

# Coruja: Sistema de apoio a gestão pedagógica da Coordenação Pedagógica do CEFET-MG *campus* Divinópolis

Alberto Elias Do Amaral Júnior<sup>1</sup>, Anthony Alves Rabelo<sup>1</sup>, Leonardo Amorim de Sena<sup>1</sup>  
Marcelo Caramuru Pimentel Fraga<sup>1</sup>, José Maurício Costa<sup>1</sup>

<sup>1</sup>Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG)  
Caixa Postal 400 – 35.503-822 – Divinópolis – MG – Brasil

albertoelias@live.com, anthonyrabelo@hotmail.com

{leonardo123.amorim, caramurucefet, jmcosta25}@gmail.com

**Abstract.** *Pedagogical management is one of the activities which the Pedagogical Coordination (CP) is responsible. Based on the demand of the members of the CEFET-MG CP and the market research done in this work, it is noted that pedagogical management lacks systems that assist them in the execution of activities of monitoring and identification of students, since both are done manually. Therefore, this work aims to develop a Web System to assist in pedagogical management based on the requirements presented by the CP of CEFET-MG campus Divinópolis. It is expected that the system will solve the problems raised by the office, supplying the lack of software in the areas mentioned.*

**Resumo.** *A gestão pedagógica é uma das atividades pelas quais a Coordenação Pedagógica (CP) é responsável. Com base na demanda dos membros da CP do CEFET-MG e na pesquisa de mercado feita neste trabalho, nota-se que a gestão pedagógica carece de sistemas que as auxiliem na realização das atividades de acompanhamento e identificação dos estudantes, uma vez que ambas são feitas manualmente. Por isso, este trabalho visa desenvolver um Sistema Web para auxiliar na gestão pedagógica com base nos requisitos apresentados pela CP do CEFET-MG campus Divinópolis. Espera-se que o sistema resolva os problemas levantados pelo órgão, suprimindo a carência por um software nos âmbitos apontados.*

## 1. Introdução

A gestão escolar constitui uma das áreas de atuação profissional na educação destinada a realizar o planejamento, a organização, a coordenação e a avaliação dos processos necessários à efetividade das ações educacionais. Em caráter específico, a gestão pedagógica é a área responsável por desenvolver ações voltadas para a promoção da aprendizagem dos alunos e sua formação. Nesse contexto, segundo Heloísa (2009), o pedagogo tem como atribuição a organização dos processos de transmissão e assimilação de saberes, assim como modos de ação na escola. De acordo com Vindi (2017), existem no mercado sistemas de gestão escolar que permitem o controle e o gerenciamento de processos como os financeiros, os contábeis, os administrativos e outros indispensáveis para uma instituição de ensino. No entanto, como será apresentado adiante neste artigo, essas aplicação de software não possuem funcionalidades que atendem satisfatoriamente os requisitos da gestão

pedagógica. Assim, percebe-se que há uma falta de sistemas que ajudem na realização de tais tipos de atividades como, por exemplo, o acompanhamento pedagógico e o registro de atendimento dos alunos, especificadas por Libâneo (2007).

A Coordenação Pedagógica (CP), situada no Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG) *campus* Divinópolis, tem como função implementar as políticas de ensino da instituição. Para isso, a CP realiza duas atividades, tema deste trabalho. A primeira delas é o acompanhamento pedagógica, na qual a CP é responsável por cuidar do atendimento e da orientação educacional dos alunos. Já a segunda é a montagem do carógrafo, que consiste no registro de dados dos alunos agrupados por sua classe, incluindo foto, para que eles possam ser identificados pela CP e pelo Conselho de Classe, órgão de natureza deliberativa quanto à avaliação escolar dos estudantes, tal qual dito por Libâneo (2001a). No entanto, ambas são feitas de forma manual, ou seja, sem o auxílio de qualquer tipo de sistema ou ferramenta tecnológica que auxilie na realização delas. Por isso, essas atividades podem ser otimizadas com o uso de um sistema de informação gerencial.

Assim, dadas as necessidades específicas que uma CP possui e a demanda de *softwares* voltados à realização das tarefas citadas anteriormente, o presente trabalho propõe o desenvolvimento de um sistema voltado à gestão pedagógica conforme os requisitos apresentados pela CP do *campus* Divinópolis. Dessa forma, foi criado um Sistema *Web* que possibilita a manutenção dos dados dos discentes e docentes, bem como dos registros de atendimento relacionados a eles. Consequentemente, a partir dessas informações, é possível gerar o carógrafo e o relatório de acompanhamento/atendimento, respectivamente. Portanto, com o uso de tais funcionalidades, o sistema proposto será capaz de auxiliar a CP na realização das suas atividades por meio da automatização dos processos de registro e consulta dos dados referentes ao acompanhamento das atividades de ensino.

## **1.1. Objetivos**

### **1.1.1. Objetivo Geral**

Este projeto tem como objetivo desenvolver um sistema que auxilie a CP na gestão de informações acerca do corpo discente e docente da instituição para a realização das atividades pedagógicas de acompanhamento e identificação dos alunos.

### **1.1.2. Objetivos Específicos**

Os objetivos específicos do trabalho são apresentados abaixo:

1. Contribuir com a melhoria da gestão pedagógica do *campus* por meio de recursos que tornem a realização de suas atividades mais ágeis;
2. Automatizar os processos de registro de atendimento e confecção do carógrafo dos alunos
3. Fornecer informações para os membros da CP e do Conselho de Classe acerca dos alunos com o intuito de auxiliar no processo de atendimento e acompanhamento deles.

## 2. Revisão de Literatura

Para Libâneo (2001b), a pedagogia é definida como o campo de conhecimento responsável por investigar a natureza, as finalidades e os processos necessários às práticas educativas com o objetivo de propor a realização desses processos nos vários contextos em que essas práticas ocorrem. Por isso, configura-se como ciência da educação.

Neste caso, a Coordenação Pedagógica tem como finalidade contribuir para a solução de problemas práticos através da organização de elementos e métodos para o exercício da educação. Dentre as práticas das Coordenações Pedagógicas está o acompanhamento pedagógico de docentes, discentes e da instituição. Essa supervisão pode ser entendida como uma série de investigações que envolvem o planejamento e a implementação de interferências, mudanças ou inovações – destinadas a produzir avanços e melhorias nos processos de aprendizagem dos sujeitos que delas participam – e a posterior avaliação dos efeitos dessas interferências, em conformidade com Damiani et al. (2013).

