

## ESPECIFICAÇÃO DO ESCOPO DE SISTEMA DE SOFTWARE A PARTIR DA UTILIZAÇÃO DA ENGENHARIA DE REQUISITOS

**Rosiane da Silva Biscaia**

Discente do curso Tecnologia em Análise e Desenvolvimento de Sistemas  
Faculdades Integradas de Três Lagoas – AEMS

**Victor Luís de Magalhães**

Discente do curso Tecnologia em Análise e Desenvolvimento de Sistemas  
Faculdades Integradas de Três Lagoas – AEMS

**Roberto Monteiro da Silva**

Discente do curso Tecnologia em Análise e Desenvolvimento de Sistemas  
Faculdades Integradas de Três Lagoas – AEMS

**Alan Pinheiro de Souza**

Docente da AEMS  
Mestre em Informática  
Área de Sistemas de Informação  
Faculdades Integradas de Três Lagoas – AEMS

### RESUMO

O objetivo desse projeto foi utilizar a abordagem de engenharia de requisitos para compreensão das características de determinado contexto corporativo, visando identificar problemas a serem solucionados por intermédio do desenvolvimento de um sistema computacional. Visando atingir esse propósito foram utilizados processos, métodos e ferramentas da engenharia de requisitos. A aplicação dessa abordagem foi realizada no contexto de uma empresa que fornece serviços de musculação e fitness. A coleta de requisitos foi realizada a partir de várias entrevistas junto aos *stakeholders* do projeto e a especificação do escopo do sistema de *software* foi obtida a partir da adoção da modelagem de diagramas de análise construídos com a linguagem de modelagem UML. A aplicação desses processos, métodos e ferramentas permitiu o levantamento das características do domínio do problema, a documentação de vasto conteúdo de informações e a elaboração de diagramas precisos e completos a respeito do negócio.

**PALAVRAS-CHAVE:** Engenharia de requisitos; Modelagem; UML; *Software*.

### INTRODUÇÃO

Hoje quando se pensa em desenvolvimento de *software*, pode parecer uma tarefa fácil levando em conta todos os recursos e ferramentas que são oferecidos no mercado para a automatização de uma empresa. Segundo (SOMMERVILLE, 2007) muita gente associa o termo *software* aos programas de computadores sem levar em conta que um *software* não é composto somente pelo programa em si, mas

também por toda documentação associada ao mesmo e aos dados de configurações necessários para fazer com que esses programas operem corretamente.

Mesmo tendo em mãos um projeto para uma empresa de médio porte, a elaboração do mesmo deve ser realizada de forma concreta, tendo como objetivo alcançar um nível mais elevado em termos de qualidade. Dessa forma, durante a elaboração deste projeto foram seguidas várias atividades de validação e revisão da documentação e demais artefatos de projeto de modo a garantir qualidade nas atividades e tarefas que estavam sendo realizadas. Segundo (SOMMERVILLE, 2007), a engenharia de *software* se ocupa de todos os aspectos da produção de um de *software*, desde os estágios iniciais de especificações do sistema até a manutenção desse sistema, depois que ele entrou em operação.

Em razão da aplicação de métodos e técnicas de engenharia de *software* espera-se que as soluções sejam obtidas de forma mais rápida e coerente alcançando assim resultados mais satisfatórios no final desse projeto.

O projeto foi elaborado para uma empresa que presta serviço de musculação e *fitness*. Os principais objetivos do projeto são ajudar a empresa, principalmente, na organização dos procedimentos e na comunicação interna ente seus funcionários e externa com seus clientes. Acredita-se que esses objetivos, quando alcançados venham a contribuir para o crescimento da empresa como é da vontade do proprietário.

Este trabalho está organizado em duas macro seções. Além das seções de introdução e considerações finais do trabalho, a seção um discute sobre engenharia de requisitos e a seção dois apresenta a aplicações dessa abordagem de engenharia de requisitos em contexto de aplicação, destacando a modelagem do diagrama de classes e diagrama de casos de uso. A seção final lista as referências bibliográficas adotadas na construção do trabalho.

## **1. ENGENHARIA DE REQUISITOS**

O conjunto das técnicas de levantamento, documentação e análise que formam a engenharia de requisitos fazem com que o projeto tenha os requisitos implícitos minimizados. Quando a documentação é bem feita, os requisitos documentados têm maiores chances de serem corretamente entendidos.

