

## LEVANTAMENTO DE REQUISITOS E ANÁLISE PARA UM SISTEMA DE CONTROLE DE ACADEMIA

**Cristhopher Liberatto Rodrigues de Frias**

Graduando em Tecnologia em Análise e Desenvolvimento de Sistemas,  
Faculdades Integradas de Três Lagoas – FITL/AEMS

**David Henrique da Silva**

Graduando em Tecnologia em Análise e Desenvolvimento de Sistemas,  
Faculdades Integradas de Três Lagoas – FITL/AEMS

**Isabella da Silva Gomes Carneiro**

Graduanda em Tecnologia em Análise e Desenvolvimento de Sistemas,  
Faculdades Integradas de Três Lagoas – FITL/AEMS

**Rodrigo de Carvalho Ribeiro**

Especialista em Redes de Computadores;  
Docente das Faculdades Integradas de Três Lagoas – FITL/AEMS

### RESUMO

O presente estudo foi desenvolvido para uma empresa no ramo de atividades físicas, e tem como objetivo desenvolver um *software*, de modo a automatizar parte de sua forma de trabalho, que atualmente conta com a utilização de fichas de papel para armazenar os dados. Como por exemplo, os cadastros e matrícula do aluno. A metodologia foi baseada em entrevistas com o proprietário da empresa e pesquisas bibliográficas. Por meio dessa entrevista com o proprietário, notou-se a falta que um sistema automatizado trás para o dia a dia. Com a academia tendo parte do seu trabalho automatizado pelo sistema, acredita-se que a empresa terá uma maior agilidade nos cadastros e também na emissão e consulta de relatórios. Dessa forma acredita-se também que a empresa contará com uma maior segurança no armazenamento dos seus dados.

**PALAVRAS-CHAVE:** levantamento de requisitos; análise; diagrama de classes; atores; diagrama de casos de uso.

### INTRODUÇÃO

Nesse artigo serão mostrados alguns dos processos de grande importância para o desenvolvimento de um *software*, nesse caso um sistema de controle de academia, sendo o Levantamento de Requisitos e a Análise. A academia em que foi pesquisado esse tema não utiliza nenhum sistema automatizado desde a sua inauguração, portanto a intenção deste trabalho será mostrar alguns processos para que seja possível desenvolver um *software* para essa empresa.

Com o crescimento dos serviços prestados por academias de exercícios físicos cresce também a concorrência, e um sistema automatizado é importante para

que se destaque entre tantas outras. Acredita-se que algumas das vantagens de se utilizar um sistema automatizado são a facilidade e agilidade de executar as tarefas da empresa. Tendo em vista o sistema em fichas manuais utilizado, observou-se que os cadastros poderiam simplesmente se perder, e dessa forma se tornar uma empresa pouco confiável. Assim sendo, a implementação desse sistema automatizado se dará também ao controle de parte da administração e dos problemas que prejudicam a agilidade, organização e segurança da empresa, facilitando então o controle dos registros dos alunos e agilidade nos serviços prestados.

Na seção 2 será abordado o levantamento de requisitos onde mostrará os problemas existentes na empresa, os desejos do cliente quanto ao sistema proposto e o impacto desse sistema em sua empresa. Na seção 3 será apresentada a análise, onde será abordado o diagrama de classes, os atores do sistema e o diagrama de casos de uso.

## 2 LEVANTAMENTO DE REQUISITOS

No levantamento de requisitos será abordada a descrição do sistema atual, que mostrará a maneira e os recursos que a empresa utiliza para a atual forma de trabalho, os problemas existentes em relação a essa forma de trabalho atual e os desejos do cliente quanto ao sistema automatizado e suas funções.

Entende-se por levantamento de requisitos uma forma de suprir e reorganizar uma devida empresa em cima de sua forma de trabalho, isso é possível diante de uma boa análise, ou seja, quando consegue ter regras. Compreender que tudo isso garante a entrada de recursos, custo e benefícios que esse *software* ira ter na empresa, assim como o andamento do processo que se da à organização da empresa e o resultado final gerado em cima desse sistema.

