, que informa o *login* do mesmo e possui os *links* para alteração de senha e *logout* do sistema.



**Figura 3.** Página principal do sistema, vista pelo Coordenador de Estágio.

Caso o usuário seja um aluno ou professor orientador recentemente cadastrado, será obrigado a alterar sua senha, uma vez que estará usando uma senha provisória cadastrada automaticamente pelo sistema.

### **3.1.2 – Cadastros**

Os cadastros do sistema são acessíveis apenas para o Coordenador do Estágio. Eles dividem-se em cadastros de aluno, professor, empresa e estágio.

#### 3.1.2.1 – Cadastro de Aluno

Na página de inclusão de aluno, o usuário digita o número da matrícula, o nome, o *login* e o e-mail do aluno, e clica no botão Incluir para efetuar a inclusão. Caso não exista um aluno com a matrícula informada ou o *login*, a inclusão é realizada e é enviada ao e-mail informado uma mensagem informando que o discente foi cadastrado no sistema com uma senha provisória e que deve alterá-la quando fizer o primeiro acesso.



The screenshot displays the 'ESTÁGIO CURRICULAR' web application. At the top, the title 'ESTÁGIO CURRICULAR' is on the left, and the 'd.s.c. UPE' logo (Departamento de Sistemas e Computação) is on the right. Below the header is a blue banner with a white silhouette of a graduate. A navigation bar contains 'Usuário: wps | [Alterar senha](#) | [Sair](#)'. A left sidebar menu lists categories: 'Cadastros' (Aluno, Empresa, Estágio, Professor), 'Opções' (FAQ, Fórum, Relatórios), and 'Arquivos' (Documentos, Relatórios de Estágio). The main content area is titled '>> Alterar Aluno' and contains a form with the following fields: 'Matrícula:' (0326055), 'Nome:' (ANTHONY CAVALCANTI), 'Login:' (arcd), and 'E-mail:' (anthonycc2@gmail.com). Below the form are 'Alterar' and 'Excluir' buttons. The footer contains the text: 'Copyright Estágio Curricular DSC 2008. Todos os direitos reservados.'

**Figura 4.** Alteração de dados de aluno.

Clicando no *link* Consultar, é exibida a página de consulta de alunos, onde o usuário pode digitar o nome (ou parte dele) do discente e clicar no botão Consultar. O resultado da consulta é exibido em uma tabela logo abaixo do botão. Do lado esquerdo de cada linha da tabela, há um *link* Alterar. Ao clicar no mesmo, é exibida uma página com todas as informações do aluno cadastradas no sistema. Caso deseje, o usuário pode alterar o nome e/ou o e-mail do discente, clicando no botão Alterar, ou excluí-lo do sistema, clicando no botão Excluir.

### 3.1.2.2 – Cadastro de Empresa

Esta funcionalidade foi projetada para ser um simples cadastro do nome da empresa, podendo ser facilmente estendida no futuro para conter mais informações, caso necessário. É semelhante à de Cadastro de Aluno, descrita anteriormente.

### 3.1.2.3 – Cadastro de Professor

Esta funcionalidade é idêntica à de Cadastro de Aluno, descrita anteriormente.

### 3.1.2.4 – Cadastro de Estágio

Esta é uma das funcionalidades mais importantes do sistema. Na página de inclusão de estágio, o usuário escolhe o aluno, a empresa e o professor orientador, cada um deles disposto em uma caixa de seleção. Digita a data de início do estágio, a data pretendida pelo aluno para terminar o estágio e a data em que o estágio terminou; é necessário que estas datas obedeçam a uma ordem cronológica. Depois, escolhe a situação do mesmo (matriculado, cancelado, aprovado ou reprovado) e clica no botão Incluir para efetuar a inclusão.



The screenshot displays the 'ESTÁGIO CURRICULAR' web application interface. At the top, the title 'ESTÁGIO CURRICULAR' is on the left, and the logo for 'd.s.c. UPE Departamento de Sistemas e Computação' is on the right. Below the header, there is a navigation bar with links for 'Usuário: wps', 'Alterar senha', and 'Sair'. A sidebar on the left contains menu items under three categories: 'Cadastros' (Aluno, Empresa, Estágio, Professor), 'Opções' (FAQ, Fórum, Relatórios), and 'Arquivos' (Documentos, Relatórios de Estágio). The main content area is titled '>> Incluir Estágio' and contains a form with the following fields: 'Aluno' (dropdown menu with 'ANTHONY CAVALCANTI'), 'Empresa' (dropdown menu with 'CESAR'), 'Professor' (dropdown menu with 'ABEL GUILHERMINO'), 'Data do início' (text input), 'Data prevista do fim' (text input), and 'Data do fim' (text input). Below these fields is a 'Situação' section with four radio buttons: 'Aprovado', 'Cancelado', 'Matriculado' (which is selected), and 'Reprovado'. At the bottom of the form are two buttons: 'Incluir' and 'Consultar'. A footer at the very bottom reads 'Copyright Estágio Curricular DSC 2008. Todos os direitos reservados.'

**Figura 5.** Inclusão de Estágio Curricular.

Caso deseje consultar os estágios cadastrados, o usuário pode clicar no *link* Consultar. É exibida a página de consulta, onde o usuário pode digitar o nome (ou parte dele) do aluno e clicar no botão Consultar. O resultado da consulta, com os estágios cadastrados do referido discente, é exibido em uma tabela logo abaixo do botão. Do lado esquerdo de cada linha da tabela, há um *link* Alterar que, quando clicado, é exibida uma página com todas as informações do estágio cadastradas no sistema. Caso deseje, o usuário pode alterar as informações incluídas por ele, clicando no botão Alterar, ou excluir o cadastro, clicando no botão Excluir; apenas não pode alterar os dados referentes à marcação de reuniões de acompanhamento entre aluno e professor, uma vez que apenas o orientador pode fazê-lo (vide **seção 3.1.3.1**).



ESTÁGIO CURRICULAR

d.s.c. UPE  
Departamento de Sistemas e Computação

Usuário: wps | [Alterar senha](#) | [Sair](#)

**Cadastros**  
Aluno  
Empresa  
Estágio  
Professor

**Opções**  
FAQ  
Fórum  
Relatórios

**Arquivos**  
Documentos  
Relatórios de Estágio

>> Consultar Estágio

Nome do aluno:

| Aluno                  | Empresa       | Professor                  | Sit. | Alterar                 |
|------------------------|---------------|----------------------------|------|-------------------------|
| ANTHONY CAVALCANTI     | UNISYS BRASIL | SERGIO SOARES              | M    | <a href="#">Alterar</a> |
| LUIS CLAUDIO DORIA     | MOTOROLA      | ABEL GUILHERMINO           | M    | <a href="#">Alterar</a> |
| GEORGE ARTHUR SILVA    | ITECE         | SERGIO CAMPELLO            | C    | <a href="#">Alterar</a> |
| FRANCISCO HAMILTON JR  | TIM           | CRISTINE GUSMÃO            | A    | <a href="#">Alterar</a> |
| ROGERIO PONTUAL        | CESAR         | SERGIO SOARES              | R    | <a href="#">Alterar</a> |
| VICTOR MEDEIROS OUTTES | PORTO DIGITAL | WELLINGTON PINHEIRO SANTOS | M    | <a href="#">Alterar</a> |

Copyright Estágio Curricular DSC 2008.  
Todos os direitos reservados.

**Figura 6.** Consulta de Estágios Curriculares.

Uma vez que a situação do estágio é alterada para cancelado ou reprovado, as informações sobre o mesmo não estará mais disponível para alteração, mas apenas para visualização.

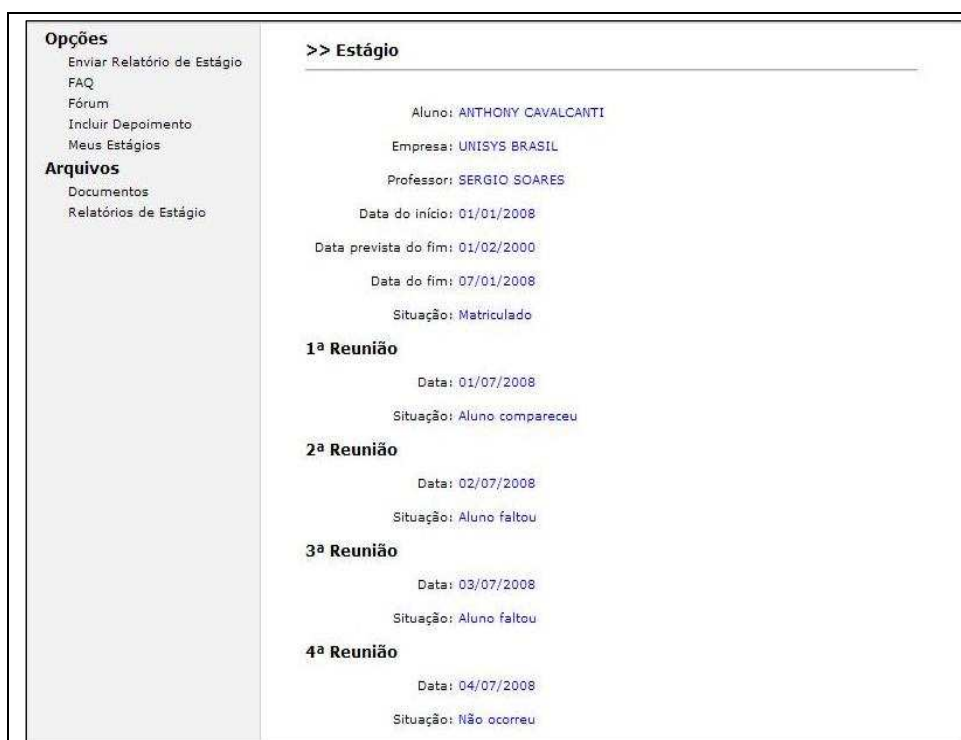
### 3.1.3 – Opções

As funcionalidades desta seção do menu estão disponíveis de acordo com o tipo de usuário. Elas são descritas a seguir.

#### 3.1.3.1 – Meus Estágios

Esta funcionalidade está disponível apenas para usuários dos tipos aluno e professor orientador, cada um com suas configurações próprias.

Para alunos, a página exibe os estágios cadastrados do discente em questão, permitindo que o mesmo, ao clicar no *link* Detalhar ao lado de cada linha da tabela contendo os cadastros, visualize todas as informações do estágio escolhido.