Uma dificuldade é que muitos clientes não entendem a necessidade da especificação de requisitos. A não utilização da engenharia de requisitos pode gerar um risco de resolver o problema errado. Por outro lado, é possível a existência de requisitos que não correspondam as reais necessidades dos clientes e usuários, essa falha indica que a engenharia de requisitos não foi feita de maneira adequada.

Cabe ao engenheiro de *software* deixar claro que uma boa especificação de requisitos é indispensável, que não representa custos supérfluos, mas sim investimentos necessários, que se pagam com altos juros quando não são bem feitas ou deixam de ser feitas, que a participação dos usuários na engenharia de requisitos é fundamental para que as necessidades sejam corretamente atendidas pelo produto, que uma boa especificação de requisitos custa tempo e dinheiro, mas que a ausência dela custa muito mais tempo e dinheiro (PAULA FILHO, 2000).

## **2. CONTEXTO DE APLICAÇÃO**

Com a engenharia de requisitos é realizado o levantamento de requisitos, onde são levantados os requisitos funcionais e não funcionais do sistema, os problemas existentes na empresa são identificados e descritos para que o sistema possa supri-los, as soluções alternativas são apresentadas aos donos e futuros usuários do sistema para escolha de um sistema específico para a empresa. Impactos virão com a implantação do novo sistema, as principais mudanças serão no ambiente físico com a implantação de maquinário específico e o treinamento dos usuários para a adaptação a nova forma de trabalho.

### **2.1. Levantamento de Requisitos**

No levantamento de requisitos é feita a definição de requisitos do sistema que segundo (SOMMERVILLE, 2007), se destina a coletar as características do sistema como um todo. Esse processo envolve várias consultas com os clientes e usuários finais. Essa fase de definição de requisitos, normalmente, se concentra em derivar três tipos de requisitos: requisitos funcionais, propriedades do sistema e características que o sistema não deve possuir.

Os principais requisitos do sistema já começaram a ser levantados nessa fase do projeto, tendo assim uma maior visão de quais seriam os requisitos

funcionais e não funcionais do sistema, com uma maior abrangência do que o usuário final desejava. Essa seção mostra a descrição do sistema atual, com os problemas existentes na empresa, os desejos do usuário, as soluções alternativas apresentadas pela equipe de desenvolvimento, a alternativa escolhida pela empresa e, por último, os diagramas de classe e caso de uso.

Esse levantamento de requisitos foi realizado pelos alunos envolvidos no projeto, através de entrevistas com os funcionários sobre os processos de atendimento, administrativo e desejos do usuário quanto ao desenvolvimento de um sistema computacional. O acompanhamento foi realizado por intermédio de visitas *in loco* na empresa para análise de características do ambiente e funcionamento do sistema atual e reuniões para definir os requisitos necessários para o desenvolvimento do sistema.

### **2.1.1 Problemas Existentes**

Como todos os procedimentos da academia são feitos manualmente e essas informações são utilizadas por todos os funcionários da academia, existem alguns problemas que geram transtornos para os funcionários da academia. Um dos problemas mais recorrentes é a perda de informação, em razão do mau armazenamento das fichas. Esse tipo de problema gera insatisfação tanto para os alunos quanto para os funcionários e donos da academia.

Essa perda ocorre porque a mesma ficha é utilizada por todos os setores da academia e, nem sempre ao final do dia as fichas são guardadas no arquivo. O controle das mensalidades também é prejudicado porque com a perda das fichas onde elas são anotadas, o gerente não sabe se o aluno realizou o pagamento e assim fica impossibilitado de realizar as devidas cobranças, prejudicando também o departamento financeiro da academia.

As informações técnicas dos alunos também sofrem com esse processo, o professor perde toda a evolução dos alunos, que por sua vez se sentem lesados por não conseguirem acompanhar sua evolução. Por outro lado não é realizado um acompanhamento financeiro da academia porque geração de torna-se uma tarefa bastante custosa.

### **2.1.2 Desejos do Cliente**

Para a solução dos problemas mencionados anteriormente, os proprietários desejam um sistema operacional e gerencial com cadastros de clientes e medidas e que permita o controle das mensalidades pagas por todos os alunos, mostrando a situação dos mesmos, ou seja, se o aluno está ativo ou não na academia. Outra vontade o usuário é a geração de relatórios para o acompanhamento tanto financeiro quanto de evolução dos alunos.