Os Sistemas de Informação (SI), por sua vez, são definidos como mecanismos de apoio a gestão, desenvolvidos com base na tecnologia de informação e com suporte da informática para atuar como condutores das informações que visam facilitar, agilizar e otimizar o processo decisório nas organizações. De forma estruturada, os SI's dão condições para que as informações estejam alicerçadas por um processo decisório forte e suficiente para garantir a resolução dos problemas, segurança, agilidade e versatilidade, tal qual dito por Bazzotti (2006).

Ainda segundo Bazzotti (2006), o Sistema de Informação do tipo SIG (Sistema de Informação Gerencial) tem como propósito básico ajudar uma organização a alcançar suas metas. Sistemas como esse são também usados por instituições de ensino e fornecem aos pedagogos e gerentes escolares detalhes sobre as operações regulares da instituição, de forma que possam controlar, organizar e planejar com mais efetividade e eficiência. Logo, os SIG's escolares são exemplos comprobatórios das possibilidades oferecidas por essa tecnologia na gestão dos diversos processos que ocorrem na escola.

Pode-se citar alguns SIG's escolares disponíveis no mercado brasileiro a efeito de ilustração. O primeiro deles é o Sponte, *software online*, não gratuito, para uso em escolas de ensino infantil ao médio, de ensino técnico, superior ou de idiomas. Essa solução integra, conforme a desenvolvedora Sponte (2017), todos os dados e processos escolares em um único sistema, oferecendo módulos para a gestão financeira, comercial, acadêmica, de biblioteca e de secretaria. Fornece também portais específicos para alunos, professores, pais e funcionários da escola de forma a apresentar dados relevantes à necessidade do usuário.

Outra solução que ilustra o uso de SIG por escolas é o Sistema Galileu. De acordo com sua desenvolvedora Inovaise (2017), ele é um serviço *online* para diversos tipos de escola, tem suporte para gestão financeira, secretarial, comercial e bibliotecária. Também apresenta a funcionalidade de portal de acesso à informação por parte de pais e alunos.

O sistema GIZ, segundo sua desenvolvedora Sistemas (2017), oferece soluções para escolas públicas e privadas de pequeno, médio e grande porte, universidades e de cursos livres. O sistema possui diversas funcionalidades para atender as demandas de instituições de ensino, como módulos para a realização de processos seletivos,

coordenação de pesquisa e extensão, controle de frequência dos alunos, controle de ponto dos funcionários, matrícula dos alunos, controle de bibliotecas, gestão financeira, integração com o sistema *Moodle* (software livre de apoio à aprendizagem, executado num ambiente virtual), entre outras funcionalidades.

Já o SI Q-Acadêmico, adotado no CEFET-MG, também é exemplo do uso da tecnologia para facilitação da gestão escolar, além de servir, em ambiente *web*, também aos alunos e a outras partes interessadas no processo educativo. Consoante a desenvolvedora Qualidata (2017), o Q-Acadêmico integra os módulos de professores e alunos para que ambos tenham registro e acesso às informações pertinentes às suas posições, como pontuações, boletins, diários de presença, material de aula, etc.

Por fim, a empresa SophiA cujo sistema possui o mesmo nome, disponibiliza, assim como os outros, meios para a administração escolar e acadêmica. Tal qual disponibilizado por SophiA (2017), esse *software* de gestão escolar inclui módulos contábeis, fiscais, financeiros, acadêmico, patrimonial, bibliotecário, comunicacional e, finalmente, secretarial. Esse sistema também é adotado pelo CEFET-MG para a gestão das suas bibliotecas.

Portanto, os exemplos de SI citados anteriormente, entre vários outros disponíveis no mercado, demonstram a viabilidade do uso da tecnologia nas escolas como alternativa para organizar, planejar e executar com mais efetividade e maior eficiência os seus processos. No entanto, como foi possível perceber pela amostra de SI's escolares, o seu enfoque não se contém na gestão pedagógica, logo, nas atividades da Coordenação Pedagógica. Por isso, a necessidade de uso exclusivo para o acompanhamento pedagógico de discentes e de docentes de uma instituição de ensino não se encontra amparada e contemplada pelos SI's de gestão escolar já desenvolvidos. A Tabela 1 demonstra essa situação.

É importante ressaltar que nem todos as plataformas apresentados são gratuitos e, por isso, demandam recursos financeiros que muitas instituições de ensino no país podem não possuir. Por esse motivo, as escolas deixam de usar essas soluções, ficando em aberto a possibilidade de melhorias com a sua implantação.

Logo, considerando a falta de *softwares* que auxiliem a coordenações pedagógicas como a do *campus* Divinópolis do CEFET-MG, por exemplo, na realização das atividades de gestão pedagógica o presente trabalho propõe uma ferramenta para a melhoria de seus processos de gestão pedagógica. Através de um desenvolvimento direcionado para os requisitos e especificidades da Coordenação Pedagógica, este sistema tem como funcionalidades: a manutenção de relatórios de acompanhamento de membros da comunidade escolar bem como a geração do carógrafo (documento de identificação dos estudantes por turmas) a partir de dados gravados no banco de dados no sistema. Assim, o SIG Coruja atuará no processo de transformação de dados em informações que serão utilizadas na estrutura decisória da CP, proporcionando a sustentação administrativa para otimizar os resultados esperados.

### **3. Materiais e Métodos**

Para que este trabalho fosse desenvolvido de acordo com os requisitos definidos pela CP, o projeto Coruja foi feito conforme o modelo de desenvolvimento de prototipação. Esse padrão, segundo Pressman and Maxim (2016), é classificado como um processo evolucionário, uma vez que o *software*, em sistemas complexos, evolui ao longo do tempo. A

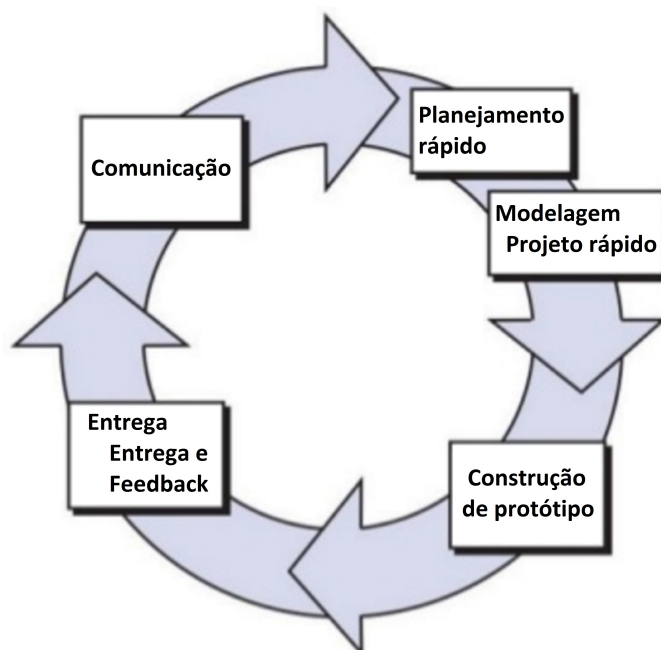
	Coruja	Sponte	Galileu	Q-Acadêmico	GIZ	SophiA
Manutenção de dados de Discentes e Docentes	✓	✓	✓	✓	✓	✓
Carógrafo	✓	✗	✗	✗	✗	✗
Gerenciamento dos Atendimentos	✓	✗	✗	✗	✗	✗
Relatório de Acompanhamento	✓	✗	✗	✗	✗	✗
Servidor de Arquivos	✗	✓	✓	✓	✓	✓
Disponibilização de notas online	✗	✓	✓	✓	✓	✓
Módulo Financeiro	✗	✓	✓	✗	✓	✓