**Opções**  
Enviar Relatório de Estágio  
FAQ  
Fórum  
Incluir Depoimento  
Meus Estágios

**Arquivos**  
Documentos  
Relatórios de Estágio

**>> Estágio**

Aluno: ANTHONY CAVALCANTI  
Empresa: UNISYS BRASIL  
Professor: SERGIO SOARES

Data do início: 01/01/2008  
Data prevista do fim: 01/02/2000  
Data do fim: 07/01/2008  
Situação: Matriculado

**1ª Reunião**  
Data: 01/07/2008  
Situação: Aluno compareceu

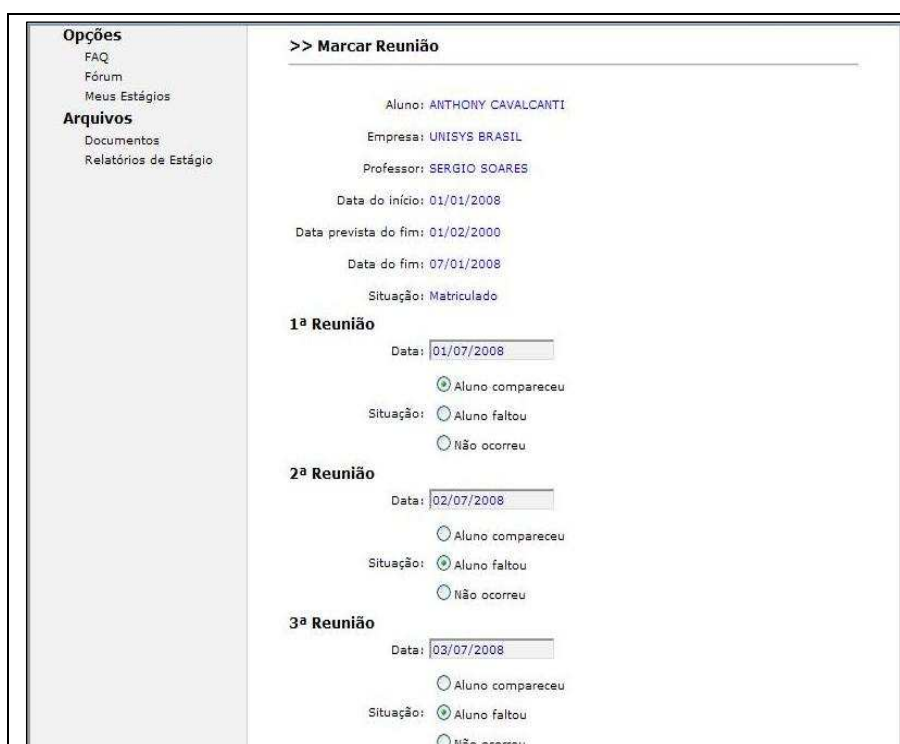
**2ª Reunião**  
Data: 02/07/2008  
Situação: Aluno faltou

**3ª Reunião**  
Data: 03/07/2008  
Situação: Aluno faltou

**4ª Reunião**  
Data: 04/07/2008  
Situação: Não ocorreu

Figura 7. Visualização de Estágio Curricular por um aluno.

Para professores orientadores, funciona de forma semelhante, com o diferencial da existência do *link* Marcar Reuniões, que remete a uma página onde é feita a marcação de reuniões de acompanhamento. Na página em questão, o docente pode informar as datas combinadas entre ele e o aluno para as reuniões de acompanhamento e definir sua situação (aluno compareceu, aluno faltou ou não ocorreu). Ao clicar no botão Alterar, são efetuadas as alterações.



**Opções**  
FAQ  
Fórum  
Meus Estágios

**Arquivos**  
Documentos  
Relatórios de Estágio

**>> Marcar Reunião**

Aluno: ANTHONY CAVALCANTI  
Empresa: UNISYS BRASIL  
Professor: SERGIO SOARES

Data do início: 01/01/2008  
Data prevista do fim: 01/02/2000  
Data do fim: 07/01/2008  
Situação: Matriculado

**1ª Reunião**  
Data: 01/07/2008  
Situação:  Aluno compareceu  
 Aluno faltou  
 Não ocorreu

**2ª Reunião**  
Data: 02/07/2008  
Situação:  Aluno compareceu  
 Aluno faltou  
 Não ocorreu

**3ª Reunião**  
Data: 03/07/2008  
Situação:  Aluno compareceu  
 Aluno faltou  
 Não ocorreu

Figura 8. Marcação de reuniões de acompanhamento.

### 3.1.3.2 – Relatórios

Disponível apenas para o Coordenador de Estágio, esta opção oferece geração de relatórios administrativos da Coordenação de Estágio, como:

- ✓ Distribuição de carga horária dos estágios.
- ✓ Distribuição de alunos em estágio por professor orientador.
- ✓ Listagem dos discentes em estágio com suas respectivas situações.

|  |               | <b>Departamento de Sistemas e Computação</b><br><b>Coordenação de Estágio</b><br>Benfica, 455, bloco C, 2º andar<br>Madalena, Recife, Pernambuco<br>CEP 50720-001<br>Fone: (81) 2119-3842<br>Fax: (81) 2119-3836 |                     |                     |      |
|---|---------------|--|---------------------|---------------------|------|
| 21/05/2008 10:29:21   |               |  |                     |                     |      |
| <b>Relatório de Alunos em Estágio Curricular</b>                                  |               |  |                     |                     |      |
| Aluno   | Empresa       | Professor  | Início              | Fim                 | Sit. |
| ANTHONY CAVALCANTI  | UNISYS BRASIL | SERGIO SOARES  | 01/01/2008 00:00:00 | 07/01/2008 00:00:00 | M    |
| LUIS CLAUDIO DORIA  | MOTOROLA      | ABEL GUILHERMINO   | 02/05/2008 00:00:00 | 02/11/2008 00:00:00 | M    |
| GEORGE ARTHUR SILVA   | ITECE         | SERGIO CAMPELLO  | 01/01/2008 00:00:00 | 02/10/2008 00:00:00 | C    |
| FRANCISCO HAMILTON JR   | TIM           | CRISTINE GUSMÃO  | 02/06/2008 00:00:00 | 02/10/2008 00:00:00 | A    |
| ROGERIO PONTUAL   | CESAR         | SERGIO SOARES  | 02/02/2008 00:00:00 | 02/08/2008 00:00:00 | R    |
| VICTOR MEDEIROS OUTTES  | PORTO DIGITAL | WELLINGTON PINHEIRO SANTOS   | 01/01/2008 00:00:00 | 01/07/2008 00:00:00 | M    |
| Estágio Curricular DSC 2008. Versão 1.1   |               |  |                     |                     |      |

Figura 9. Exemplo de relatório administrativo.

### 3.1.3.3 – Fórum

Disponível para todos os tipos de usuários, o fórum reúne depoimentos de situações pelas quais os alunos passaram no exercício dos seus estágios, divididos por assunto.

## ESTÁGIO CURRICULAR





Usuário: wps | [Alterar senha](#) | [Sair](#)

**Cadastros**

- Aluno
- Empresa
- Estágio
- Professor

**Opções**

- FAQ
- Fórum
- Relatórios

**Arquivos**

- Documentos
- Relatórios de Estágio

**>> Fórum**

Escolha o assunto: Programação

| Texto   | Usuário |
|---|---------|
| Estagiei em uma excelente empresa de desenvolvimento de software chamada Unisys Brasil. Lá fiz excelentes contatos profissionais; aprendi muitas dicas de programação e fui bem pago.                   | tony    |
| Eu tambem ja estagiei nesta empresa e asseguro que é um excelente local para se trabalhar. O ambiente de trabalho é muito bom, os colegas são muito competentes e experientes e se aprende bastante lá. | rp      |
| Estou procurando estágio e gostaria de saber se há alguma vaga disponível nesta empresa ou em outra similar. Tenho amplo conhecimento em Java e PHP, e conhecimento básico em C++.                      | vmo     |

Copyright Estágio Curricular DSC 2008.  
Todos os direitos reservados.



**Figura 10.** Fórum.

O usuário escolhe em uma barra de seleção o assunto do fórum que deseja consultar. Serão exibidos abaixo os depoimentos em ordem de inclusão, informando o *login* do aluno ao lado.

### 3.1.3.4 – Incluir Depoimento

Apenas alunos podem incluir depoimentos no fórum. O usuário precisa escolher um assunto para o depoimento, disposto em uma barra de seleção, e digitar o texto do depoimento que deseja deixar público para todos os usuários que consultem o fórum. Ao clicar no botão Incluir, é efetuada a inclusão do depoimento.

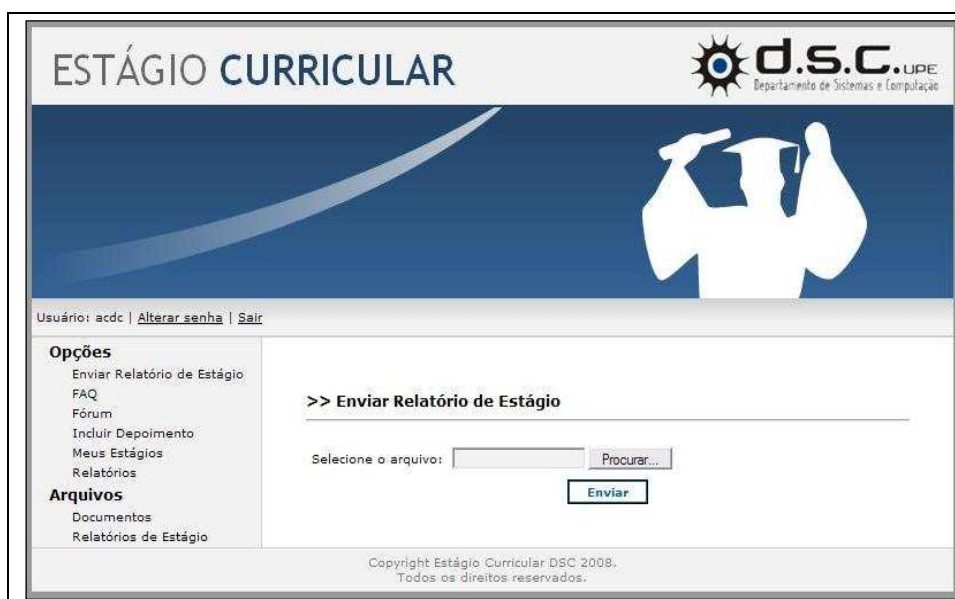


The screenshot displays the 'ESTÁGIO CURRICULAR' web application interface. At the top, the title 'ESTÁGIO CURRICULAR' is on the left, and the logo for 'd.s.c. LIPE Departamento de Sistemas e Computação' is on the right. Below the header, there is a navigation bar with the user 'ecdc' and links for 'Alterar senha' and 'Sair'. The main content area is titled '>> Incluir Depoimento'. It features a dropdown menu for 'Assunto' with 'Banco de Dados' selected, and a large text area for 'Texto:'. An 'Incluir' button is positioned at the bottom of the form. A sidebar on the left lists 'Opções' (Enviar Relatório de Estágio, FAQ, Fórum, Incluir Depoimento, Meus Estágios, Relatórios) and 'Arquivos' (Documentos, Relatórios de Estágio). The footer contains the text 'Copyright Estágio Curricular DSC 2008. Todos os direitos reservados.'