Os proprietários necessitam que o sistema viabilize o cadastro de clientes e que contenha pesquisa por nomes deixando assim o atendimento mais dinâmico e rápido. O sistema também deve conter o cadastro de medidas para que o professor consiga fazer as anotações necessárias sobre os alunos, tendo assim fácil acesso às informações técnicas. Na visão do cliente, a construção de um sistema com essas características deixará os alunos mais satisfeitos com o atendimento disponibilizado pela academia.

O sistema deverá disponibilizar relatórios diários, mensais ou anuais para o gerente sobre as mensalidades e o professor também terá a acesso a relatórios sobre a evolução dos alunos. Esses relatórios poderão ser impressos, caso seja de interesse do aluno. Outro desejo do proprietário da academia é que o sistema tenha uma *interface* simples, para que o sistema seja de rápido aprendizado e fácil manuseio.

### **2.1.3 Soluções Alternativas**

Foram apresentadas três soluções para implantação na empresa. Todas possuem aspectos importantes e que contribuem para melhoria de processos de administração da empresa.

1. Através de uma pesquisa de mercado foi cogitada a hipótese de adquirir uma licença de um *software* para academia de terceiros que já está no mercado há algum tempo e que tenha credibilidade para atender os desejos dos proprietários, funcionários e alunos da academia. O *software* em questão poderia atender as necessidades de todos os interessados, porém, poderia vir com opções que os proprietários não aprovariam e que o tornaria uma ferramenta muito complicada levando em conta que os funcionários necessitam de uma ferramenta de trabalho que seja rápida e de fácil compreensão. O aspecto financeiro também foi

levado em conta, pois a licença foi considerada cara em relação às condições financeiras disponíveis no momento pela empresa.

2. O desenvolvimento de um sistema na ferramenta *Microsoft Office Excel* foi uma opção apresentada ao cliente levando em conta todas as exigências do mesmo. Essa alternativa foi discutida com o intuito de poupar esforços na parte de codificação e implantação do sistema por ser uma ferramenta de fácil acesso e aquisição. Essa alternativa foi descartada nos quesitos conhecimento da ferramenta e custo de aquisição. O cliente deixou claro que sua equipe não era habituada a essa ferramenta e que seria difícil a adaptação da empresa ao novo sistema, principalmente caso aconteça algum erro em relação às formulas. O valor que seria gasto para a aquisição da licença foi considerado alto, fora os gastos com treinamento dos funcionários e estrutura física (*hardware*) que seria necessária para a instalação do sistema. A segurança dos dados também foi estudada, pois o usuário poderia facilmente alterar alguma informação causando transtornos para ambas as partes.

3. A terceira opção oferecida ao cliente seria o desenvolvimento de um *software* sob encomenda. Esse sistema teria uma interface mais organizada em relação ao Excel, além disso, teria um custo de desenvolvimento bem menor, pois dispensa a necessidade de aquisição de uma licença. A segurança dos dados também é uma vantagem a favor, pois o *software* é gerado em modo executável, sendo assim menos improvável alteração nos dados ou programação.

#### **2.1.4 Solução Escolhida**

Tendo em mãos todos os prós e os contras das opções apresentadas e visando as melhorias que eram esperadas para a academia e a qualidade no atendimento aos alunos, a terceira opção foi considerada a melhor a ser escolhida. Tendo um *software* que contenha exatamente aquilo que se procura para suprir as necessidades de proprietários, funcionários e alunos serão mais fáceis a elaboração do projeto e adequação de todos ao novo sistema de trabalho.

A primeira e a segunda opção foram descartadas por causa de condições financeiras, além disso, as funcionalidades dos dois sistemas e o grau de dificuldade de manipular representaram obstáculos, para o atendimento das expectativas dos proprietários.

### **2.1.5 Impactos do Sistema na Organização**

Com a implantação do novo sistema espera-se que traga melhorias para a academia, aperfeiçoando assim o atendimento e a opinião dos alunos a respeito da organização da empresa. Com um sistema automatizado, todas as operações que eram feitas manualmente serão abolidas, assim todos os cadastros de cliente e medida serão feitos no sistema, diminuindo riscos de perda de dados.