**Tabela 1. Tabela comparativa com os softwares disponíveis no mercado**

medida que o *software* é desenvolvido, os requisitos podem mudar, tornando-se inadequado seguir um planejamento linear. Nesse caso, o processo de desenvolvimento tem que ser encadeado ciclicamente nas etapas de 1) Comunicação, 2) Planejamento e Modelagem, 3) Construção de Protótipos e 4) Entrega e *Feedback*. Como este modelo é evolucionário, ele segue um processo cíclico de desenvolvimento em que a cada execução as etapas citadas anteriormente são realizadas, respeitando a sequência em que elas devem ser feitas. Portanto, tal ciclo de etapas é repetido até que o sistema seja concluído.

Os estágios do processo de prototipação e a sequência em que elas ocorrem são ilustradas na Figura 1. A primeira delas é a de comunicação, fase em que é realizado o levantamento dos requisitos. Desta forma, foram feitas reuniões e entrevistas com os usuários do sistema para que pudessem ser identificadas as suas necessidades bem com as principais funcionalidades que o sistema deve ter. Após a definição do escopo do projeto, deu-se início a sua modelagem e planejamento executada por meio da criação dos planos e diagramas de Tabela Relacional, Casos de Uso e de Classes que representam as estruturas de armazenamento e funcionamento do sistema. Em seguida, foi realizada a fase de construção de protótipos, ou seja, a criação de uma versão dos módulos do sistema para que tanto os usuários quanto a equipe de desenvolvimento pudessem apurar a execução do programa. Por fim, na etapa de entrega e *feedbacks* procedeu-se a submissão da versão mais recente do sistema, até então, para a realização de testes e com o intuito de se verificar a sua adequação aos requisitos do projeto e a ocorrência de erros. Por fim, após a coleta das informações acerca das melhorias que devem ser aplicadas no sistema, o processo de desenvolvimento do sistema segue com o seu ciclo evolucionário, passando por todas as etapas que o compõem novamente. Assim, tal ciclo de tarefas sucedeu-se até

que o projeto ficasse pronto.



**Figura 1. Paradigma de prototipação**

### **3.1. Tecnologias**

#### **3.1.1. Programação Orientada a Objetos (POO)**

Um paradigma de programação, conceituado por Dall'Oglio (2015), fornece e determina a visão que o programador possui sobre a estruturação e execução do programa. A POO aborda a concepção dos sistemas por meio de uma ótica mais próxima do mundo real. Os objetos são instâncias de determinadas classes. Estas definem atributos e métodos (semelhante a dados e ações, respectivamente), entre outras características, dos objetos. Atualmente é o paradigma de programação mais utilizado, pois é mais fácil lidar com estruturas que carregam dados e comportamentos do que com procedimentos e variáveis desagrupadas de um contexto, características do paradigma procedural.

#### **3.1.2. PHP**

PHP é uma das linguagens de programação orientada a objeto que mais crescem no mundo, estando presente em mais de 80% dos servidores de *Internet* existentes, de acordo com Dall'Oglio (2015). O PHP está em uso na maior parte das páginas web dinâmicas, pois permite a conexão do banco de dados com suporte a linguagem SQL, dentre outras vantagens. Se destaca também, em concordância com Niederauer (2005), a portabilidade, já que pode ser executado em diversos SO's, a gratuidade, o código aberto, a integração com HTML, a execução do código no lado servidor, o fácil aprendizado e a orientação a objetos.

### 3.1.3. Model View Crontroller

Modelo, visão e controlador, o MVC - *Model, View, Controller* - é um padrão de arquitetura de software que separa todo o desenvolvimento de uma aplicação em três camadas:

- *Model* – Responsável por gerenciar o comportamento dos dados da aplicação.
- *View* – Responsável por gerenciar a saída gráfica e textual da parte da aplicação visível ao usuário.
- *Controller* – interpreta as entradas do usuário, comandando a Visão e o Modelo para se alterarem de forma apropriada.

O MVC tem como vantagem, segundo Minetto (2007), a separação de lógica e apresentação, o que facilita o trabalho em equipe. Com o aumento da complexidade das aplicações desenvolvidas, sempre visando a programação orientada a objeto, torna-se relevante a separação entre os dados e a apresentação das aplicações. O MVC foi amplamente adaptado como uma arquitetura para as aplicações *Web*, por meio de uma abordagem *thin client*, ou seja, que coloca quase o modelo, a visão e a lógica do controlador inteiros no servidor, e quanto ao cliente, recebe a *view*.

Além disso, numa aplicação que utilize a arquitetura MVC, todas as funcionalidades de bancos de dados, tais como obter as conexões, mapear objetos para tipos de dados SQL ou executar comandos SQL, devem ser feitas por classes de objeto de acesso a dados. As classes DAO (acrônimo de *Data Access Object*), como são conhecidas, são um padrão para persistência de dados que permite separar regras de negócio das regras de acesso a banco de dados.

### 3.1.4. Unified Modeling Language

A Linguagem Unificada de Modelagem (UML, do inglês *Unified Modeling Language*) é uma linguagem gráfica para visualização, especialização, construção e documentação de artefatos de sistemas complexos de *software*, em consonância com Booch et al. (2006). Os modelos são uma simplificação da realidade pois fornecem uma visão, descrição e abstração de uma descrição abrangente do projeto do sistema. Com eles, quatro objetivos estabelecidos por Booch et al. (2006) são alcançados: visualização do sistema como se deseja e como é; especificação da estrutura e comportamento; guia de construção do sistema; e, por fim, documentação de decisões tomadas.

### 3.1.5. Banco de Dados

O Banco de Dados (BD) é definido por Heuser (2009) como sendo um conjunto de dados ou arquivos integrados que tem por objetivo atender a uma comunidade de usuários e sistemas. Dessa forma, evita-se a redundância não controlada de dados, ou seja, cada informação é armazenada uma única vez, sendo acessada pelos vários sistemas que dela necessitam.