**Figura 11.** Inclusão de depoimento no fórum.

### 3.1.3.5 – Enviar Relatório de Estágio

Uma vez que seu estágio tenha sido aprovado, o aluno pode, por meio desta funcionalidade, realizar o *upload* do seu Relatório de Estágio para sistema, para que ele fique disponível na área Arquivos (vide seção 3.1.4.2).



**Figura 12.** Envio de Relatório de Estágio.

Nesta página, o usuário clica no botão Procurar. É aberta uma janela do próprio sistema operacional pela qual pode navegar pelos diretórios do computador e escolher o arquivo que deseja enviar. Este arquivo precisa ser de texto, ter a extensão permitida (PDF) e tamanho até 03 (três) MB (megabytes). Ao clicar no botão Enviar, o envio do arquivo é efetuado.

#### 3.1.3.6 – FAQ

O FAQ (abreviatura para *Frequently Asked Questions*) está presente nos mais variados tipos de *sites* na internet, reunindo perguntas comumente perguntadas por usuários. No sistema em questão, tem o objetivo de sanar as dúvidas mais comuns dos alunos em relação ao Estágio Curricular, como: qual o tempo de duração do Estágio Curricular, qual o seu critério de avaliação, o que é necessário para realizar a matrícula, etc. Está disponível para todos os tipos de usuários.

### **3.1.4 – Arquivos**

Os *links* desta seção do menu estão disponíveis para todos os tipos de usuários.

#### 3.1.4.1 – Documentos

Esta página disponibiliza para *download* documentos e modelos úteis para os alunos durante o curso de Estágio Curricular, como:

- ✓ Manual do Estágio Supervisionado da POLI.
- ✓ Ficha de Solicitação de Professor Orientador.
- ✓ Modelo de Relatório de Estágio da Divisão de Estágio da POLI.

#### 3.1.4.2 – Relatórios de Estágio

Esta página disponibiliza para *download* o acervo de Relatórios de Estágio enviados pelos alunos ao fim dos seus respectivos estágios (vide **seção 3.1.3.6**).

## 3.2 – Arquitetura

A arquitetura do sistema está dividida em três camadas: *Web Forms*, *DAOs* e *Framework*, como mostra a Figura 13.

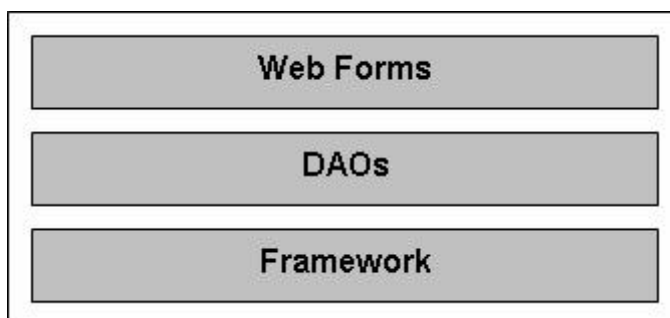


Figura 13. Arquitetura do sistema.

Os *Web Forms* são as páginas que serão acessadas pelo *browser* e fazem a interface com o usuário. Os *DAOs* (*Data Access Objects*) fazem o acesso ao Banco de Dados e implementam a lógica de negócio. Já o *Framework* do sistema é composto de entidades de negócio, biblioteca de exceções e listas de funções e de constantes.

## 3.3 – Metodologia

### 3.3.1 – Atividades

A seguir, são apresentadas as atividades realizadas no desenvolvimento deste projeto.

#### 3.3.1.1 – Levantamento de requisitos

Foram feitas entrevistas com o Coordenador da Divisão de Estágio a fim de coletar o máximo de detalhes possíveis do processo administrativo dos estágios dos alunos, bem como sugestões e desejos para melhorias no referido processo.

#### 3.3.1.2 – Definição da metodologia e tecnologia utilizadas

Com base nos requisitos colhidos na etapa anterior, foram definidos a plataforma de trabalho, o ambiente de desenvolvimento, a tecnologia utilizada, e linguagem de programação adotada e o banco de dados que fará o armazenamento das informações dos Estágios Curriculares.

#### 3.3.1.3 – Documentação

Foram confeccionados o Documento de Requisitos do Sistema, o Planejamento da Monografia e o Projeto da Monografia, seguindo os padrões adotados pela metodologia de desenvolvimento.

#### 3.3.1.4 – Obtenção de material pedagógico

Foi feita a aquisição de livros especializados em desenvolvimento de software disponíveis no mercado, além de uma busca em sites na internet por material que pudesse sugerir melhorias ou novidades.

#### 3.3.1.5 – Definição da hospedagem

Foi definido em que parte do servidor do DSC o sistema ficará hospedado. O ambiente foi preparado para suportar o sistema de acordo com os requisitos obtidos na seção 3.2.1.2: ambiente Windows 2000 ou superior, servidor *web* IIS 6.0 e banco de dados SQL Server 2000 ou superior. Foi realizada a instalação de servidor de FTP para facilitar o envio de arquivos.

#### 3.3.1.6 – Definição da Interface gráfica

O desenvolvimento dessa atividade foi feito utilizando-se Macromedia Fireworks 8 e Macromedia Dreamweaver 8. Foi buscada uma interface leve e atrativa, com a qual tanto professores quanto alunos possam se familiarizar.

#### 3.3.1.7 – Implementação do sistema

Foi realizada de acordo com todas as especificações obtidas na **seção 3.2.1.2**: ambiente Windows 2000 ou superior, servidor Web IIS 6.0, banco de dados SQL Server 2000 ou superior, plataforma ASP.NET e linguagem C#.

#### 3.3.1.8 – Escrita da Monografia

Foi realizada ao logo do desenvolvimento do projeto, a fim de que no final da disciplina, seja feita a defesa da mesma para a bancada de professores.

### **3.3.2 – Cronograma de Atividades**

1. Levantamento de requisitos
2. Definição da metodologia e tecnologia utilizadas
3. Documentação
4. Obtenção de material pedagógico
5. Definição da hospedagem
6. Definição da Interface gráfica
7. Implementação do sistema
8. Escrita da Monografia

**Tabela 2.** Cronograma das Atividades (03/03 - 27/05)

| Atividade | Semana |   |   |   |   |   |   |   |   |    |    |    |    |
|-----------|--------|---|---|---|---|---|---|---|---|----|----|----|----|
|           | 1      | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |
| 1         | X      |   |   |   |   |   |   |   |   |    |    |    |    |
| 2         |        | X |   |   |   |   |   |   |   |    |    |    |    |
| 3         |        |   | X | X |   |   |   |   |   |    |    |    |    |
| 4         |        |   | X | X |   |   |   |   |   |    |    |    |    |
| 5         |        |   |   | X |   |   |   |   |   |    |    |    |    |
| 6         |        |   |   |   | X |   |   |   |   |    |    |    |    |
| 7         |        |   |   |   | X | X | X | X | X | X  | X  | X  |    |
| 8         | X      | X | X | X | X | X | X | X | X | X  | X  | X  | X  |

### **3.4 – Lições Aprendidas no Desenvolvimento**

A principal lição aprendida durante o desenvolvimento desse projeto foi o conhecimento de uma nova tecnologia, o ASP.NET, que permite ao desenvolvedor utilizar as melhores práticas de desenvolvimento de sistemas *Web*: programação orientada a objetos, reusabilidade de componentes, modularização de código e tratamento de erros, entre outras.

A segunda lição importante é a importância do uso de práticas de Gerenciamento de Projetos, que são imprescindíveis em qualquer desenvolvimento de produto, a fim de evitar atrasos de cronograma, e gastos extras que possam aumentar excessivamente o custo de produção do sistema, o que pode comprometer a aquisição do sistema por parte do cliente ou tornar o projeto inviável.

### **3.5 – Recomendações**

Apesar de ser necessário apenas um computador com acesso à Internet e um *brower* para navegação em *sites*, para melhor utilização do sistema, recomenda-se usar o sistema operacional Microsoft Windows versão 2000 ou superior com o *browser* Microsoft Internet Explorer versão 6.0 ou superior, com permissão de execução de *scripts*.

### **3.6 – Conclusão**

Diante do que foi apresentado nas seções anteriores deste capítulo, pode-se constatar que este projeto foi desenvolvido usando tecnologias de ponta, como ASP.NET e SQL Server, a fim de se obter uma aplicação segura e de alto desempenho, e técnicas modernas de Engenharia de Software e Gerenciamento de Projetos, objetivando uma produção planejada e prevenida de contratempos e gastos extras.

# Capítulo 4

## Conclusões e Trabalhos Futuros

### 4.1 – Contribuições

Desenvolvido usando tecnologias de ponta e técnicas modernas de Engenharia de Software e Gerenciamento de Projetos, este projeto traz vários benefícios visíveis para a Coordenação de Estágio do DSC, alguns dos quais são:

- ✓ Ao permitir o gerenciamento de forma automatizada dos alunos em Estágio Curricular e geração de relatórios, acaba-se com acúmulo excessivo de documentos feito de forma manual e a duplicação de informações.
- ✓ Ao descentralizar e dividir o controle de informações dos Estágios Curriculares entre o Coordenador de Estágio e os professores orientadores, evita-se a sobrecarga de tarefas sobre uma única pessoa.
- ✓ Ao prover uma análise prévia a respeito da qualidade do estágio proposto, evita-se que os alunos se matriculem em estágios não recomendados pelo Departamento.
- ✓ Ao disponibilizar uma vasta biblioteca de relatórios de estágio e um fórum para alunos, proporciona-se a troca de experiências profissionais e conhecimento técnico entre alunos.

Após a implantação deste sistema, são grandes as chances de se atingir uma melhora no controle em relação às reuniões de acompanhamento e na obtenção de informações sobre o status atual dos estágios, além do aumento da participação dos alunos em relação ao Estágio Curricular, uma vez que os mesmos poderão acompanhar sua situação através do sistema e interagir com outros estagiários através do fórum.

Por fim, ficam claros os benefícios que se obtém ao adotar a Tecnologia da Informação como base nas operações de um departamento acadêmico: agilidade, eficiência, integridade de informações, precaução contra dados repetidos ou dúbios, automatização de serviços e informações analíticas de carga horária e notas, entre outros. Benefícios estes que, no caso em questão, têm grandes chances de amenizar as dificuldades encontradas no gerenciamento dos alunos em Estágio Curricular e até mesmo sanar determinadas deficiências do processo atual.

### 4.2 – Trabalhos Futuros

- ✓ Geração de relatórios em formato PDF, para possibilitar salvamento dos relatórios no computador do usuário do sistema.

- ✓ Geração de atas de presença em reuniões de acompanhamento de alunos em Estágio Curricular.
- ✓ Visualização de consultas em formato de gráfico.
- ✓ Implementação de acesso a Banco de Dados através de *Stored Procedures*, para otimização do processo.
- ✓ Atualização da plataforma do sistema para ASP.NET 3.5, a mais atual até o momento.
- ✓ Uso de novas tecnologias desenvolvidas pela Microsoft que integram o .NET Framework 3.5, como LINQ e WPF.
- ✓ Implantação de criptografia através do protocolo de segurança SSL para acesso ao sistema.
- ✓ Criação de componentes *web* personalizados para o sistema gerados em tempo de execução de acordo com o tipo de usuário usando repositórios em XML.