Os requisitos de sistema são descrições mais detalhadas dos requisitos do usuário. Eles podem servir como base para um contrato destinado à implementação do sistema e, portanto, devem ser uma especificação completa e consistente de todo o sistema. Eles são utilizados pelos engenheiros de *software* como ponto de partida para o projeto de sistema (SOMMERVILLE, 2003, p.91).

Diante dessas características o intuito deste artigo é mostrar que esse método de levantamento e análise requer a avaliação e compreensão do profissional

e da empresa. Para haver um bom funcionamento do sistema de modo que ira ter um valor positivo a empresa, é necessário informações suficientes, porém não só embasada em entrevista mais sim em processos subdividas em categorias que supram a necessidade para que se crie um bom sistema de modo que satisfaça a empresa. Isto pode ser feito através de intervenções em cima do sistema atual, para que assim haja um maior desenvolvimento sobre a empresa, e posteriormente esses desenvolvimentos sejam atribuídos de forma a melhorar a prática do mesmo do serviço prestado e uma melhoria da qualidade dos alunos.

O fluxo de Requisitos reúne as atividades que visam a obter o enunciado completo, claro e preciso dos requisitos de um produto de *software*, Esses requisitos devem ser levantados pela equipe do projeto, em conjunto com representantes do cliente, usuários chaves e outros especialistas da área de aplicação (PAULA FILHO, 2000, p.87).

Segundo Sommerville (2003, p.104) no levantamento e a análise de requisitos, os membros da equipe técnica de desenvolvimento de *software* trabalham com o cliente e os usuários finais do sistema para descobrir mais informações sobre o domínio da aplicação, que serviços o sistema deve fornecer, o desempenho exigido do sistema, as restrições de hardware e assim por diante.

## 2.1 Descrição do Sistema Atual

Nesse artigo, a empresa que segue em analise para implementação de um sistema automatizado, tem um método de trabalho onde a eficiência não está atingindo o objetivo do cliente. O sistema atual para os cadastros, pagamentos e demais registros dos alunos trata-se por meio de fichas de papel, onde seu preenchimento acontece de forma manual pela recepcionista, ou seja, são registros manuscritos. Esses registros são colocados em uma caixa de arquivo e organizados pelas iniciais dos nomes dos alunos.

Observa-se na Figura 1 o modelo de ficha de papel utilizado pela empresa, nela contem as informações pessoais do aluno, assim como o pagamento para utilização dos serviços oferecidos. Essas fichas depois de arquivadas ficam de fácil acesso para todos da empresa, localizadas no balcão da recepção.

**Figura 1.** Ficha de cadastro de aluno.

The image shows a handwritten form on lined paper. The fields and their handwritten entries are as follows:

- Nome: [Handwritten name]
- Endereço: [Handwritten address]
- Idade: [Handwritten age]
- Sexo: [Handwritten sex]
- Objetivo: [Handwritten objective]
- Endereço: [Handwritten address]
- Tel: [Handwritten phone number]
- Email: [Handwritten email]
- Data de nascimento: [Handwritten date of birth]
- Pagamentos: [Handwritten payment information]

**Fonte:** Extraído da empresa sendo analisada. Editado pelos autores.

## 2.2 Problemas Existentes

Tendo conhecimento do sistema atual de trabalho da empresa, tanto para os cadastros quanto pagamentos ou quaisquer outros registros dos alunos, são feitos a partir das fichas de papel, ou seja, registros manuscritos. Dessa forma acredita-se que a empresa não transmite segurança para os alunos em relação a seus dados registrados e arquivados, como também a falta na agilidade para atendê-los, que por conter muitos registros de alunos, pode acarretar demora em uma futura busca por uma ficha específica, pois muitas das vezes é necessário que se faça alguma alteração nesses registros, como por exemplo, os pagamentos que são realizados mensalmente.

Outro problema também é referente às pessoas que podem ter acesso a essas fichas depois de arquivadas, que depois do processo de arquivo ficam armazenadas no balcão da recepção, ficando de fácil acesso para todos da empresa, sem nenhum tipo de segurança e restrição.