Com o novo sistema o controle das mensalidades será feito de maneira automatizada, diminuindo problemas com mensalidades atrasadas ou cobradas mais de uma vez. Além disso, a geração de relatórios facilitará na hora das cobranças e permitirá conhecer mais a respeito do desenvolvimento financeiro da academia. Com os relatórios de medidas, o professor poderá ter uma maior noção sobre a evolução e desempenho dos alunos, que por sua vez poderá acompanhar suas próprias evoluções, se sentindo melhor atendido.

A implantação do novo sistema o ambiente físico causará alterações, por exemplo, a aquisição de computadores para a instalação do *software*. Será necessária a compra de três computadores, um para a recepção, uma para o escritório e outro para uso dos professores. A compra de uma impressora também será necessária para a geração de relatórios e recibos.

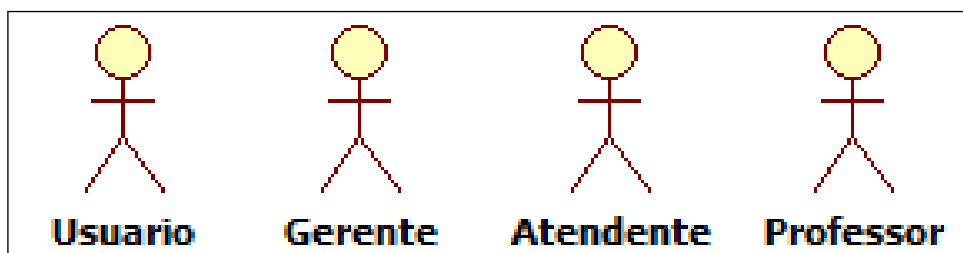
Para a manipulação do sistema os funcionários também passarão por treinamentos para que a implantação do *software* não se torne um problema, esses treinamentos serão para sanar as dúvidas dos funcionários e para a apresentação do sistema a toda a equipe.

## **2.2. Especificação do Escopo**

Após o levantamento de requisitos já será possível realizar a descrição dos atores que são quem integram com o sistema, a lista de eventos que o sistema realizada como cadastro de cliente, o diagrama de caso de uso vira para demonstrar os eventos do sistema graficamente como seus cursos normais e alternativos e também o diagrama de classe que mostra os conjuntos de classes, interfaces e colaboração e seus respectivos relacionamentos existentes no *software*.

### 2.2.1 Definição dos Atores

Os atores são quem interage com o sistema, mas sobre o qual não se tem controle eles estão fora da influência do sistema, ou seja, os atores tem um papel externo e são quem dão início aos casos de uso. Tipicamente os atores representam o papel que o ser humano, outro processo, outro sistema ou até mesmo um dispositivo de *hardware* desempenha ao interagir com o sistema. Cada ator corresponde a um papel específico, uma pessoa que desempenha diferentes papéis nas interações com o sistema é representado por diferentes atores, por outro lado, diversas pessoas que desempenham o mesmo papel correspondem a um único ator. São eles quem utiliza o sistema, fiscalizam o sistema, fornecem os dados e usam as informações do sistema (MATOS, 2002). A Figura 1 demonstra os atores que desempenharam as funções do sistema: usuário, gerente, atendente e professor. O ator usuário representa qualquer um dos demais atores do sistema.



**Figura 1** – Descrição dos atores.  
**Fonte:** Elaborada pelos autores.

### 2.2.2 Lista de Eventos

A lista de eventos mostra as funcionalidades do sistema que vai de efetuar *login*, cadastros e geração de relatórios. Na sequência, foram apresentados os eventos que serão atendidos pelo sistema.