A manutenção de um repositório compartilhado de dados, ou BD, se dá com o uso de Sistemas de Gerência de Banco de dados (SGBD), que é “um *software* que incorpora as funções de definição, recuperação e alteração de dados em um BD”, explanado também

por Heuser (2009). Dentre as vantagens dadas pelo SGBD está a maior simplicidade na manutenção de programas, pois uma separação clara das funções relacionadas ao BD torna-os mais facilmente compreensíveis. Além disso, a produtividade dos programadores também aumenta, já que os programas ficam menores, pois usam funções modulares.

Para o banco de dados do Coruja, foi utilizado o SGBD MySQL, pertencente a Oracle. Ele é um servidor e Sistema Gerenciador de Banco de Dados (SGBD) relacional que utiliza a linguagem SQL (*Structured Query Language*). Para Milani (2007), a finalidade de gerenciar e armazenar dados e informações atende tanto aplicações de pequeno porte quanto de grande porte. Os dados fornecidos por Niederauer (2005) mostram que o MySQL é o mais popular SGBD, com mais de 5 milhões de instalações, inclusive usado por grandes empresas. Tal fato se deve as vantagens oferecidas pelo banco de dados *open source*, como velocidade, escalabilidade e confiabilidade.

### 3.1.6. Tecnologias Web

Um Sistema *Web* é uma aplicação distribuída, ou seja, parte do processamento ocorre em um computador e parte em outro, denominado servidor, conforme explanado por Miletto and de Castro Bertagnolli (2014). Segundo Loudon (2010), esse tipo de sistema é organizado como um sistema cliente-servidor e geralmente apresentam as seguintes características: disponibilidade contínua, grande base de usuários, entrega em partes ao cliente, diversidade de aplicações *web*, longevidade da aplicação, múltiplos ambientes e atualização em tempo real.

Para o desenvolvimento *web* são utilizadas tecnologias de apresentação para o usuário explicadas por Loudon (2010). As mais comuns são HTML, CSS e JavaScript. HTML é a linguagem de marcação utilizada na construção de páginas na *web*. Já CSS é um mecanismo para adicionar estilo e personalização a um documento *web*. Por fim, *JavaScript* é a linguagem de programação interpretada para as páginas *web*. Juntas, essas três tecnologias, dentre outras, alicerçam o desenvolvimento para a *Internet*.

## 3.2. Ferramentas

Nesta seção são descritas as ferramentas CASE's e IDE's utilizadas para a realização deste trabalho.

Ferramentas CASE (*Computer-Aided Software Engineering* ou Engenharia de Software Assistida por Computador), para Weinrich (1999), é uma classificação que abrange todas as ferramentas baseadas em computadores que auxiliam atividades de engenharia de software.

Já as IDEs (*Integrated Development Environment* ou Ambiente de Desenvolvimento Integrado), segundo Chaves and SILVA (2008), são programas de computador que reúnem CASE's, características e ferramentas de apoio ao desenvolvimento de software com o objetivo de agilizar este processo.



### **3.2.1. PhpStorm**

O PhpStorm é uma IDE desenvolvida pela JetBrains distribuída gratuitamente para estudantes. Segundo JetBrains (2017), com ele é possível editar, compilar, depurar, refatorar e moldar o código fonte do PHP. Além disso, o PhpStorm oferece a ferramenta de construção do Diagrama de Classes. Ele foi escolhido por ser gratuita e por oferecer as CASEs necessárias para o desenvolvimento do sistema, mencionadas anteriormente.

### **3.2.2. MySQL Workbench**

Em conformidade com Oracle (2017), a ferramenta CASE MySQL Workbench, desenvolvida pela Oracle, permite a modelagem do banco de dados que usa o SGBD MySQL. Ele é gerado pela interpretação do Diagrama de Tabelas Relacional construído no próprio programa. O MySQL Workbench foi utilizado para o banco de dados, pois é um *software* livre e apresenta vantagens como a portabilidade, estabilidade, alta performance, compatibilidade, confiabilidade e facilidade de manuseio.

### **3.2.3. XAMPP**

De acordo com Friends (2017), o XAMPP é um programa desenvolvido pela Apache Friends que permite criar um servidor local para os testes de implementação antes de se portar para um servidor efetivo. Esse ambiente de desenvolvimento em PHP é muito popular, gratuito, fácil de instalar e de usar. Por isso, o XAMPP foi utilizado para teste de implementação do sistema.

### **3.2.4. Microsoft Visio**

Microsoft Visio é um aplicativo para criação de visão gráfica de sistemas no ambiente Windows. O programa serve, segundo Microsoft (2017), para gerar diagramas de diversos tipos, tais como organogramas, fluxogramas, modelagem de dados (usando UML ou outra notação gráfica qualquer), diagramas de redes, plantas baixas, cartazes, etc. Neste projeto ele foi utilizado para gerar o Diagrama de Casos de Uso.

## **4. Modelagem**

Nesta seção são exibidos os modelos que abstraem semanticamente o sistema Coruja. Podem ser definidos como uma simplificação autoconsistente e completa da realidade. Para Booch et al. (2006), com os modelos, é possível ter uma melhor compreensão do sistema.

Para haver uma visão compreensível do projeto, ou seja, uma projeção da organização e estrutura do modelo do sistema, foi usado o Diagrama de Casos de Uso, o Diagrama de Classes, e o Diagrama de Tabela Relacional. Todos apresentam graficamente seu conjunto de elementos.

## 4.1. Diagrama de Casos de Uso

O Diagrama de Casos de Uso mostra os atores, suas ações e seus relacionamentos no software. Eles são importantes principalmente para organização e modelagem do comportamento do sistema.

Na Figura 2 é apresentado o Diagrama de Casos de Uso do sistema Coruja. Como é possível perceber, existem dois tipos de atores que interagem com o sistema, Estagiário e Pedagogo, sendo este uma especialização do primeiro, isto é, tudo que o Estagiário faz também pode ser feito pelo Pedagogo.