## 2.3 Desejos do Cliente

Tendo em vista que hoje há uma competitividade no mercado de trabalho, as empresas estão investindo em melhorar e ampliar sua potencialidade e mostrar

segurança para o cliente, diante disso se faz necessário, mudanças nos sistemas atuais utilizados.

A implantação de um sistema automatizado onde somente o gerente terá acesso total quanto às funcionalidades, tem como objetivo principal a segurança no armazenamento dos dados do aluno e também na privacidade de alguns dados, como por exemplo, um relatório de pagamentos de alunos.

O cliente atualmente trabalha com um sistema pré-pago para a utilização dos serviços oferecidos por sua empresa, esse sistema funciona basicamente como uma recarga de créditos para um telefone celular. Dessa forma o aluno antes de começar a utilizar os serviços deve fazer o pagamento da mensalidade, sendo assim, o vencimento será no mesmo dia do mês seguinte. Não existe, portanto uma mensalidade fixa e obrigatória para o aluno, e o cliente deseja manter esse sistema de trabalho. Ele propôs que o sistema tenha funções práticas e uma *interface* de fácil entendimento para o usuário final, que no caso será ele como gerente e a recepcionista.

### 3 ANÁLISE

Na análise será mostrado um dos diagramas mais importantes para a modelagem de um sistema automatizado, o Diagrama de Classes. A lista de eventos que o sistema irá conter. Os atores do sistema mostrando cada usuário que terá interação com o sistema. E o caso de uso que mostra ações que um ou mais atores realizam no sistema.

De acordo com Paula Filho (2000, p.120):

O modelo de análise deve conter os detalhes necessários para servir de base para o desenho do produto, mas deve-se evitar a inclusão de detalhes que pertençam ao domínio da implementação e não do problema.

Análise consiste em reconhecimento do problema, ou seja, conhecer a realidade da empresa, fazendo assim uma boa síntese dos requisitos levantados para que se faça uma modelagem do sistema que irá ser utilizado, especificando e revisando toda linha de desenvolvimento que permitirá uma solução e ampliação da empresa.

A análise de requisitos resulta na especificação das características operacionais do *software*; indica a *interface* do *software* com outros elementos do sistema e estabelece restrições a que o *software* (algumas vezes chamado de analista ou modelador nesse papel) elaborar requisitos e construir modelos que descrevam cenários de usuário, atividades funcionais, classes de problemas e seus relacionamentos, comportamento do sistema e das classes, e fluxo dos dados à medida que são transformados (PRESSMAN, 2006, p.145).

### 3.1 Diagrama de Classes

O diagrama de classes é o mais importante diagrama da UML (*Unified Modeling Language*), ele está no centro da sua arquitetura e a partir desse diagrama outros diagramas são elaborados. O diagrama de classes é uma importante ferramenta para a documentação de um sistema ou produto de *software*.

Segundo Booch, Rumbaugh e Jacobson (2000), os diagramas de classes são os diagramas encontrados com maior frequência na modelagem de sistemas orientados a objetos. Um diagrama de classes mostra um conjunto de classes, *interfaces* e colaborações e seus relacionamentos, os diagramas de classes são importantes não só para a visualização, a especificação e a documentação de modelos estruturais, mas também para a construção de sistemas executáveis por intermédio de engenharia de produção e reversa.

O diagrama de classes é uma das técnicas mais utilizadas no desenvolvimento orientado a objetos. Esse diagrama é uma descrição formal da estrutura de objetos em um sistema. Para cada tipo de objeto, o diagrama de classes descreve a sua identidade, os seus relacionamentos, os seus atributos e a suas operações.

O diagrama de classes não é somente amplamente usado, mas também o receptáculo para o maior escopo de conceitos de modelagem. Embora os elementos básicos sejam necessários para todos, os conceitos avançados são utilizados com menos frequência. Um diagrama de classes descreve os tipos de objetos no sistema e os vários tipos de relacionamentos estáticos que existem entre eles (FOWLER, 2000, p.57).