# Bibliografia

- [1] Pressman, R. S. (2006). Engenharia de Software. McGraw-Hill Books.
- [2] Reynolds, G. W., Stair, R. M. (2005). Princípios de Sistemas de Informação. Thompson Editora.
- [3] Lotar, A. (2003). ASP.NET com C#: Curso Prático. Novatec Editora.
- [4] Ullman, C., Kauffman, J., Hart, C., Sussman, D., Maharry, D. (2004). *Beginning ASP.NET 1.1 with Visual C#.NET 2003*. Wiley Publishing, Inc.
- [5] Utley, C. (2006). Desenvolvendo Aplicativos para Web com SQL Server 2000. Makron Books.
- [6] Tulloch, M., Santry, P. (2001). Dominando IIS 5.0. Editora Moderna.
- [7] ARAÚJO, A. V. (2006). Treinamento Avançado em .NET. Digerati Books.
- [8] Wikipédia. (2008). <http://www.wikipedia.org>, último acesso em 04/05/2008.
- [9] MSDN Brasil. (2008). <http://msdn.microsoft.com/pt-br/default.aspx>, último acesso em 20/05/2008.
- [10] DevMedia Editora - Central de Conteúdo .Net. (2008). <http://www.devmedia.com.br/canais/default.asp?site=35>, último acesso em 20/05/2008.
- [11] Dez razões para migrar para ASP.NET. (2008). [http://www.microsoft.com/brasil/msdn/tecnologias/aspnet/aspnet\\_migrar.aspx](http://www.microsoft.com/brasil/msdn/tecnologias/aspnet/aspnet_migrar.aspx), último acesso em 15/11/2006.



# Apêndice A

## Caso de Negócio

### 1.0 – Objetivos

Explicar o projeto que está sendo desenvolvido, o Estágio Curricular DSC, analisando aspectos tais como: descrição, características gerais do produto, aspectos financeiros, etc.

### 2.0 – Referências

Análise de Riscos, Especificação de Casos de Uso e Análise de Casos de Uso.

### 3.0 – Descrição do Produto

O sistema fornece ao Coordenador de Estágio do Departamento de Sistemas e Computação (DSC) da Escola Politécnica de Pernambuco (POLI) acesso completo aos cadastros de alunos, empresas, estágios e professores orientadores e a relatórios administrativos da Coordenação de Estágio. O professor orientador marca reuniões de acompanhamento e registra o comparecimento do aluno às mesmas. O aluno, por sua vez, posta em um fórum depoimentos sobre exercício da sua função e, ao fim do estágio, envia seu Relatório de Estágio para livre consulta por outros usuários.

O sistema ainda disponibiliza a todos os usuários um FAQ, documentos úteis para o decurso da disciplina e um acervo de Relatórios de Estágio.

### 4.0 – Contexto do Negócio

O sistema tem como foco o uso a Coordenação de Estágio do DSC da POLI, mas a idéia é que qualquer faculdade ou universidade que tenha interesse no produto possa adquiri-lo e utilizá-lo sem maiores transtornos, uma vez que, as funcionalidades implementadas são comuns a praticamente todas as instituições.

Com relação ao nicho de mercado, uma breve análise nos mostra que existem vários clientes em potencial, visto à grande quantidade de faculdades existentes atualmente no cenário nacional.

O sistema será desenvolvido na forma mais geral possível, podendo ser facilmente adaptado e vendido qualquer cliente.

## 5.0 – Objetivos do Produto

O desenvolvimento do Estágio Curricular DSC é fundamentado em uma análise criteriosa dos requisitos, bem como na realização de testes práticos para observação do comportamento do sistema.

A viabilidade do produto é comprovada por um conjunto de fatores, dentre os quais destacamos: *software* simples, baixo custo, rápido desenvolvimento, um cliente como garantia, vários outros clientes em potencial e nenhum concorrente conhecido no mercado.

O prazo previsto para a conclusão do projeto, desde a sua concepção até a sua implantação, é de três meses, detalhados no cronograma abaixo:

**Tabela 3.** Cronograma Inicial.

| <b>Cronograma Inicial</b>                        |              |              |              |
|--|--------------|--------------|--------------|
| <b>Etapa</b>                                     | <b>Mês 1</b> | <b>Mês 2</b> | <b>Mês 3</b> |
| Levantamento dos requisitos                      | X            |              |              |
| Definição da metodologia e tecnologia utilizadas | X            |              |              |
| Documentação                                     | X            |              |              |
| Obtenção de material pedagógico                  | X            |              |              |
| Definição da hospedagem                          | X            |              |              |
| Definição da Interface gráfica                   | X            |              |              |
| Implementação do sistema                         | X            | X            |              |
| Escrita da Monografia                            | X            | X            | X            |

## 6.0 – Previsão Financeira

Por se tratar de uma aplicação que será desenvolvida por um único programador, de forma independente, durante o decorrer da disciplina Projeto de Final de Curso do curso de Engenharia da Computação da POLI, com previsão de ser implantada no DSC como doação, não há nenhum custo ou lucro diretamente associados ao Estágio Curricular DSC no momento. Porém, ao se imaginar os clientes em potencial que podem se interessar em adquirir-lo sistema, pode-se estipular uma margem de preço e lucro para o mesmo.

Por se tratar de um sistema simples, de rápido desenvolvimento e que usa plataforma robusta, o Estágio Curricular DSC pode ser facilmente adaptado para as necessidades que qualquer cliente.

Leve-se em conta que em uma capital ou grande cidade há cerca de 10 (dez) possíveis clientes (universidades e faculdades públicas e privadas). Ao se supor que, destes, apenas 20% (vinte por cento) se interessem pelo produto, ou seja, dois clientes, por motivos diversos (impossibilidade financeira, possuir sistema similar, o software não agradou, etc.), sendo cada cópia vendida por um preço estipulado inicialmente em R\$ 2.000,00 (dois mil reais), chega-se a conclusão de que com a venda de apenas duas licenças, ter-se-á um lucro de R\$ 4.000,00 (quatro mil reais), o que é considerável, tendo em vista que o investimento é praticamente nulo.

## 7.0 – Restrições



Por tratar-se de uma aplicação *web*, é requerido para hospedar o sistema um servidor *web* com processador de 2,0 GHz de velocidade ou superior, HD de 80 GB de capacidade ou superior, memória RAM de 256 MB de capacidade ou superior, sistema operacional Windows 2000 ou superior com .NET Framework versão 2.0 ou superior, banco de dados SQL Server Express Edition versão 2005 ou superior e servidor *web* IIS versão 6.0 instalados.

Nos terminais de operação, os únicos requisitos são acesso à Internet e um *browser* para navegação de *sites*. Recomenda-se usar processador de 1,8 GHz de velocidade ou superior, memória RAM de 256 MB de capacidade ou superior, sistema operacional Microsoft Windows 2000 ou superior com *browser* Internet Explorer 6.0 ou superior.

# Apêndice B

## Análise de Riscos

### 1.0 - Objetivos

Neste documento os riscos serão listados e identificados Além disso, cada risco será avaliado com relação a diferentes aspectos, tais como: descrição, magnitude, impacto, planos de contingência, etc. Tudo isso, para que com este documento em mãos, possa-se avaliar a importância de uma Análise de Riscos; prever, listar e analisar os riscos inerentes ao desenvolvimento de um *software*; pôr em funcionamento a boa prática da Engenharia de Software e, principalmente, minimizar ou até mesmo eliminar os impactos caso os riscos tornem-se eventos concretos.

Foi levado em consideração no levantamento dos riscos que este se trata de um projeto simples, desenvolvido por uma única pessoa (que incorpora os personagens de Analista de Requisitos, Arquiteto de Software, Gerente de Projeto e Desenvolvedor) durante o decorrer de uma disciplina universitária, sem custo ou orçamento associado e, a princípio, sem fins lucrativos, pois tem a finalidade de ser doado.

### 2.0 - Referências

- ✓ Pressman, R. S. (2006). Engenharia de Software. McGraw-Hill Books.

## 3.0 – Riscos

### 3.1 – A tecnologia utilizada para o desenvolvimento do projeto não ser a mais indicada

|   |  |                   |       |
|---|--|-------------------|-------|
| <b>ID:</b>  | RC01   | <b>Magnitude:</b> | Baixa |
| <b>Nome:</b>  | A tecnologia utilizada para o desenvolvimento do projeto não ser a mais indicada |                   |       |
| <b>Descrição:</b>   |  |                   |       |
| A tecnologia utilizada para o desenvolvimento do projeto (linguagem de programação, ferramenta de desenvolvimento, servidor de banco de dados, etc.) não ser a mais indicada.   |  |                   |       |
| <b>Impacto:</b>   |  |                   |       |
| Dificuldade de se implementar o que está previsto no projeto, atrasos no cronograma, a necessidade de se refazer o <i>software</i> do ponto inicial (mesmo que este esteja em fase final de desenvolvimento), mal desempenho do <i>software</i> .   |  |                   |       |
| <b>Estratégia de Controle:</b>  |  |                   |       |
| A tecnologia escolhida deve ser avaliada cuidadosamente, levando-se em conta os requisitos do projeto, o desenvolvedor deve estar plenamente treinado na mesma. É importante também modularizar o projeto, de forma que se um módulo der problema, ele possa ser substituído por outro feito em outra tecnologia.   |  |                   |       |
| <b>Indicadores:</b>   |  |                   |       |
| Falta de documentação de apoio da tecnologia adotada, dificuldade de se encontrar profissionais habilitados a trabalhar com tal tecnologia, mal desempenho do <i>software</i> , entre outros.   |  |                   |       |
| <b>Plano de Contingência:</b>   |  |                   |       |
| Primeiramente, deve-se substituir os módulos problemáticos por outros desenvolvidos em outra tecnologia. Caso todo o projeto esteja comprometido, a solução é migrar completamente para outra tecnologia. Para isso, é preciso primeiramente negociar com o cliente um novo prazo de entrega; caso o mesmo seja aprovado, um novo cronograma será feito e o desenvolvedor será treinado na nova tecnologia. |  |                   |       |