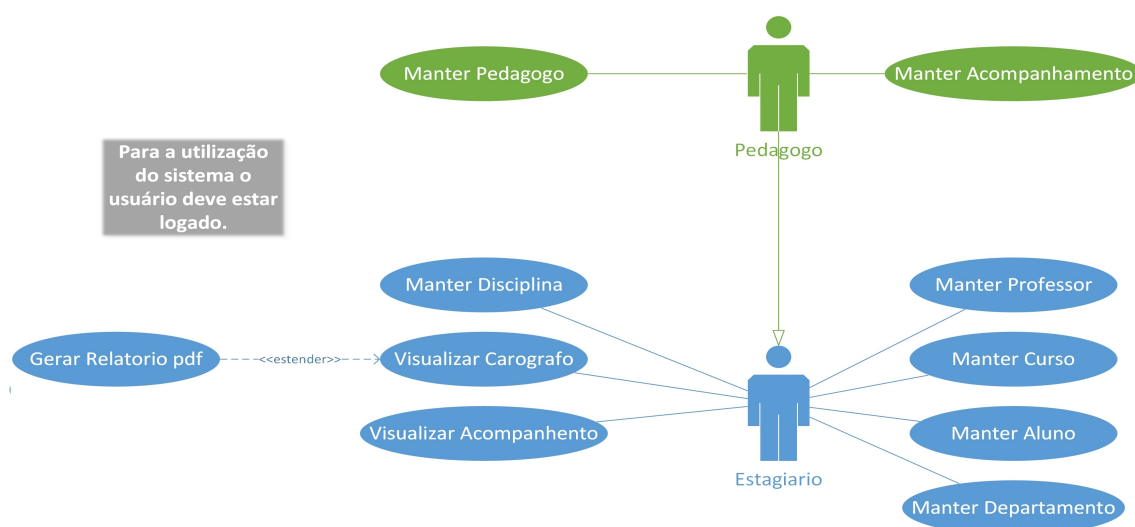


Figura 2. Diagrama de Casos de Uso

Quanto aos casos de uso de cada ator, as atividades do Estagiário são, em suma, de manutenção dos dados, como Disciplina, Professor, Curso, Aluno e Departamento, assim como a Visualização do Carógrafo e dos Acompanhamentos. Os detalhes de cada caso de uso estão especificados a seguir:

- Manter Disciplina: o usuário pode manter, ou seja, cadastrar, atualizar, deletar, e visualizar os dados de cada disciplina associada a cursos, um professor e departamentos.
- Manter Professor: o usuário do sistema pode cadastrar, atualizar, deletar e visualizar os dados de cada professor associado a disciplinas, cursos e departamentos.
- Manter Curso: o usuário do sistema pode cadastrar, atualizar, deletar e visualizar os dados de cada curso associado a um departamento.
- Manter Aluno: o usuário do sistema pode cadastrar, atualizar, deletar e visualizar os dados de cada aluno associado a um curso.
- Manter Departamento: o usuário do sistema pode cadastrar, atualizar, deletar e visualizar cada departamento.
- Visualizar Carógrafo: o usuário pode visualizar o carógrafo gerado automaticamente pelo sistema a partir da manipulação dos dados mantidos e associados nos casos de uso anteriores. Opcionalmente, este caso de uso pode ser estendido, se o usuário desejar para Gerar Relatório PDF, no qual terá como saída o carógrafo em PDF.

- Visualizar Acompanhamento: o usuário pode somente visualizar os atendimentos feitos pelos pedagogos da CP aos docentes e discentes.

O Pedagogo tem duas interações exclusivas que são a Manutenção de Pedagogos e de Acompanhamentos especificados a seguir:

- Manter Pedagogo: o usuário pode cadastrar, atualizar, deletar e visualizar os dados dos pedagogos.
- Manter Acompanhamento: o usuário pode cadastrar, atualizar, deletar e visualizar os dados dos atendimentos aos docentes e discentes.

Cabe ressaltar que, para a utilização do sistema, ambos usuários, pedagogo e estagiário, devem estar devidamente logados com seu usuário e senha.

## 4.2. Diagrama de Classes

O Diagrama de Classes mostra o conjunto de classes e relacionamentos definidas do sistema Coruja. Este tipo de diagrama, para Booch et al. (2006), é o mais utilizado em modelagens de sistemas orientados a objetos, pois apresenta a estrutura estática das classes. Na Figura 3 é apresentado o diagrama das classes do *Model* do sistema Coruja.

As principais classes Aluno, Pedagogo e Atendimento comunicam entre si para que seja possível a correlação das informações necessárias para o atendimento ao aluno. Assim, para cada atendimento que é feito a um discente, o sistema permite o registro das informações acerca das suas solicitações como a disciplina, o professor, o curso, o departamento e o relato do atendimento. Portanto, esses dados são relacionados a fim de que, de forma mais eficiente, o pedagogo tenha um esquema de informações correlatas necessárias para planejar o processo de intervenção e apoio pedagógico ao aluno.

## 4.3. Diagrama de Tabela Relacional

Para Heuser (2009), um modelo de dados é uma descrição dos tipos de informações que estão armazenadas em um banco de dados. O modelo relacional é composto de tabelas ou relações. Sua descrição contém as tabelas que formam o banco de dados, as colunas que a tabela possui e as restrições de integridade. Na Figura 4 é apresentado o Diagrama de Tabela Relacional do sistema Coruja.

A explicação do diagrama também é dada por Heuser (2009). Cada tabela possui um nome que a especifica e identifica. Logo abaixo do nome, estão identificadas as colunas que determinam os atributos e o tipo de dado a serem armazenados de cada registro. Já as restrições de integridade são mantidas pelo SGBD a partir de relacionamentos e chaves que preservam a integridade dos dados para refletirem corretamente a realidade retratada pelo banco de dados. Assim sendo, toda tabela representa uma entidade ou um relacionamento entre duas ou mais entidades no modelo de banco de dados relacional capaz de armazenar dados em um conjunto não ordenado de tuplas - cada instância ou linha de uma tabela - relativos as classes do sistema.

De acordo com a estrutura do Diagrama de Tabela Relacional, percebe-se que as tabelas que armazenam as principais informações do sistemas são Aluno, Pedagogo e Atendimento. Isso porque, por meio delas, é possível obter as informações indispensáveis para a CP e necessárias para o gerenciamento dos atendimentos, dos dados dos estudantes, suas situações acadêmicas e geração do carógrafo.