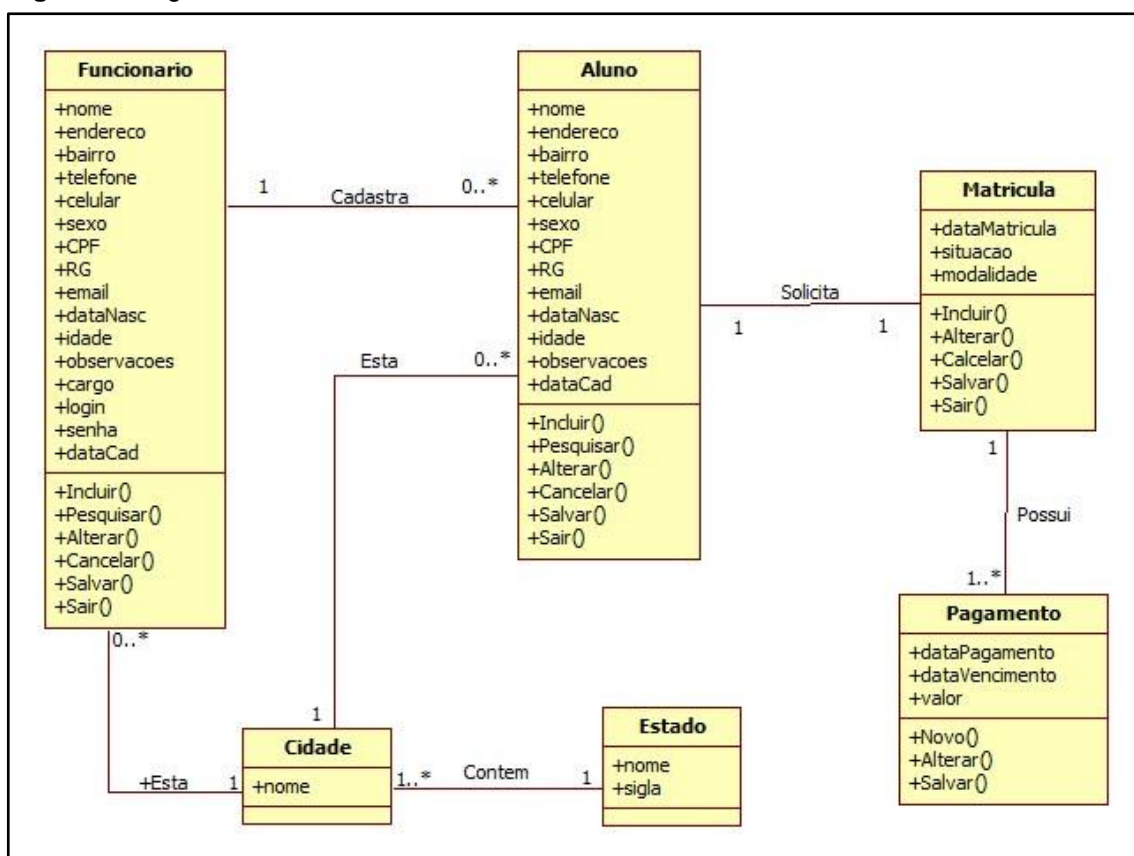
Segundo Furlan (1998, p.180), “a modelagem de caso de uso é uma técnica utilizada para descrever a funcionalidade de um sistema através de atores externos interagindo em casos de uso”. Na Figura 2 é apresentado o diagrama de classes, suas relações e cardinalidades. Na classe Funcionário estão contidas todas as informações necessárias para o cadastro do funcionário e essa classe recebe interações de outras duas classes, a cidade e estado, essas duas classes não terão



uma tela de cadastro próprio, pois já terão cidades e estados predefinidos no banco de dados.