### 3.2 – Levantamento incorreto dos requisitos do projeto

|                                |  |                   |      |
|--------------------------------|--|-------------------|------|
| <b>ID:</b>                     | RC02   | <b>Magnitude:</b> | Alta |
| <b>Nome:</b>                   | Levantamento incorreto dos requisitos do projeto   |                   |      |
| <b>Descrição:</b>              | Os requisitos do sistema são levantados inicialmente de forma errada, seja por má interpretação por parte do Analista de Requisitos, omissão de requisitos necessários ou por simples má descrição dos mesmos por parte do cliente, ou então eles são mais complexos do que o esperado.                  |                   |      |
| <b>Impacto:</b>                | Desenvolvimento de um sistema totalmente diferente das expectativas do cliente, gerando insatisfação do mesmo com o produto, o que pode provocar o cancelamento do projeto.  |                   |      |
| <b>Estratégia de Controle:</b> | O Analista de Requisitos ficará em encarregado de fazer uma entrevista inicial com o cliente para obter do mesmo todos os requisitos necessários para o sistema, de forma que assim se possa minimizar a possibilidade de um requisito não ser coletado ou ser coletado de forma errônea.                |                   |      |
| <b>Indicadores:</b>            | Nas reuniões com o cliente, as funcionalidades desenvolvidas não estão de acordo com o que era esperado e os testes do <i>software</i> não saem de acordo com o que ele especificou. Ou ainda, a o desenvolvedor apresenta dificuldade excessiva em tratar alguns dos problemas, atrasando o cronograma. |                   |      |
| <b>Plano de Contingência:</b>  | Caso o sistema final não atenda aos desejos do cliente, inicialmente deve-se negociar com o mesmo um novo prazo de entrega; caso seja aprovado, será feita uma nova entrevista para se levantar novamente os requisitos do sistema. Em seguida, deve-se proceder às alterações necessárias.              |                   |      |

### 3.3 – Não conseguir implementar o sistema da forma prevista

|                                |  |                   |      |
|--------------------------------|--|-------------------|------|
| <b>ID:</b>                     | RC03   | <b>Magnitude:</b> | Alta |
| <b>Nome:</b>                   | Não conseguir implementar o sistema da forma prevista  |                   |      |
| <b>Descrição:</b>              | Risco de não conseguir implementar o sistema da forma prevista.  |                   |      |
| <b>Impacto:</b>                | Insatisfações por parte do cliente, resultados inconsistentes, fraco desempenho, extensão do cronograma para realizar reestruturação e até mesmo o cancelamento do projeto.  |                   |      |
| <b>Estratégia de Controle:</b> | Fazer testes do <i>software</i> com a presença do cliente para perceber possíveis falhas ou funcionalidades implementadas de forma errada.                                   |                   |      |
| <b>Indicadores:</b>            | Durante a fase de testes do sistema, ocorrem resultados não previstos e reclamações por parte do cliente de que o <i>software</i> não está funcionando da maneira planejada. |                   |      |
| <b>Plano de Contingência:</b>  | Negociar com o cliente um prazo de entrega para as atualizações do sistema. Caso seja aprovado, reavaliar os casos de uso e implementar as mudanças necessárias.             |                   |      |

### 3.4 – Interface do sistema apresenta alta complexidade

|   |  |                   |       |
|---|--|-------------------|-------|
| <b>ID:</b>  | RC04   | <b>Magnitude:</b> | Média |
| <b>Nome:</b>  | Interface do sistema apresenta alta complexidade |                   |       |
| <b>Descrição:</b>   |  |                   |       |
| Devido às funcionalidades requisitadas pelo cliente o sistema apresenta uma interface com o usuário muito complexa e possivelmente pouco amigável.  |  |                   |       |
| <b>Impacto:</b>   |  |                   |       |
| Os usuários do sistema terão dificuldades em aprender a operá-lo, podem se sentir desmotivados a usá-lo e serão necessárias mais horas de treinamento.  |  |                   |       |
| <b>Estratégia de Controle:</b>  |  |                   |       |
| Coletar informações e/ou dúvidas dos usuários a fim de tentar realizar o treinamento no em paralelo com o desenvolvimento do <i>software</i> , de modo que o tempo gasto seja minimizado. Haverá um estudo forte no sentido de separar funcionalidades correlatas para evitar sobrecarga de informações na tela e deixar a interface mais amigável. |  |                   |       |
| <b>Indicadores:</b>   |  |                   |       |
| Queixas por parte dos usuários de que estão tendo dificuldades excessivas em operar o sistema ou mesmo que não estão conseguindo fazê-lo.   |  |                   |       |
| <b>Plano de Contingência:</b>   |  |                   |       |
| Elaboração de uma equipe com a finalidade de treinar os usuários do sistema para que suas dificuldades sejam identificadas e sanadas o quanto antes.  |  |                   |       |

### 3.5 – Faltar equipamentos para o desenvolvimento do projeto

|  |   |                   |      |
|--|---|-------------------|------|
| <b>ID:</b>   | RC05  | <b>Magnitude:</b> | Alta |
| <b>Nome:</b>   | Faltar equipamentos para o desenvolvimento do projeto |                   |      |
| <b>Descrição:</b>  |   |                   |      |
| Faltam equipamentos necessários para o desenvolvimento do projeto (periféricos ou computadores completos).   |   |                   |      |
| <b>Impacto:</b>  |   |                   |      |
| Atrasos no cronograma, gastos extras para adquirir os equipamentos necessários e, em uma situação mais drástica, o não desenvolvimento do projeto.   |   |                   |      |
| <b>Estratégia de Controle:</b>   |   |                   |      |
| Verificar no início do desenvolvimento do projeto o programador tem a sua disposição todos os equipamentos necessários para o bom desempenho do seu trabalho e providenciar prontamente a compra daqueles que estiverem em falta.          |   |                   |      |
| <b>Indicadores:</b>  |   |                   |      |
| Ociosidade por falta de máquinas para trabalhar, equipamentos com requisitos abaixo do necessário para o desenvolvimento, sobrecarga de equipamentos que possuem os requisitos mínimos, entre outros.                                      |   |                   |      |
| <b>Plano de Contingência:</b>  |   |                   |      |
| Efetuar com urgência a compra dos equipamentos necessários. Caso não se disponha de capital suficiente, recorrer a um empréstimo bancário. Caso o mesmo não seja alcançado, locar os equipamentos em uma firma de aluguel de computadores. |   |                   |      |

### 3.6 – Mudanças bruscas no desenvolvimento impostas pelo cliente numa etapa avançada

|   |   |                 |      |
|---|---|-----------------|------|
| <b>ID:</b>  | RC06  | <b>Ranking:</b> | Alta |
| <b>Nome:</b>  | Mudanças bruscas no desenvolvimento impostas pelo cliente numa etapa avançada |                 |      |
| <b>Descrição:</b>   |   |                 |      |
| O cliente solicita mudanças bruscas no projeto, como a adição de mais funcionalidades ao projeto após a coleta dos requisitos, o que pode causar atraso no projeto, pois o mesmo precisará novamente passar pelas fases de planejamento.  |   |                 |      |
| <b>Impacto:</b>   |   |                 |      |
| Atraso no cronograma. Em caso de mudanças críticas e de complexidade elevada, o projeto pode ser inviabilizado por parte do desenvolvedor.  |   |                 |      |
| <b>Estratégia de Controle:</b>  |   |                 |      |
| Deixar claro ao cliente na fase de planejamento do projeto que modificações de alta complexidade imprevistas podem inviabilizar o projeto.  |   |                 |      |
| <b>Indicadores:</b>   |   |                 |      |
| O cliente informa que necessita de mais funcionalidades no <i>software</i> ou que deseja a mudanças de algumas já implementadas, de modo a suprir totalmente suas necessidades.   |   |                 |      |
| <b>Plano de Contingência:</b>   |   |                 |      |
| O Gerente do Projeto deve informar ao cliente que mudanças nesta etapa da elaboração implicará em mais tempo necessário para a conclusão do mesmo do que o que foi acertado inicialmente. Caso o mesmo aceite estas condições, reajustar o cronograma, fazer um novo levantamento de requisitos e dar início às modificações solicitadas. |   |                 |      |



# Apêndice C

## Análise de Casos de Uso

### 1.0 - Efetuar Login

#### 2.1 – Classe Usuário

**Descrição:** classe que contém informações cadastrais de um usuário do sistema.

**Tabela 4.** Descrição da classe Usuário.

| Atributos | Descrição   | Tipo   |
|-----------|---|--------|
| login     | <i>Login</i> de acesso ao sistema.                            | String |
| senha     | Senha de acesso ao sistema.                                   | String |
| tipo      | Tipo do usuário: aluno (A), professor (P) ou Coordenador (C). | char   |

#### 2.2 – Classe DAOUsuario

**Descrição:** classe que faz o acesso à tabela USUARIO do Banco de Dados.

**Tabela 5.** Descrição da classe DAOUsuario.

| Métodos          | Descrição  |
|------------------|--|
| Inserir(usuario) | Insere o cadastro de um usuário no Banco de Dados.             |
| Alterar(usuario) | Altera o cadastro de um usuário no Banco de Dados.             |
| Excluir(login)   | Exclui o cadastro de um usuário do Banco de Dados              |
| Consultar(login) | Recupera o cadastro de um usuário do Banco de Dados.           |
| Existe(usuario)  | Verifica se o cadastro de um usuário existe no Banco de Dados. |

### 2.0 - Alterar Senha

Semelhante a Efetuar Login.

### 3.0 – Incluir Aluno

#### 3.1 – Classe Acadêmico

**Descrição:** classe abstrata que contém informações cadastrais de um acadêmico, que pode ser um aluno ou um professor. Esta classe precisa ser herdada.

**Tabela 6.** Descrição da classe Acadêmico.

| Atributos | Descrição  | Tipo   |
|-----------|--|--------|
| matricula | Número de Matrícula do acadêmico na POLI.        | String |
| nome      | Nome do acadêmico.                               | String |
| login     | <i>Login</i> de acesso ao sistema do acadêmico . | String |
| email     | E-mail para contato do acadêmico.                | String |

### 3.2 – Classe Aluno

**Descrição:** subclasse da classe Acadêmico que contém informações cadastrais de um aluno. Não possui nenhum atributo adicional.

### 3.3 – Classe Usuário

Já descrita anteriormente.

### 3.4 – Classe DAOAluno

**Descrição:** classe que faz o acesso à tabela ALUNO do Banco de Dados.

**Tabela 7.** Descrição da classe DAOAluno.

| Métodos              | Descrição  |
|----------------------|--|
| Inserir(aluno)       | Insere o cadastro de um aluno no Banco de Dados.             |
| Alterar(aluno)       | Altera o cadastro de um aluno no Banco de Dados.             |
| Excluir(matricula)   | Exclui o cadastro de um aluno do Banco de Dados              |
| Consultar(matricula) | Recupera o cadastro de um aluno do Banco de Dados.           |
| Existe(aluno)        | Verifica se o cadastro de um aluno existe no Banco de Dados. |

### 3.5 – Classe DAOUsuario

Já descrita anteriormente.

## 4.0 – Consultar Aluno

### 4.1 – Classe Aluno

Já descrita anteriormente.

### 4.2 – Classe DAOAluno

Já descrita anteriormente.

## 5.0 – Alterar Aluno

Semelhante a Consultar Aluno.

## 6.0 – Excluir Aluno

Semelhante a Incluir Aluno.