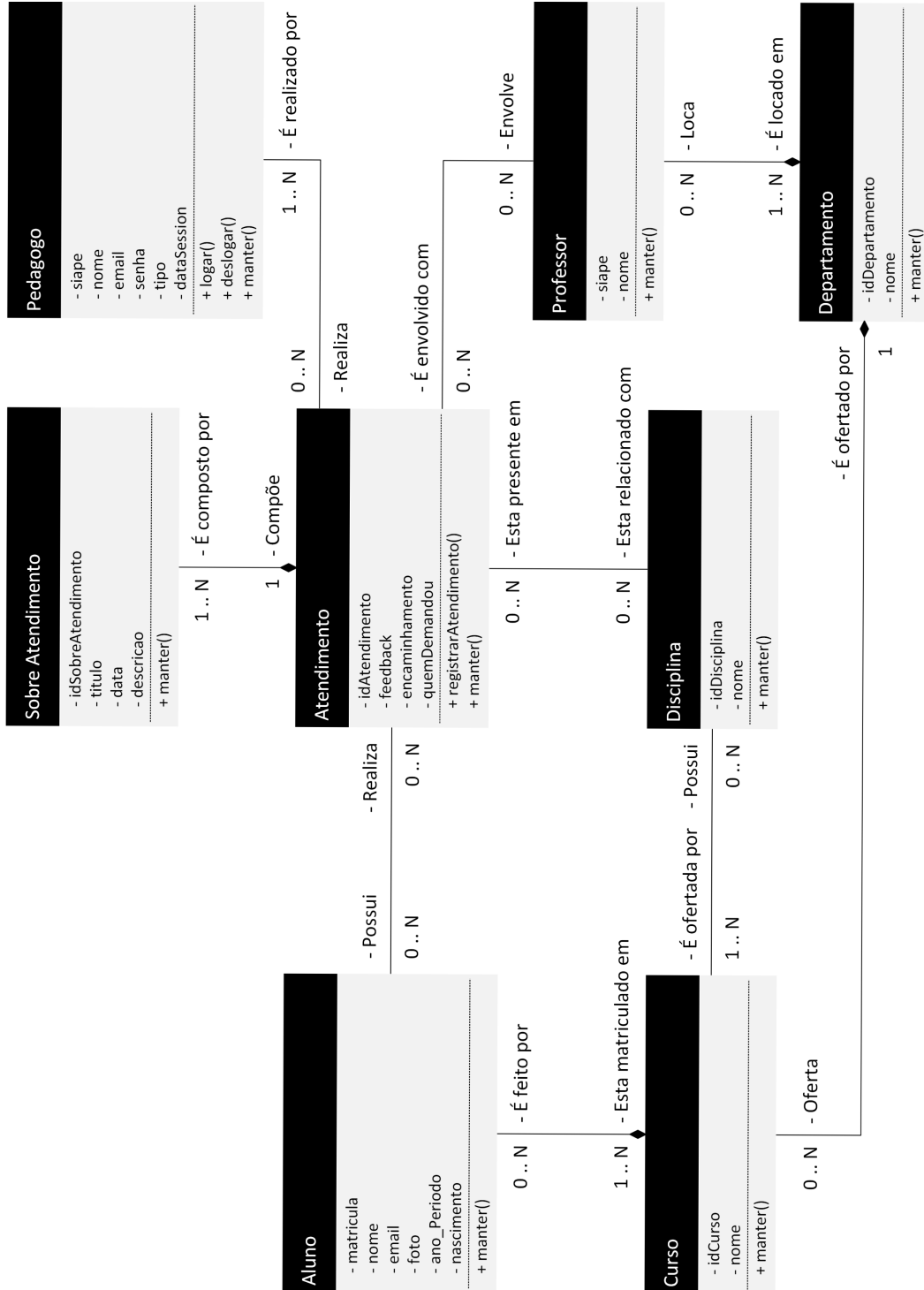


Figura 3. Diagrama de Classes

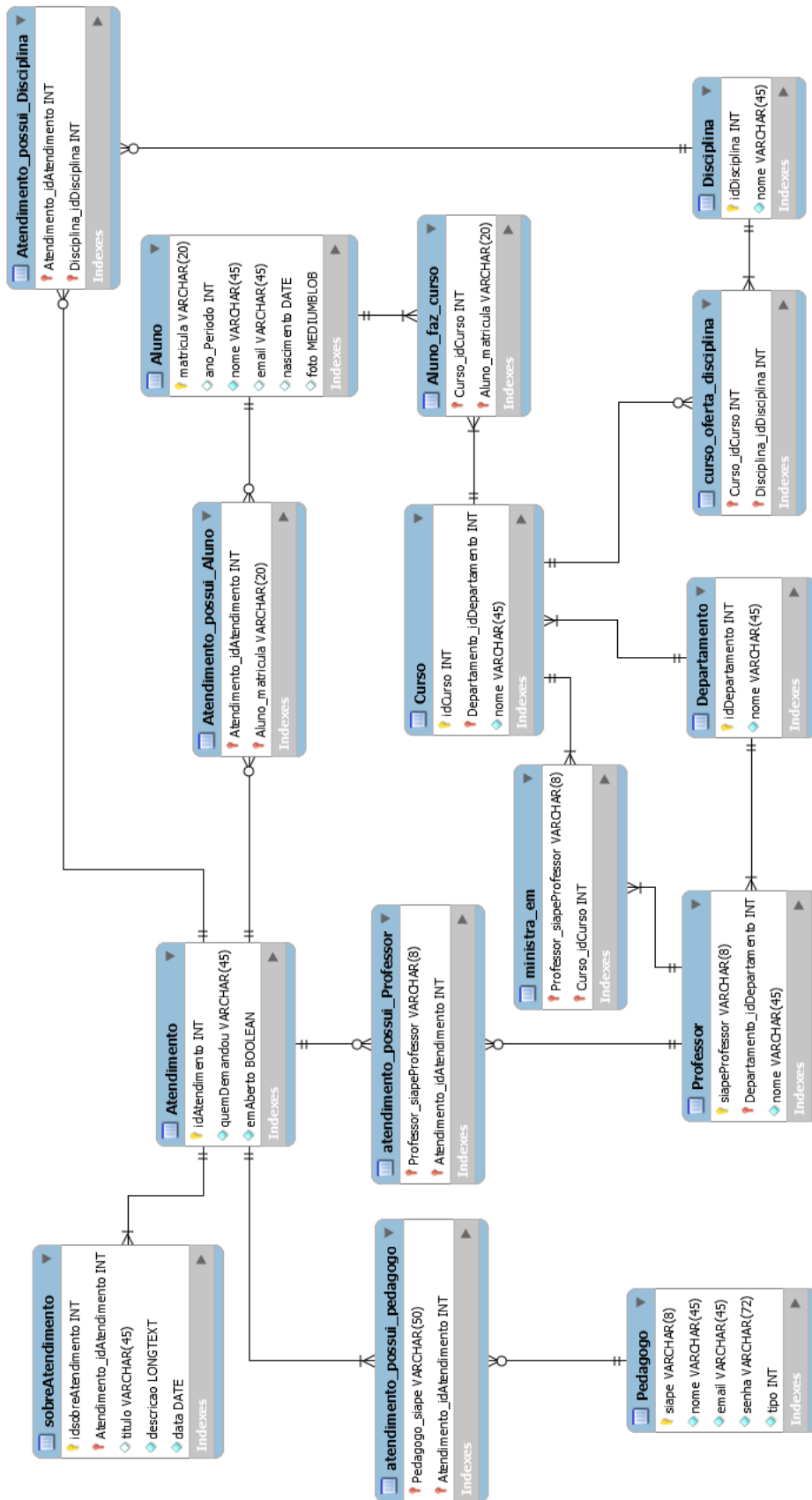
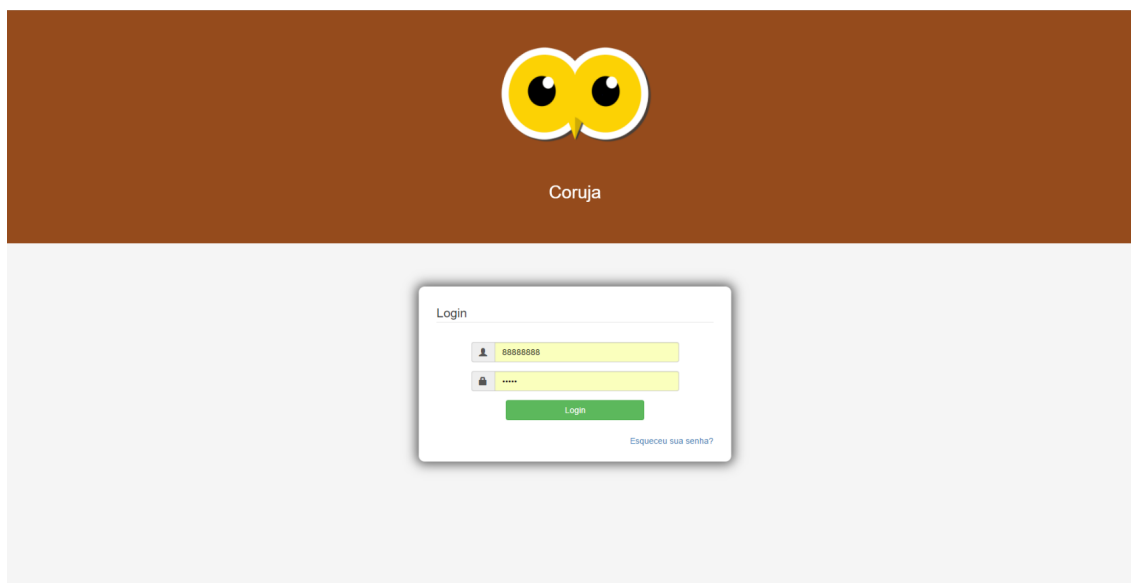