Evento 1 – Efetuar *Login*

Evento 2 – Cadastrar Usuário

Evento 3 – Cadastrar Cliente

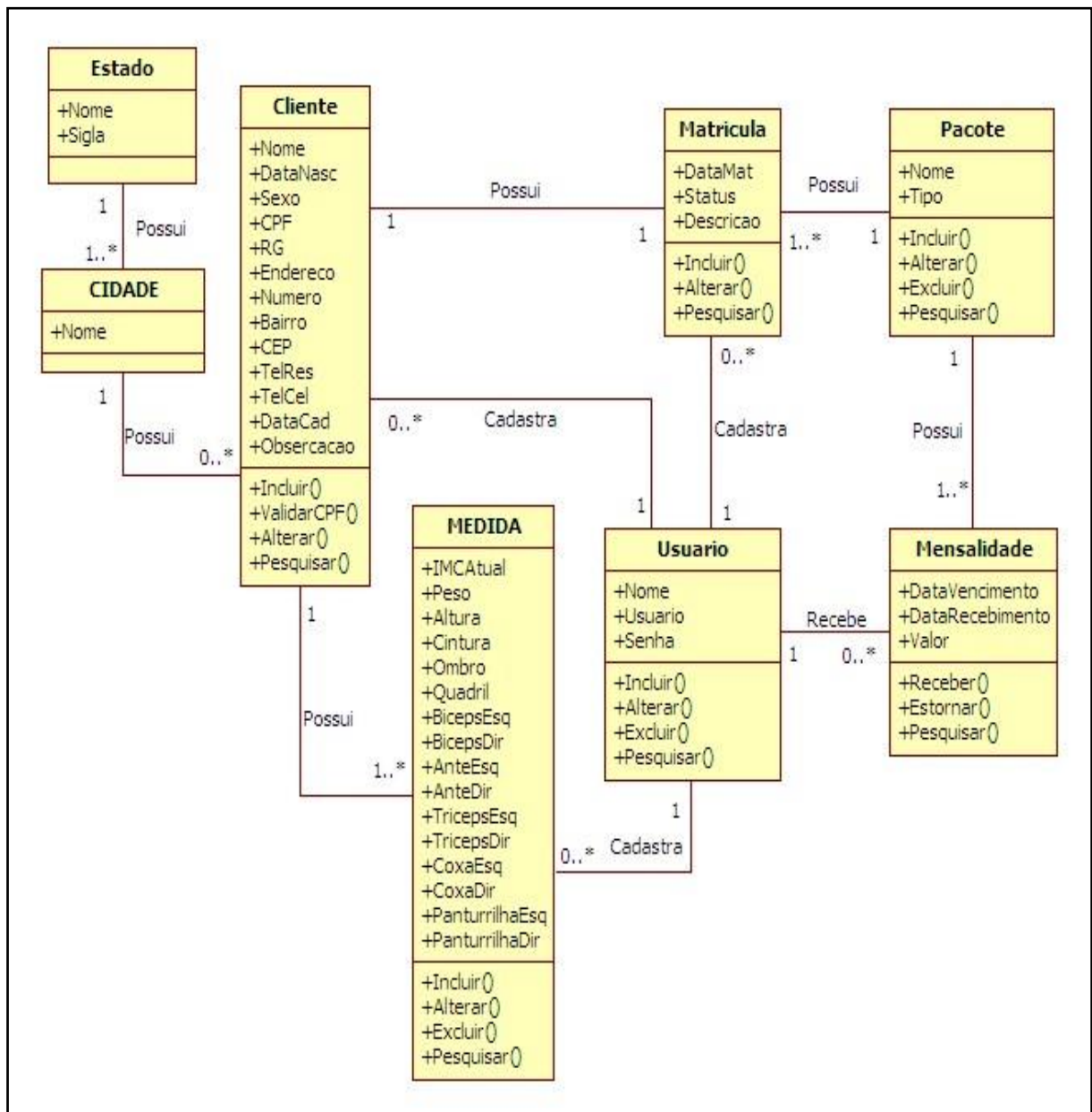
- Evento 4 – Cadastrar Matrícula
- Evento 5 – Cadastrar Medida
- Evento 6 – Cadastrar Pacote
- Evento 7 – Receber Mensalidade
- Evento 8 – Gerar Relatório de Usuário
- Evento 9 – Gerar Relatório de Cliente
- Evento 10 – Gerar Relatório de Medida
- Evento 11 – Gerar Relatório de Matrícula
- Evento 12 – Gerar Relatório de Pacote
- Evento 13 – Gerar Relatório de Recebimento

### **2.2.3 Diagrama de Classes**

A equipe de projeto adotou a UML (Linguagem Unificada de Modelagem) para visualização, especificação, construção e documentação dos artefatos. O diagrama de classe é o diagrama encontrado com maior frequência na modelagem de sistemas orientada a objeto, ele é importante tanto para a visualização, a especificação e a documentação de modelos, quanto para a construção de sistemas executáveis por intermédio de engenharia de produção e reserva (BOOCH, RUMBAUGH, JACOBSON, 2000). Um diagrama de classes mostra um conjunto de classes, *interfaces* e colaboração e seus respectivos relacionamentos, graficamente é uma coleção de vértices e arcos.