## 7.0 – Incluir Empresa

### 7.1 – Classe Empresa

**Descrição:** classe que contém informações cadastrais de uma empresa.

**Tabela 8.** Descrição da classe Empresa.

| Atributos | Descrição                     | Tipo   |
|-----------|-------------------------------|--------|
| empresaID | Número do código da empresa . | int    |
| nome      | Nome.                         | String |

### 7.2 – Classe DAOEmpresa

**Descrição:** classe que faz o acesso à tabela EMPRESA do Banco de Dados.

**Tabela 9.** Descrição da classe DAOEmpresa.

| Métodos              | Descrição   |
|----------------------|---|
| Inserir(empresa)     | Inserir o cadastro de uma empresa no Banco de Dados.            |
| Alterar(empresa)     | Alterar o cadastro de uma empresa no Banco de Dados.            |
| Excluir(empresaID)   | Excluir o cadastro de uma empresa do Banco de Dados             |
| Consultar(empresaID) | Recupera o cadastro de uma empresa do Banco de Dados.           |
| Existe(empresa)      | Verifica se o cadastro de uma empresa existe no Banco de Dados. |

## 8.0 – Consultar Empresa

Semelhante a Incluir Empresa.

## 9.0 – Alterar Empresa

Semelhante a Incluir Empresa.

## 10.0 – Excluir Empresa

Semelhante a Incluir Empresa.

## 11.0 – Incluir Professor

### 11.1 – Classe Professor

**Descrição:** subclasse da classe Acadêmico que contém informações cadastrais de um professor. Não possui nenhum atributo adicional.

## 11.2 – Classe Usuário

Já descrita anteriormente.

## 11.3 – Classe DAOProfessor

**Descrição:** classe que faz o acesso à tabela PROFESSOR do Banco de Dados.

**Tabela 10.** Descrição da classe DAOProfessor.

| Métodos              | Descrição  |
|----------------------|--|
| Inserir(professor)   | Insere o cadastro de um professor no Banco de Dados.             |
| Alterar(professor)   | Altera o cadastro de um professor no Banco de Dados.             |
| Excluir(matricula)   | Exclui o cadastro de um professor do Banco de Dados              |
| Consultar(matricula) | Recupera o cadastro de um professor do Banco de Dados.           |
| Existe(professor)    | Verifica se o cadastro de um professor existe no Banco de Dados. |

## 11.4 – Classe DAOUsuario

Já descrita anteriormente.

## 12.0 – Consultar Professor

### 12.1 – Classe Professor

Já descrita anteriormente.

### 12.2 – Classe DAOProfessor

Já descrita anteriormente.

## 13.0 – Alterar Professor

Semelhante a Consultar Professor.

## 14.0 – Excluir Professor

Semelhante a Excluir Professor.

## 15.0 – Incluir Estágio

### 15.1 – Classe Estágio

**Descrição:** classe que contém informações cadastrais de um estágio.

**Tabela 11.** Descrição da classe Estágio.

| <b>Atributos</b>   | <b>Descrição</b>   | <b>Tipo</b> |
|--------------------|--|-------------|
| estagioid          | Número do código da empresa.   | int         |
| matriculaAluno     | Número de Matrícula do aluno na POLI.  | String      |
| matriculaProfessor | Número de Matrícula do professor na POLI.  | String      |
| empresald          | Número do código da empresa.   | int         |
| dataInicio         | Data do início do estágio.   | DateTime    |
| dataPrevisataFim   | Data prevista para o fim do estágio.   | DateTime    |
| dataFim            | Data do fim do estágio.  | DateTime    |
| situacao           | Situação do estágio: aprovado (A), cancelado (C) , matriculado (M) ou reprovado (R). | char        |
| dataReuniao1       | Data marcada para a 1ª reunião de acompanhamento                                     | DateTime    |
| dataReuniao2       | Data marcada para a 2ª reunião de acompanhamento                                     | DateTime    |
| dataReuniao3       | Data marcada para a 3ª reunião de acompanhamento                                     | DateTime    |
| dataReuniao4       | Data marcada para a 4ª reunião de acompanhamento                                     | DateTime    |
| situacaoReuniao1   | Situação da 1ª reunião de acompanhamento   | char        |
| situacaoReuniao2   | Situação da 2ª reunião de acompanhamento   | char        |
| situacaoReuniao3   | Situação da 3ª reunião de acompanhamento   | char        |
| situacaoReuniao4   | Situação da 4ª reunião de acompanhamento   | char        |

### 15.2 – Classe Aluno

Já descrita anteriormente.

### 15.3 – Classe Professor

Já descrita anteriormente.

### 15.4 – Classe Empresa

Já descrita anteriormente.

### 15.5 – Classe DAOEstagio

**Descrição:** classe que faz o acesso à tabela ESTAGIO do Banco de Dados.

**Tabela 12.** Descrição da classe DAOEstagio.

| <b>Métodos</b>       | <b>Descrição</b>   |
|----------------------|--|
| Inserir(estagio)     | Inserir o cadastro de um estágio no Banco de Dados.            |
| Alterar(estagio)     | Altera o cadastro de um estágio no Banco de Dados.             |
| Excluir(estagioid)   | Exclui o cadastro de um estágio do Banco de Dados              |
| Consultar(estagioid) | Recupera o cadastro de um estágio do Banco de Dados.           |
| Existe(estagio)      | Verifica se o cadastro de um estágio existe no Banco de Dados. |

### 15.6 – Classe DAOAluno

Já descrita anteriormente.

### 15.7 – Classe DAOProfessor

Já descrita anteriormente.

### 15.8 – Classe DAOEmpresa

Já descrita anteriormente.

## 16.0 – Consultar Estágio

Semelhante a Incluir Estágio.

## 17.0 – Alterar Estágio

Semelhante a Incluir Estágio.

## 18.0 – Excluir Estágio

### 18.1 – Classe Estágio

Já descrita anteriormente.

### 18.2 – Classe DAOEstagio

Já descrita anteriormente.

## 19.0 – Incluir Depoimento

### 19.1 – Classe Depoimento

**Descrição:** classe que contém informações cadastrais de um depoimento.

**Tabela 13.** Descrição da classe Depoimento.

| Atributos    | Descrição                                   | Tipo   |
|--------------|---|--------|
| depoimentoID | Número do código do depoimento.             | int    |
| assuntoID    | Número do código do assunto do depoimento . | int    |
| texto        | Texto do depoimento.                        | String |

### 19.2 – Classe Assunto

**Descrição:** classe que contém informações cadastrais de um assunto.

**Tabela 14.** Descrição da classe Assunto.

| Atributos | Descrição                    | Tipo   |
|-----------|------------------------------|--------|
| assuntoID | Número do código do assunto. | int    |
| descrição | Descrição do assunto.        | String |

### 19.3 – Classe DAODEpoimento

**Descrição:** classe que faz o acesso à tabela DEPOIMENTO do Banco de Dados.

**Tabela 15.** Descrição da classe DAODEpoimento.

| Métodos                 | Descrição   |
|-------------------------|---|
| Inserir(depoimento)     | Inserir o cadastro de um depoimento no Banco de Dados.            |
| Alterar(depoimento)     | Altera o cadastro de um usuário no Banco de Dados.                |
| Excluir(depoimentoID)   | Exclui o cadastro de um depoimento do Banco de Dados              |
| Consultar(depoimentoID) | Recupera o cadastro de um depoimento do Banco de Dados.           |
| Existe(depoimento)      | Verifica se o cadastro de um depoimento existe no Banco de Dados. |

### 19.4 – Classe DAOAssunto

**Descrição:** classe que faz o acesso à tabela ASSUNTO do Banco de Dados.

**Tabela 16.** Descrição da classe DAOAssunto.

| Métodos              | Descrição  |
|----------------------|--|
| Inserir(assunto)     | Inserir o cadastro de um assunto no Banco de Dados.            |
| Alterar(assunto)     | Altera o cadastro de um assunto no Banco de Dados.             |
| Excluir(assuntoID)   | Exclui o cadastro de um assunto do Banco de Dados              |
| Consultar(assuntoID) | Recupera o cadastro de um assunto do Banco de Dados.           |
| Existe(assunto)      | Verifica se o cadastro de um assunto existe no Banco de Dados. |

## 20.0 – Visualizar Fórum

Semelhante a Incluir Depoimento.

## 21.0 – Excluir Depoimento

### 21.1 – Classe Depoimento

Já descrita anteriormente.

### 21.2 – Classe DAODEpoimento

Já descrita anteriormente.

## 22.0 – Marcar Reunião



Semelhante a Alterar Estágio.

### **23.0 – Consultar Meus Estágios**

Semelhante a Consultar Estágio.

### **24.0 – Enviar Relatório de Estágio**

Semelhante a Consultar Aluno.

### **25.0 – Visualizar Relatório**

Semelhante a Consultar Estágio.



# Apêndice D

## Especificação de Casos de Uso

### 1.0 - Efetuar Login

**1.1 - Descrição:** Efetua a entrada do usuário no sistema.

#### 1.2 - Fluxo Básico de Eventos:

1. Na página de *login*, o coordenador insere seu *login* e a sua senha. Pressiona o botão Entrar.
2. O coordenador passa para a página principal do sistema.

#### 1.3 - Fluxos Alternativos de Eventos:

1. O coordenador não está cadastrado no sistema. Uma mensagem de erro é emitida.
2. O coordenador digita uma senha inválida. Uma mensagem de erro é emitida.

**1.4 - Pré-condições:** não há.

**1.5 - Pós-condições:** o usuário está logado no sistema ou uma mensagem de erro é emitida.

### 2.0 - Alterar Senha

**2.1 - Descrição:** altera a senha do usuário.

#### 2.2 - Fluxo Básico de Eventos:

1. O coordenador clica na opção Mudar Senha.
2. Na página de alteração de senha, o coordenador digita a senha antiga, a nova senha e também a confirmação da nova senha. Clica no botão Alterar.
3. Uma mensagem é exibida informando que a senha foi alterada com sucesso.

#### 2.3 - Fluxos Alternativos de Eventos:

1. O coordenador digita uma senha diferente da senha antiga. Uma mensagem de erro é emitida.
2. O coordenador digita uma nova senha diferente da senha de confirmação. Uma mensagem de erro é emitida.

**2.4 - Pré-condições:** o usuário precisa estar logado no sistema.

**2.5 - Pós-condições:** a senha do coordenador é alterada ou uma mensagem de erro é emitida.

### **3.0 – Incluir Aluno**

**3.1 - Descrição:** cadastra um aluno no sistema.

#### **3.2 - Fluxo Básico de Eventos:**

1. O usuário clica na opção Aluno no menu Cadastros.
2. Na página de inclusão de alunos, o usuário entra com os dados nos respectivos campos (matrícula, nome, e-mail e login). Clica no botão Incluir.
3. Uma mensagem é exibida informando que o aluno foi incluído com sucesso (com uma senha padrão).
4. Um e-mail é enviado para o aluno informando que o mesmo foi cadastrado no sistema e pedindo para que o acesse para alterar sua senha.