A classe Aluno também recebe as interações das classes Cidade e Estados e têm as mesmas informações contidas na classe Funcionário com exceção de cargo, *login* e senha, que são informações específicas do funcionário, onde cargo irá definir as restrições de acesso as funções do sistema, e o *login* e senha para que consiga o acesso. A classe Aluno tem interação com a classe Matrícula, após o cadastro de um aluno é que poderá acontecer a matrícula, e a classe Matrícula é ligada a classe Pagamento, pois no ato da matrícula são requeridos os dados do primeiro pagamento, para que o aluno então possa começar a utilizar os serviços oferecidos pela academia.

**Figura 2.** Diagrama de classes.



**Fonte:** Elaborado pelos autores.

### 3.2 Lista de Eventos

Seguindo os desejos do cliente, abaixo são listados todos eventos que o sistema disponibilizará. Apesar de o sistema completo conter todas essas funções, nesse artigo em específico será mostrado apenas o “Evento 04 – Matricular Aluno”,

que é uma das mais importantes funções do sistema. (i) Evento 01 - Entrar no Sistema; (ii) Evento 02 - Cadastrar Funcionário; (iii) Evento 03 - Cadastrar Aluno; (iv) Evento 04 - Matricular Aluno; (v) Evento 05 - Realizar Pagamento; (vi) Evento 06 - Gerar Relatório de Alunos; (vii) Evento 07 - Gerar Relatório de Funcionários; (viii) Evento 08 - Gerar Relatório de Pagamentos; (iv) Evento 09 - Gerar Relatório de Inatividades.

### 3.3 Definições dos Atores

Os atores são pessoas ou sistema que interagem diretamente com o sistema. É necessário ter em mente a diferença entre os reais usuários do sistema e suas respectivas categorias. Assim, o nome ator é aplicado para uma categoria de usuários, definido o papel que esta categoria desempenha no sistema.

Um ator é um papel que um usuário desempenha em relação ao sistema. Os atores desempenham os casos de uso. Os atores não precisam ser humanos, embora sejam representados como bonecos humanos nos diagramas de caso de uso. Um ator também pode ser um sistema externo que necessita de alguma informação do sistema atual (FOWLER, 2000, p.52).

Um ator pode invocar vários casos de uso e um caso de uso pode ser invocado por vários atores, que deverão ser detalhados através de uma pequena descrição de forma a assegurar uma correta compreensão do significado do ator por todos os elementos envolvidos no processo de desenvolvimento. De acordo com Furlan (1998, p.175), “o ator comunica-se com o sistema através do envio e recebimento de mensagens, sendo que um caso de uso é sempre iniciado a partir do momento que um ator envia sua mensagem (estímulo)”.

A Figura 3 apresenta os atores que participarão dos eventos durante a utilização do sistema. O Gerente sendo o dono e responsável geral da empresa. A Recepcionista sendo a responsável por utilizar o sistema, mas com algumas restrições em seu acesso. E o Funcionário, esse ator em específico será utilizado quando, tanto o Gerente quanto a Recepcionistas tiverem as mesmas permissões do sistema e puderem fazer as mesmas ações.



**Figura 3.** Atores do sistema.



Fonte: Elaborado pelos autores.

### 3.4 Diagramas de Casos de Uso

O modelo de casos de uso permite capturar os requisitos de um sistema através do detalhe de todos os cenários que os usuários podem realizar. Os casos de uso, mais que iniciar a modelagem de requisitos de um sistema, dirigem todo o processo do desenvolvimento. O diagrama ajuda o analista e o usuário a entenderem melhor todas as funcionalidades do sistema.

Segundo Sommerville (2003), os casos de uso são técnicas baseadas em cenários para obtenção de requisitos. Eles se tornaram, atualmente, uma característica fundamental da notação em UML para descrever modelos de sistemas orientados a objeto.

Os casos de usos podem ser aplicados para captar o comportamento pretendido do sistema que está sendo desenvolvido, sem ser necessário especificar como esse comportamento é implementado. Os casos de uso fornecem uma maneira para os desenvolvedores chegarem a uma compreensão comum com os usuários finais do sistema e com os especialistas do domínio (BOOCH; RUMBAUGH; JACOBSON, 2000, p.217).