Sua utilização é para fazer uma modelagem da visão estática do sistema em questão, oferecendo suporte para os requisitos funcionais do sistema, ou seja, as ações que o sistema irá executar. Segundo Deboni (2003), diagramas de classes são modelados para esquematizar o domínio de um sistema informacional, contribuindo para elaboração da estrutura inicial do *software* a ser desenvolvido, as classes representam as entidades que serão operadas pelo sistema, além das interações durante funcionamento da aplicação.

A Figura 2 demonstra o diagrama de classes do sistema, onde foram modeladas as classes Cliente, Cidade, Estado, Medida, Usuário, Matrícula e Mensalidade e Pacote. Para cada classe foi apontada respectivos atributos e métodos. Os relacionamentos e as cardinalidades entre as classes também são exibidos.



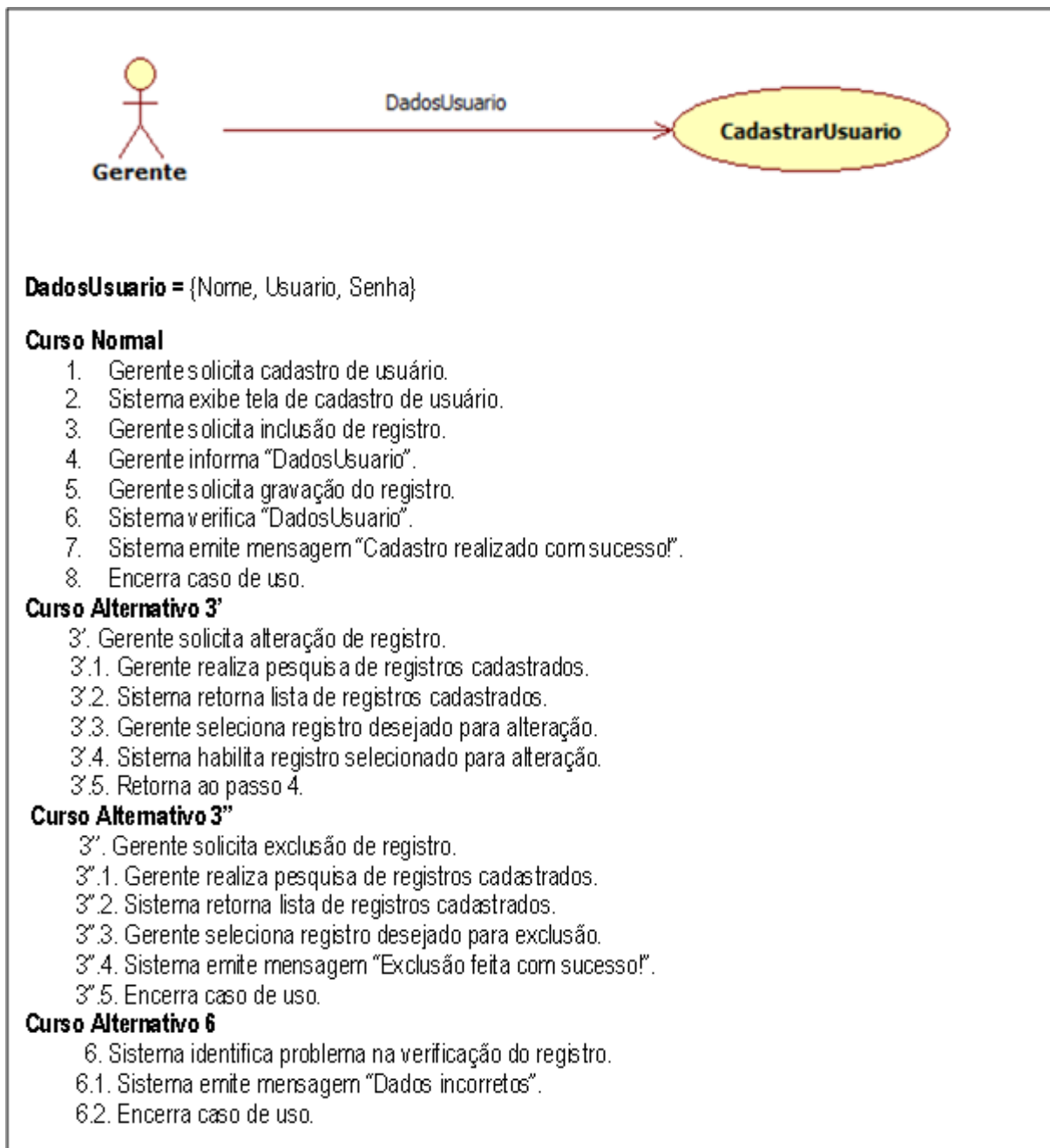
**Figura 2** – Diagrama de Classes.  
Fonte: Elaborada pelos autores.