#### **3.3 - Fluxos Alternativos de Eventos:**

1. O usuário deixa de digitar um dado obrigatório. Uma mensagem de erro é emitida.
2. O usuário digita uma matrícula ou *login* já existente no cadastro de alunos ou usuários, respectivamente. Uma mensagem de erro é emitida.

**3.4 - Pré-condições:** o usuário precisa estar logado no sistema.

**3.5 - Pós-condições:** o aluno é cadastrado ou uma mensagem de erro é emitida.

### **4.0 – Consultar Aluno**

**4.1 - Descrição:** recupera o cadastro de um aluno.

#### **4.2 - Fluxo Básico de Eventos:**

1. O usuário clica na opção Aluno no menu Cadastros.
2. Na página de inclusão de alunos, o usuário clica no botão Consultar.
3. Na página de consulta de alunos, o coordenador entra com o nome (completo ou uma parte) do aluno. Clica no botão Consultar.
4. É exibido o resultado da consulta numa tabela logo abaixo do botão.
5. O usuário clica no *link* Detalhar ao lado do nome do aluno o qual deseja consultar.

6. Uma página é exibida com as informações do aluno.

#### **4.3 - Fluxos Alternativos de Eventos:**

1. O aluno não está cadastrado no sistema. Uma mensagem de erro é emitida.

#### **4.4 - Pré-condições:**

1. O usuário precisa estar logado no sistema.
2. O cadastro do aluno deve constar no sistema.

**4.5 - Pós-condições:** uma página é exibida com as informações do aluno ou uma mensagem de erro é emitida.

### **5.0 – Alterar Aluno**

**5.1 - Descrição:** altera o cadastro de um aluno.

#### **5.2 - Fluxo Básico de Eventos:**

1. Na página com as informações do aluno, o usuário entra com os novos dados nos respectivos campos. Clica em Alterar.
2. Uma mensagem é exibida informando que o aluno foi alterado com sucesso.

#### **5.3 - Fluxos Alternativos de Eventos:**

1. O aluno não está cadastrado no sistema. Uma mensagem de erro é emitida.
2. O usuário deixa de digitar um dado obrigatório (por exemplo, nome). Uma mensagem de erro é emitida.

#### **5.4 - Pré-condições:**

1. O usuário precisa estar logado no sistema.
2. O cadastro do aluno deve constar no sistema.

**5.5 - Pós-condições:** o cadastro do aluno é alterado ou uma mensagem de erro é emitida.

### **6.0 – Excluir Aluno**

**6.1 - Descrição:** exclui o cadastro de um aluno.

#### **6.2 - Fluxo Básico de Eventos:**

1. Na página com as informações do aluno, o coordenador clica no botão Excluir.

2. Uma mensagem é exibida informando que o aluno foi excluído com sucesso.

### **6.3 - Fluxos Alternativos de Eventos:**

1. O aluno não está cadastrado no sistema. Uma mensagem de erro é emitida.

### **6.4 - Pré-condições:**

1. O usuário precisa estar logado no sistema.
2. O cadastro do aluno deve constar no sistema.

**6.5 - Pós-condições:** o cadastro do aluno é excluído ou uma mensagem de erro é emitida.

## **7.0 – Incluir Empresa**

**7.1 - Descrição:** cadastra uma empresa no sistema.

### **7.2 - Fluxo Básico de Eventos:**

1. O usuário clica na opção Empresa no menu Cadastros.
2. Na página de inclusão de empresas, o usuário entra com o nome da empresa. Clica no botão Incluir.
3. Uma mensagem é exibida informando que a empresa foi incluída.

### **7.3 - Fluxos Alternativos de Eventos:**

1. O usuário deixa de digitar o nome da empresa. Uma mensagem de erro é emitida.
2. O usuário digita o nome de uma empresa já existente no cadastro de empresas. Uma mensagem de erro é emitida.

**7.4 - Pré-condições:** o usuário precisa estar logado no sistema.

**7.5 - Pós-condições:** o cadastro da empresa é incluído ou uma mensagem de erro é emitida.

## **8.0 – Consultar Empresa**

Semelhante a Consultar Aluno.

## **9.0 – Alterar Empresa**

Semelhante a Alterar Aluno.

## **10.0 – Excluir Empresa**

Semelhante a Excluir Aluno.

## **11.0 – Incluir Professor**

Semelhante a Incluir Empresa.

## **12.0 – Consultar Professor**

Semelhante a Consultar Aluno.

## **13.0 – Alterar Professor**

Semelhante a Alterar Aluno.

## **14.0 – Excluir Professor**

Semelhante a Excluir Aluno.

## **15.0 – Incluir Estágio**

**15.1 - Descrição:** cadastra um estágio no sistema.

### **15.2 - Fluxo Básico de Eventos:**

1. O usuário clica na opção Estágio no menu Cadastros.
2. Na página de inclusão de estágios, o usuário escolhe um aluno, uma empresa e um professor orientador, digita a data de início e data do fim do mesmo. Clica no botão Incluir.
3. Uma mensagem é exibida informando que o estágio foi incluído com sucesso.

### **15.3 - Fluxos Alternativos de Eventos:**

1. O usuário deixa de digitar um dado obrigatório. Uma mensagem de erro é emitida.
2. O usuário digita um dado inválido (por exemplo, uma data no formato errado). Uma mensagem de erro é emitida.
3. O usuário digita um estágio já existente no cadastro. Uma mensagem de erro é emitida.

#### **15.4 - Pré-condições:**

1. O usuário precisa estar logado no sistema.
2. O aluno escolhido deve estar cadastrado no sistema.
3. O professor escolhido deve estar cadastrado no sistema.
4. A empresa escolhida deve estar cadastrada no sistema.

**15.5 - Pós-condições:** O estágio é cadastrado ou uma mensagem de erro é emitida.

### **16.0 – Consultar Estágio**

**16.1 - Descrição:** recupera o cadastro de um estágio.

#### **16.2 - Fluxo Básico de Eventos:**

1. O usuário clica na opção Estágio no menu Cadastros.
2. Na página de inclusão de estágios, o usuário clica no botão Consultar.
3. Na página de consulta de estágios, o usuário entra com o nome (completo ou uma parte) do aluno, da empresa e/ou do professor. Clica no botão Consultar.
4. É exibido o resultado da consulta numa tabela logo abaixo do botão.
5. O usuário clica no *link* Detalhar ao lado do estágio o qual deseja consultar.
6. Uma página é exibida com as informações do estágio.

#### **16.3 - Fluxos Alternativos de Eventos:**

1. O estágio não está cadastrado no sistema. Uma mensagem de erro é emitida.

#### **16.4 - Pré-condições:**

1. O usuário precisa estar logado no sistema.
2. O cadastro do estágio deve constar no sistema.

**16.5 - Pós-condições:** uma página é exibida com as informações do estágio ou uma mensagem de erro é emitida.

### **17.0 – Alterar Estágio**

**17.1 - Descrição:** altera o cadastro de um estágio.

#### **17.2 - Fluxo Básico de Eventos:**

1. Na página com as informações do estágio, o usuário entra com os novos dados nos respectivos campos. Clica em Alterar.
2. Uma mensagem é exibida informando que o estágio foi alterado com sucesso.

#### **17.3 - Fluxos Alternativos de Eventos:**

1. O estágio não está cadastrado no sistema. Uma mensagem de erro é emitida.
2. O usuário digita um dado inválido (por exemplo, uma data no formato errado). Uma mensagem de erro é emitida.
3. O usuário deixa de digitar um dado obrigatório. Uma mensagem de erro é emitida.

#### **17.4 - Pré-condições:**

1. O usuário precisa estar logado no sistema.
2. O cadastro do estágio deve constar no sistema.

**17.5 - Pós-condições:** o cadastro do estágio é alterado ou uma mensagem de erro é emitida.

### **18.0 – Excluir Estágio**

Semelhante a Excluir Aluno.

### **19.0 – Marcar Reuniões**

**19.1 - Descrição:** marca as datas das quatro reuniões de acompanhamento de um estágio e define as situações dos mesmos.

#### **19.2 - Fluxo Básico de Eventos:**

1. Na página de marcação de reuniões de acompanhamento, o usuário digita a data de uma ou mais reuniões de acompanhamento e/ou suas situações. Clica em Alterar.
2. Uma mensagem é exibida informando que as reuniões foram marcadas com sucesso.

#### **19.3 - Fluxos Alternativos de Eventos:**

1. O estágio não está cadastrado no sistema. Uma mensagem de erro é emitida.
2. O usuário digita um dado inválido (por exemplo, uma data no formato errado). Uma mensagem de erro é emitida.
3. O usuário deixa de digitar um dado obrigatório. Uma mensagem de erro é emitida.

#### **19.4 - Pré-condições:**

1. O usuário precisa estar logado no sistema.
2. O cadastro do estágio deve constar no sistema.

**19.5 - Pós-condições:** as reuniões são marcadas ou uma mensagem de erro é emitida.

## **20.0 – Consultar Meus Estágios**

**20.1 - Descrição:** recupera o cadastro de um estágio do usuário, sendo ele um aluno ou professor.

### **20.2 - Fluxo Básico de Eventos:**

1. O usuário clica na opção Estágio no menu Cadastros.
2. Na página de inclusão de estágios, o usuário clica no botão Consultar.
3. Uma página é exibida com os estágios do usuário numa tabela.
4. O usuário clica no *link* Detalhar ao lado do estágio o qual deseja consultar.
5. Uma página é exibida com as informações do estágio, inclusive com as datas das reuniões marcadas e as presenças nas mesmas.

### **20.3 - Fluxos Alternativos de Eventos:**

1. O estágio não está cadastrado no sistema. Uma mensagem de erro é emitida.

### **20.4 - Pré-condições:**

1. O usuário precisa estar logado no sistema.
2. O cadastro do estágio deve constar no sistema.

**20.5 - Pós-condições:** uma página é exibida com as informações do estágio do usuário ou uma mensagem de erro é emitida.

## **21.0 – Enviar Relatório**

**21.1 - Descrição:** envia um arquivo PDF contendo o Relatório de Estágio de um aluno.

### **21.2 - Fluxo Básico de Eventos:**

1. O usuário clica na opção Enviar Relatório no menu Opções.
2. Uma página é exibida. O usuário clica no botão Selecionar.
3. Abre-se uma janela onde o usuário poderá navegar pelos arquivos do seu computador.
4. O usuário seleciona o arquivo de texto em formato PDF que contém seu Relatório de



Estágio e clica no botão OK.

5. O nome do arquivo e seu caminho aparecem na caixa de texto Arquivo. O usuário clica no botão Enviar.
6. Uma mensagem é exibida informando que o *upload* do arquivo para o sistema foi realizado com sucesso.