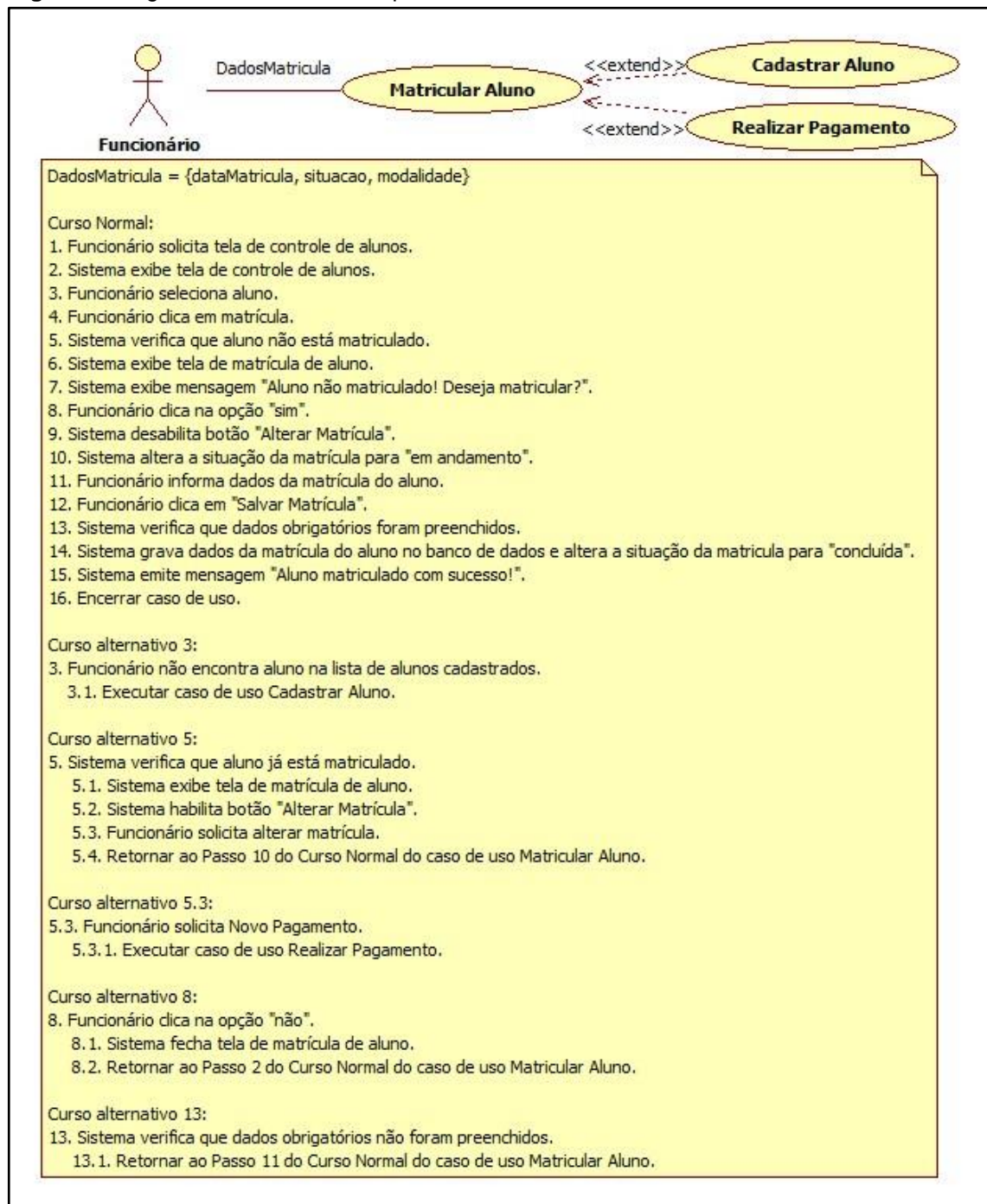
Segundo Furlan (1998), um caso de uso é uma sequência de ações que um ou mais atores realizam num sistema de modo a obterem um resultado particular. O diagrama ajuda o analista e o usuário a entenderem melhor todas as funcionalidades do sistema. A modelagem de caso de uso é uma técnica utilizada para descrever a funcionalidade de um sistema através de atores externos interagindo em casos de uso.