Figura 4. Diagrama de Tabela Relacional

## 5. Desenvolvimento do Sistema

Nesta seção são apresentados os módulos do sistema desenvolvido com algumas de suas principais funcionalidades.

Um dos módulos que o sistema possui é o de *login* em que o usuário se identifica para acessar o sistema. Neste caso, o acesso de cada usuário é feito de acordo com o seu papel no sistema. Para cada papel do usuário é definido um conjunto de diferentes atribuições no sistema. Este módulo é ilustrado pela Figura 5.



**Figura 5. Login**

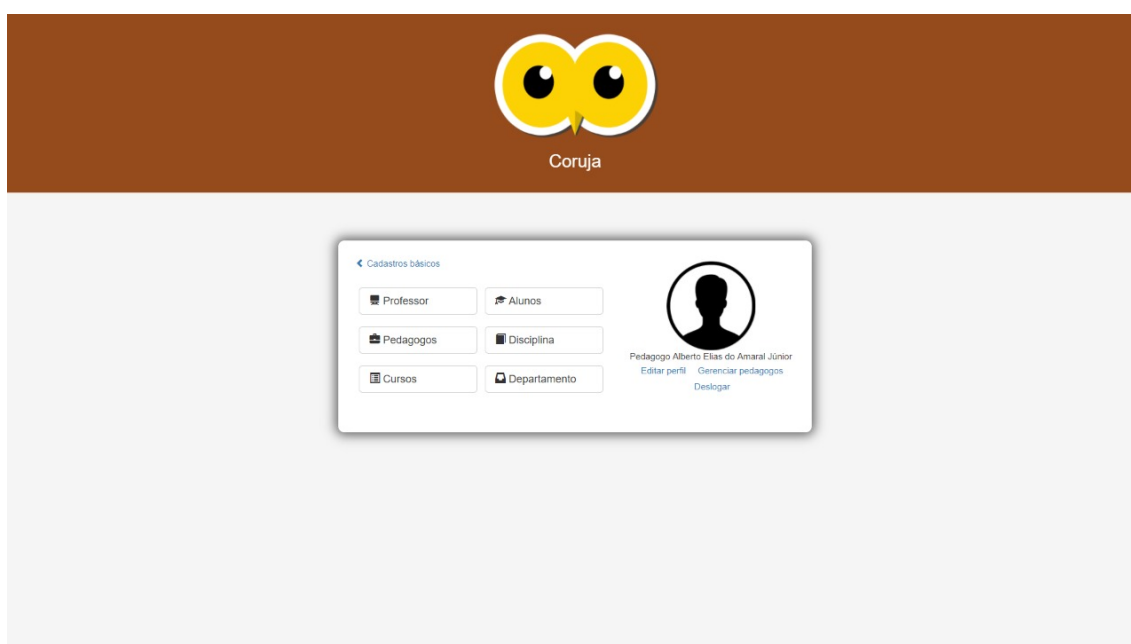
No módulo de menu ou tela inicial do sistema o usuário seleciona qual ação deseja realizar. Como pode ser visto na Figura 6, nesse módulo o usuário pode manter o cadastro de dados, visualizar o carógrafo, registrar um atendimento e visualizar o relatório de atendimentos.

No menu de Cadastros Básicos exposto na Figura 7, o usuário escolhe qual classe deseja gerenciar dentre as de professor, pedagogo, curso, aluno, disciplina e departamento.

Logo após selecionar qual classe será gerenciada, o usuário será redirecionado para o módulo de Gerenciamento dos registros da respectiva classe. Na Figura 8, é mostrado um exemplo da tela de gerenciamento dos dados da classe pedagogo, e, por isso, é exibida uma lista dos registros desse tipo já cadastrados, bem como a opção para o usuário visualizar, editar, excluir cada registro ou adicionar um novo ao sistema. Também é possível realizar uma busca inserindo o nome do registro desejado no campo de busca e o sistema retornará um resultado.