### **22.3 - Fluxos Alternativos de Eventos:**

1. O usuário clica no botão Enviar sem que tenha selecionado nenhum arquivo. Uma mensagem de erro é emitida.
2. O usuário (aluno) ainda não foi aprovado no Estágio Curricular e, portanto, ainda não pode enviar seu relatório. Uma mensagem de erro é emitida.
3. O usuário seleciona um arquivo que não existe. Uma mensagem de erro é emitida.
4. O usuário seleciona um arquivo com extensão inválida. Uma mensagem de erro é emitida.

### **21.4 - Pré-condições:**

1. O usuário precisa estar logado no sistema.
2. O usuário (aluno) precisa ter sido aprovado no Estágio Curricular.
3. O arquivo deve existir no computador do usuário.
4. O arquivo precisa ter a extensão PDF.

**21.5 - Pós-condições:** o *upload* do arquivo para o sistema é realizado ou uma mensagem de erro é emitida.

## **22.0 – Incluir Depoimento**

**22.1 - Descrição:** envia um arquivo PDF contendo o Relatório de Estágio de um aluno.

### **22.2 - Fluxo Básico de Eventos:**

1. O usuário clica na opção Incluir Depoimento no menu Cadastros.
2. Na página de inclusão de depoimentos, o usuário escolhe o assunto e digita o depoimento em no espaço reservado. Clica no botão Incluir.
3. Uma mensagem é exibida informando que o depoimento foi incluído com sucesso.

### **22.3 - Fluxos Alternativos de Eventos:**

1. O usuário deixa de escolher um assunto ou de digitar o depoimento. Uma mensagem de erro é emitida.

#### **22.4 - Pré-condições:**

1. O usuário precisa estar logado no sistema.
2. O assunto escolhido deve estar cadastrado no sistema.

**22.5 - Pós-condições:** um depoimento é incluído ou uma mensagem de erro é emitida.

### **23.0 – Excluir Depoimento**

Semelhante a Excluir Aluno.

### **24.0 – Visualizar Fórum**

**24.1 - Descrição:** exibe o fórum de depoimento dos alunos em Estágio Curricular.

#### **24.2 - Fluxo Básico de Eventos:**

1. O coordenador clica em Fórum no menu Opções.
2. É exibida uma página contendo os assuntos do fórum.
3. O coordenador seleciona um assunto desejado.
4. É exibida uma página contendo os depoimentos dos alunos dispostos em ordem cronológica sobre o determinado tópico.

#### **24.3 - Fluxos Alternativos de Eventos:**

Não há.

#### **24.4 - Pré-condições:**

1. O usuário precisa estar logado no sistema.

**24.5 - Pós-condições:** uma página contendo o fórum com os depoimentos dos alunos é exibida.

### **25.0 – Visualizar Relatório**

**25.1 - Descrição:** exibe um relatório administrativo da Coordenação de Estágio do DSC.

#### **25.2 - Fluxo Básico de Eventos:**

1. O usuário clica em Relatórios no menu Opções.
2. É exibida uma página contendo as opções de relatório disponibilizadas pelo sistema.

3. O usuário seleciona um tipo de relatório.
4. É exibida uma página com o relatório administrativo solicitado.

**25.3 - Fluxos Alternativos de Eventos:**

Não há.

**25.4 - Pré-condições:**

1. O usuário precisa estar logado no sistema.

**25.5 - Pós-condições:** um relatório administrativo é exibido.

# Apêndice E

## Diagramas

### 1.0 – Diagrama de Classes

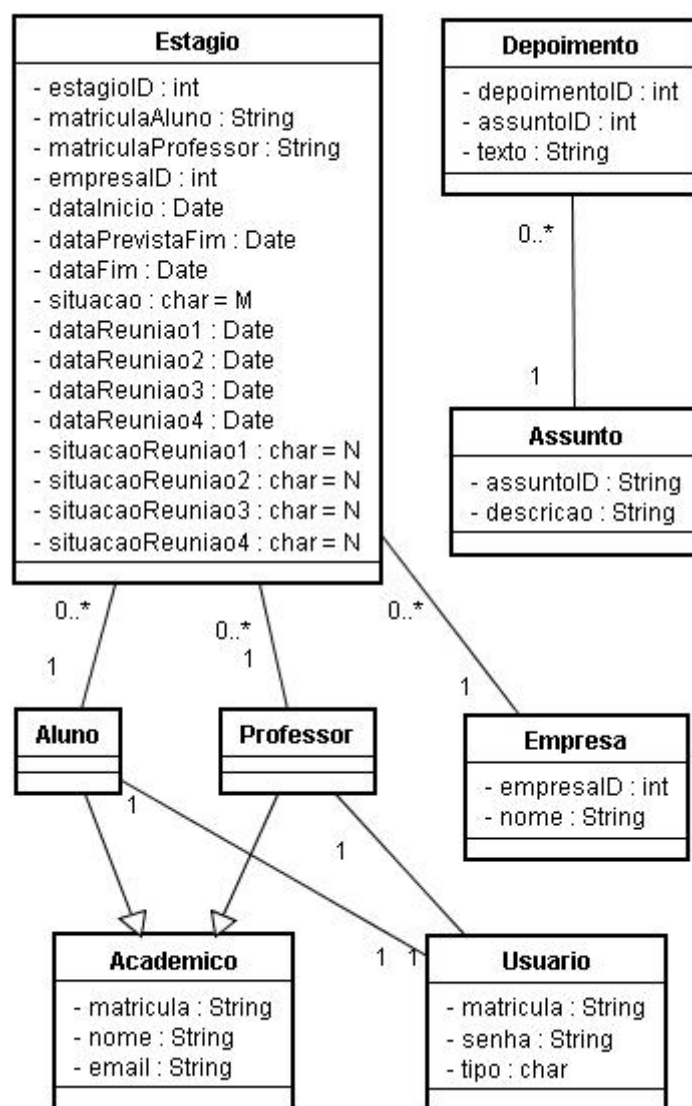


Figura 14. Diagrama de Classes.

### 2.0 – Diagrama de Casos de Uso

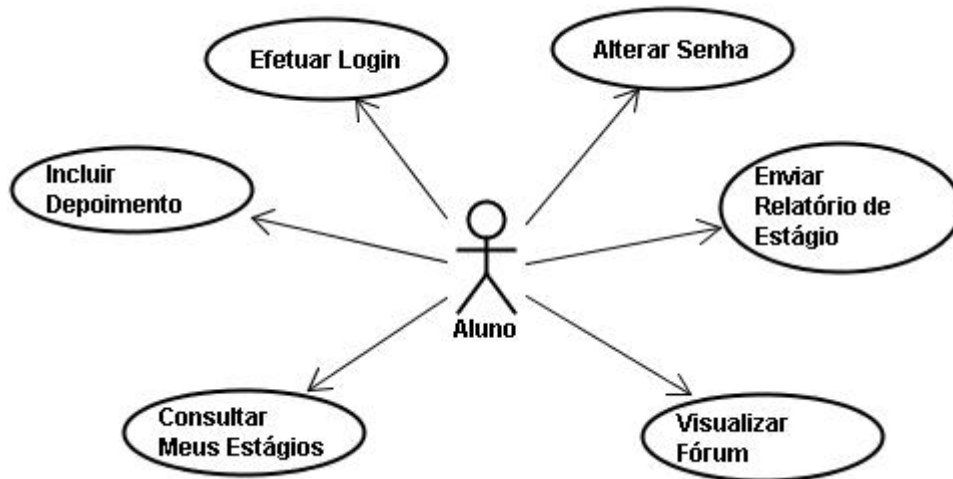


Figura 15. Diagrama de Casos de Uso de Aluno.

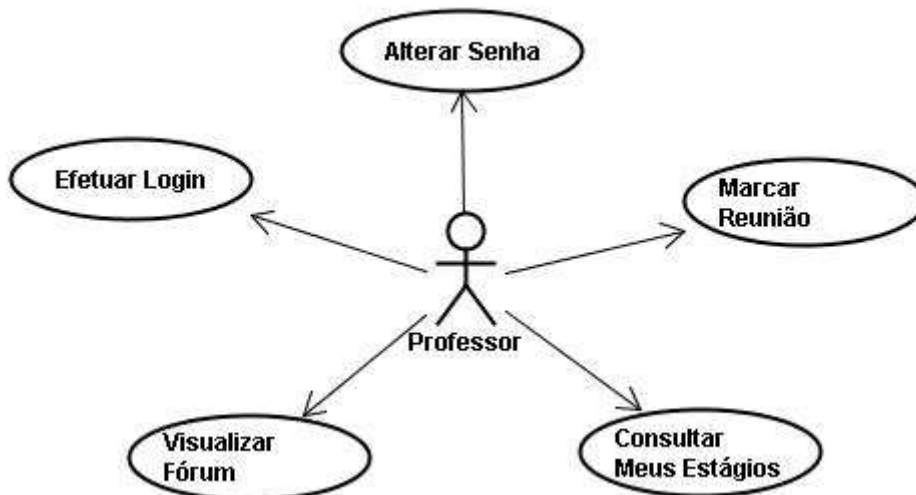


Figura 16. Diagrama de Casos de Uso de Professor.

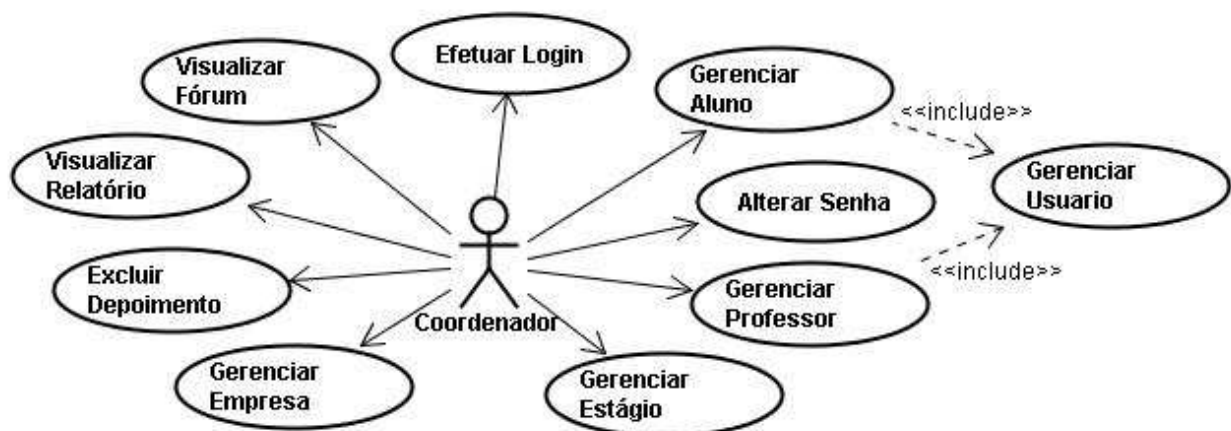


Figura 17. Diagrama de Casos de Uso de Coordenador.

**Observação:** os casos de uso Gerenciar englobam os casos de uso Incluir, Consultar, Alterar e Excluir.

## 2.0 – Diagrama do Banco de Dados



**Figura 18.** Diagrama do Banco de Dados.