#### 3.4.1 Evento 04 – Matricular Aluno

Na Figura 4 mostra a interação do funcionário com a tela de matrícula de aluno, os “Dados Matrícula” exibidos na figura é referente aos dados que o

funcionário irá preencher na tela em questão para que a matrícula seja concluída, o diagrama mostra também os “extends”, que nesse caso em específico podem ser entendidos como uma extensão do Caso de Uso Matricular Aluno, ou seja, tem a probabilidade de acontecer do funcionário ter que executar outro Caso de Uso para que possa dar continuidade na Matrícula.

**Figura 4.** Diagrama de casos de uso para matricular aluno.



**Fonte:** Elaborado pelos autores.

Nesse exemplo em questão, o “extend” pode chamar o Caso de Uso Cadastrar Aluno caso o funcionário não o encontre na lista de alunos cadastrados, por isso ele terá que cadastrar o aluno primeiro para que após esse procedimento consiga realizar o Caso de Uso Matricular Aluno.

O curso normal nesse exemplo mostra os passos que o funcionário irá realizar para que a matrícula do aluno seja concluída com sucesso, o curso alternativo 3 irá acontecer caso o funcionário não encontre o aluno desejado na lista de alunos cadastrados para poder fazer sua matrícula, portanto é chamado o Caso de Uso Cadastrar Aluno para poder prosseguir. Já o curso alternativo 5 acontece quando o sistema verifica que o aluno já está matriculado e com isso o sistema exibe a tela de matrícula com suas funções. O curso alternativo 5.3 mostra também um “extend” do caso de uso, e nesse caso ele é executado quando o funcionário solicita um Novo Pagamento para algum aluno, e isso fará com que o Caso de Uso Realizar Pagamento entre em ação. No curso alternativo 8 mostra o funcionário optando por não desejar matricular um aluno novo, e com isso a tela de matrícula é finalizada. E o curso alternativo 13 mostra quando um funcionário não digita os dados nos campos obrigatórios que o sistema solicita.

#### **4 CONSIDERAÇÕES FINAIS**

Os objetivos aqui apresentados serviram de exemplo para a elaboração de um sistema, diante da vivência da empresa possibilitando sua implantação. Para a implementação foi essencial que alguns pontos relevantes fossem observados, por exemplo, mudança sistema utilizado, orientação a processos e treinamento. Muitas empresas tiveram crescimento diante do avanço tecnológico, bem como à confiabilidade e o crescimento no mercado de trabalho. O uso desse novo sistema irá proporcionar um auxílio ao processo de gestão bem como a ação gerencial do serviço prestado. O método do desenvolvimento permitiu apresentar para uma maneira de buscando, traduzir e transmitir as necessidades e desejos do cliente garantindo assim uma qualidade e necessidade dos desejos do cliente. O sistema se mostra adequado no planejamento e controle da ações/organizações que a empresa procura. A análise detalhada no momento de seleção do sistema, com o intuito de se trazer segurança, foi essencial para o sucesso do trabalho apresentado.

## 5 REFERÊNCIAS

BOOCH, G.; RUMBAUGH, J.; JACOBSON, I. UML – Guia do Usuário. 12. ed., Rio de Janeiro, 2000.

FOWLER, M. UML Essencial: Um Breve Guia Para a Linguagem Padrão de Modelagem de Objetos. 2. ed., Porto Alegre: Bookman, 2000.

FURLAN, J. Modelagem de Objetos através da UML. São Paulo: Makron Books, 1998.

PAULA FILHO, W. Engenharia de Software. 2. ed., Rio de Janeiro: LTC, 2000.

PRESSMAN, R. Engenharia de Software - Uma Abordagem Prática. 6. ed., São Paulo: MacGraw-Hill, 2006.

SOMMERVILLE, I. Engenharia de Software. 6. ed., São Paulo: Addison Wesley, 2003.