#### **2.2.4 Diagrama de Casos de Uso**

Os diagramas de casos de uso têm um papel central para a modelagem do comportamento de um sistema cada um mostra um conjunto de casos de uso e atores e seus relacionamentos, os diagramas de casos de uso são importantes para visualizar, especificar e documentar o comportamento de um elemento. Tecnicamente, um diagrama de caso de uso é um diagrama que mostra um conjunto de casos de uso, atores e seus relacionamentos (BOOCH, RUMBAUGH, JACOBSON, 2000).

Os diagramas de caso de uso devem conter casos de uso, atores e relacionamentos de dependência, generalização e a associação. Assim como os outros diagramas podem conter notas e restrições. Esse diagrama é aplicado para fazer modelagem da visão estática de caso de uso do sistema, essa visão proporciona suporte principalmente para o comportamento de um sistema, os serviços extremamente visíveis que o sistema fornece no contexto de seu ambiente (BOOCH, RUMBAUGH, JACOBSON, 2000).

Foram elaborados diagramas de casos de uso, assim como roteiro normal e roteiros alternativos para cada um dos eventos apresentados na Seção 2.2.2. Esse detalhamento foi necessário para compreender todas as situações de interação dos atores com as funcionalidades do sistema. A Figura 3 exemplifica o diagrama de caso de uso de cadastro de usuário, nele é demonstrado o curso normal do cadastro de usuário, onde o gerente solicita um novo cadastro, informa os dados do novo usuário e confirma o salvamento do registro. Além disso, os cursos alternativos foram modelados, por exemplo, caso o gerente solicite uma alteração, pesquisa ou exclusão de algum registro; ou quando os dados informados para a realização do novo cadastro estejam errados.



**Figura 3** – Diagrama de Caso de Uso – Cadastrar Usuário  
**Fonte:** Elaborada pelos autores

## CONSIDERAÇÕES FINAIS

Esse projeto proporcionou percepção da necessidade de realizar uma análise *in loco* e avaliar as necessidades do cliente para que seja possível desenvolver um sistema que acomode as características solicitadas. O *software* auxiliará no controle da empresa, melhorando a organização dos processos, facilitando os serviços que hoje são executados de forma manual. Este trabalho tem

como objetivo construir um *software* que seja de fácil operação para o cliente, visando minimizar sua dificuldade para realizar cadastro de cliente, matrícula, usuário, medidas e pacote, além da geração e impressão de relatórios.

Entre os desafios do projeto está o esforço de migração dos dados arquivados em fichas em papel para repositório de dados do sistema. Destaca-se ainda a importância do *software* para crescimento do negócio e seu desenvolvimento e futura manutenção por um profissional que possa atender ao que a empresa necessita de maneira eficiente e com qualidade.

Uma limitação existente dentro do projeto foi à dificuldade de comunicação entre a equipe do projeto e o proprietário da empresa. Ainda assim, foi possível realizar uma análise bem detalhada quanto ao levantamento de requisitos, entrevistando todos os possíveis usuários do sistema de modo a coletar e especificar conjunto satisfatório das características do sistema a ser desenvolvido.

## REFERÊNCIAS

BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. **UML: Guia do Usuário**. 12ª Ed., Rio de Janeiro: Campus, 2000.

DEBONI, José Eduardo Zindel. **Modelagem Orientada a Objetos com a UML**. São Paulo: Editora Futura, 2003.

MATOS, Alexandre Veloso. **UML Prático e Descomplicado**. São Paulo: Érica, 2002.

PAULA FILHO, Wilson de Pádua. **Engenharia de Software: Fundamentos, Métodos e Padrões**. 2ª Ed., Rio de Janeiro: LTC, 2000.

SOMMERVILLE, Ian. **Engenharia de Software**. 8ª Ed., São Paulo: Addison Wesley Brasil, 2007.