**Figura 6. Menu**



**Figura 7. Cadastros Básicos**

Caso o usuário deseje inserir um novo registro de uma entidade no sistema, ele pode inserir os dados daquela selecionada por ele, como aluno, professor, disciplina, curso, etc. Após o cadastro, esse registro é salvo no banco de dados para posterior processamento na geração do carógrafa e do relatório de atendimento. Esse módulo é ilustrado na Figura 9. Nesse caso, é mostrado como exemplo o cadastro de alunos.

Siape	Nome	Ações
11345678	Leonardo Amorim	<a href="#">Visualizar</a> <a href="#">Editar</a> <a href="#">Excluir</a>
12345678	Anthony Rabelo	<a href="#">Visualizar</a> <a href="#">Editar</a> <a href="#">Excluir</a>

Figura 8. Gerenciamento dos registros

Matricula

Nome

Email

Nascimento

Ano/período

Selecione o curso...

Escolher arquivo N...

Resetar Enviar

Coruja - CEFET MG

Figura 9. Inserção de dados



## 6. Considerações Finais

Após ser implementado para uso na CP, pretende-se que o gerenciamento dos atendimentos aos alunos seja melhorado e a confecção do carógrafo automatizada, cumprindo dessa forma com os requisitos antes delimitados para este trabalho. Para trabalhos futuros, é viável que o Coruja seja implementado em outros *campi* do CEFET-MG ou em outras instituições de ensino. Ademais, é possível também desenvolver uma interface de comunicação com sistemas de gestão escolar para transferir dados dos alunos quanto a suas notas nas disciplinas. Dessa forma, poderá-se gerar gráficos de desempenho que complementam a gestão pedagógica.

Além disso, durante a realização do projeto o grupo observou a importância do trabalho desenvolvido, tendo em vista a inexistência de um sistema com o mesmo propósito ou semelhante às pretensões objetivadas. Os membros deste trabalho também adquiriram conhecimentos específicos com a realização da pesquisa, sobretudo, temas como, educação, pedagogia, administração e gerência de *software*. Em caráter inovador, o sistema Coruja tratou especificamente da área da gestão pedagógica realizada pela CP.

Portanto, o projeto visou a atender as necessidades da CP quanto a um sistema que gerencie os registros e gere relatórios de identificação e de acompanhamento dos estudantes. Dessa forma, os integrantes do grupo esperam contribuir para a melhoria do monitoramento dos estudantes, tanto individualmente quanto coletivamente, além de minimizar esforços gerando o carógrafo automaticamente.

## Referências

- G. E. Bazzotti, Cristiane. A importância do sistema de informação gerencial na gestão empresarial para tomada de decisões. *Ciências Sociais aplicadas em revista*, 6(11), 2006.
- G. Booch, J. Rumbaugh, and I. Jacobson. *UML: guia do usuário*. Elsevier Brasil, 2006.
- A. M. Chaves and G. SILVA. proposta de uma arquitetura de software e funcionalidades para implementação de um ambiente integrado de desenvolvimento para a linguagem php. i jornada científica e vi fipa do cefet bambuí. *Centro Federal de Educação Tecnológica de Bambuí. Bambuí*, page 5, 2008.
- P. Dall'Oglio. *PHP Programando com Orientação a Objetos 3ª Edição*. Novatec Editora, 2015.
- M. F. Damiani, R. S. Rochefort, R. F. de Castro, M. R. Dariz, and S. S. Pinheiro. Discutindo pesquisas do tipo intervenção pedagógica. *Cadernos de Educação*, 2013.
- A. Friends. Xampp. Disponível em: [https://www.apachefriends.org/pt\\_br/index.html](https://www.apachefriends.org/pt_br/index.html) Acesso em 22 setembro de 2017, 2017.
- L. Heloísa. *Dimensões de gestão escolar e suas competências*. Editora Positivo, 2009.
- C. A. Heuser. *Projeto de banco de dados: Volume 4 da Série Livros didáticos informática UFRGS*. Bookman Editora, 2009.
- Inovaise. Galileu sistema de gestão escolar. Disponível em: <https://www.sistemagalileu.com.br/> Acesso em 22 setembro de 2017, 2017.
- JetBrains. PhpStorm. Disponível em: <https://www.jetbrains.com/phpstorm/> Acesso em 22 setembro de 2017, 2017.
- J. C. Libâneo. Organização e gestão da escola. *Goiânia: alternativa*, 2001a.
- J. C. Libâneo. Pedagogia e pedagogos: inquietações e buscas. *Educar em Revista*, 2001b.

- J. C. Libâneo. *Pedagogia e pedagogos, para quê. Cadernos de Pesquisa*, 2007.
- K. Loudon. *Desenvolvimento de grandes aplicações Web*. Novatec, 2010.
- Microsoft. Microsoft visio. Disponível em: <https://products.office.com/pt-br/visio/flowchart-software?tab=tabs-1> Acesso em 22 setembro de 2017, 2017.
- A. Milani. *MySQL-guia do programador*. Novatec Editora, 2007.
- E. M. Miletto and S. de Castro Bertagnolli. *Desenvolvimento de Software II: Introdução ao Desenvolvimento Web com HTML, CSS, JavaScript e PHP-Eixo: Informação e Comunicação-Série Tekne*. Bookman Editora, 2014.
- E. L. Minetto. *Frameworks para desenvolvimento em php*. São Paulo: Novatec Editora Ltda, 2007.
- J. Niederauer. *Integrando php 5 com mysql*. São Paulo. Novatec Editora Ltda, 2005.
- Oracle. Mysql workbench. Disponível em: <https://www.mysql.com/products/workbench/> Acesso em 22 setembro de 2017, 2017.
- R. Pressman and B. Maxim. *Engenharia de Software-8ª Edição*. McGraw Hill Brasil, 2016.
- Qualidata. Q-acadêmico. Disponível em: <http://www2.qualidata.com.br/> Acesso em 22 setembro de 2017, 2017.
- A. Sistemas. Giz. Disponível em: <http://aix.com.br/> Acesso em 22 setembro de 2017, 2017.
- SophiA. Sophia. Disponível em: <http://www.sophia.com.br/> Acesso em 22 setembro de 2017, 2017.
- Sponte. Sponte software de gestão educacional. Disponível em: <http://site.sponte.com.br/> Acesso em 22 setembro de 2017, 2017.
- Vindi. Os 12 principais softwares de gestão escolar. Disponível em: <https://blog.vindi.com.br/os-12-principais-softwares-de-gestao-escolar/> Acesso em 22 setembro de 2017, 2017.
- J. Weinrich. *Software de apoio à avaliação e seleção de ferramentas case baseado na norma iso/iec 14102. Trabalho de Conclusão de Curso, Universidade Regional de Blumenau. Blumenau*, 1999.