

UNIP EaD

Projeto Integrado Multidisciplinar

Cursos Superiores de Tecnologia

**PROJETO DE *DESIGN* PARA CONTROLE DE MATRICULAS DE
CURSOS LIVRES**

Universidade Paulista Unip

Unidade EaD

Outubro de 2019

UNIP EaD

Projeto Integrado Multidisciplinar

Cursos Superiores de Tecnologia

BRUNO AURELIO S. C. RODRIGUES	RA: 0501813	Unidade Araraquara / SP
CHRISTIAN L. S. DE TOLEDO	RA: 1897116	Unidade Capão Redondo / SP
JOABE REZENDE RIBEIRO	RA: 0505093	Unidade Barueri / SP

**PROJETO DE *DESIGN* PARA CONTROLE DE MATRICULAS DE
CUSOS LIVRES**

Trabalho teórico apresentado como único requisito à
Disciplina de Projeto Integrado Multidisciplinar VII
(4º Bimestre), curso de Análise e Desenvolvimento de
Sistemas.

Orientador(a): Prof. Cassiano Gunji.

Curso: Análise e Desenvolvimento de Sistemas 4º Bimestre

Universidade Paulista Unip

Unidade EaD

Outubro de 2019

Resumo

As funcionalidades abordadas por esse projeto, traz uma visão ampla de como um sistema de *software* deve ser projetado, mostrando a importância e o auxílio de um bom *design* de *software* e as ferramentas utilizadas para facilitar a vida do programador, desde a fase de aplicação do sistema e a sua fácil compreensão para facilitação da criação da codificação.

O projeto utiliza como parâmetros desde a sua fase de análise, diagramas detalhados do sistema, que tem como objetivo mostrar o comportamento de um sistema para controle de matrícula de cursos livres, e a importância de se usar um padrão de arquitetura de *software*, pois na fase de projeto começamos a nos preocupar com a arquitetura da aplicação e neste caso usaremos o MVC (*Model-View-Controller*), para a modelagem da linguagem orientada a objetos C#, trazendo conceito da utilização de padrão do projeto, juntamente com suas características e seu uso, mostrando que o MVC pode facilitar em muito a manutenção do código e sua reutilização.

Palavras-chave: Diagrama; Arquitetura; Sistema; Estrutura.

Abstract

The functionalities covered by this project, brings a broad view of how a software system should be designed, showing the importance and aid of a good software design and the tools used to make the programmer's life easier, from the application phase of the program. system and its easy comprehension to facilitate coding creation.

The project uses as parameters from its analysis phase, detailed system diagrams, which aims to show the behavior of a system to control free course enrollment, and the importance of using a software architecture standard, because in the In the design phase we start to worry about the architecture of the application and in this case we will use the MVC (Model-View-Controller), for modeling object oriented language C #, bringing concept of the use of design pattern along with its features and its use, showing that MVC can greatly facilitate code maintenance and reuse.

Keywords: Diagram; Architecture; System; Structure.

Sumário

Introdução.....	4
Atores e contexto de uso	5
Diagrama de casos de uso	8
Diagrama de Sequência de Implementação.....	10
<i>Model View Controller</i> (MVC)	11
Desenho da Arquitetura MVC.....	13
Requisitos Funcionais	15
Requisitos não funcionais.....	18
Diagrama de Classe de Implantação - Manter Curso.....	20
Descrição dos casos de uso – Manter Curso	21
Diagrama de sequência – Manter Curso – Excluir curso no sistema.....	28
Diagrama de classe de implantação Manter Aluno	29
Caso de uso – Manter Aluno	30
Diagrama de sequência de implantação - Manter aluno – Excluir Aluno.....	37
Diagrama de classe de implantação manter matricula.....	38
Caso de uso – Manter Matricula.....	39
Diagrama de classe de implantação efetuar <i>login</i> ao sistema.....	48
Diagrama de sequência de implantação efetuar <i>login</i> ao sistema.....	50
Diagrama de Classe de Implementação – Consultar Matricula.....	51
Diagrama de Classe de Implementação - Gerar Relatório de Matricula.....	53
Diagrama de sequência – Gerar Relatório de Matricula	54
Diagrama de Classe de implementação - Consultar Curso	55
Diagrama de Atividades	56
Diagrama de atividade – calcular valor de curso.....	57
Diagrama de distribuição	58
Diagrama de distribuição ou implantação do sistema.....	59
Conclusão	60

Introdução

Uma empresa de treinamentos resolveu contratar a empresa JBC Software, que vem empreendendo na área de tecnologia trazendo como diferencial padrões de qualidade certificados pela ISO (*International Organization for Standardization*), para construir um sistema de controle de matrículas de cursos de curta duração, com a finalidade de facilitar de forma intuitiva o acesso de alunos, que fariam de forma autônoma a consulta de suas informações e aos atendentes, para ter acesso aos *status* de pagamentos, atividades e cadastrar novos alunos e cursos no sistema.

Para que o projeto do sistema, que será orientado a objetos, fique completo serão elaborados os diagramas necessários entre eles, diagramas de classe, de sequência de atividade de distribuição, e assim sendo capaz de desmembrar e desmistificar o sistema, demonstrado todas as suas funcionalidades e estrutura necessária para o desenvolvimento do sistema.

Após a fase de análise finalizada, o arquiteto responsável pela análise de requisitos do projeto foi realocado nos sendo passado os requisitos, regras de negócio e todos os dados necessários para o desenvolvimento do projeto. Com a bagagem de gestão da qualidade, proporcionando padrões dentro das equipes desta empresa, assim nossa equipe tem plenas condições de seguir com a arquitetura do sistema. Entraremos então na fase de projeto (*Design*), que é a fase responsável em realizar todo planejamento anterior ao desenvolvimento dos linhas de código do sistema, deste modo poderemos transformar tudo em documentação capaz de ser interpretado por clientes, programadores que irão de fato codificar o *software* e todos os *stakeholders*, sendo assim, o projeto deve ser capaz de demonstrar para todos os envolvidos toda a sua capacidade e funcionalidades permitindo então a aprovação desta arquitetura para a efetiva codificação.

Atores e contexto de uso

O sistema será utilizado por atendentes e alunos que tenham sido previamente matriculados por meio de *login* e senha em terminais no interior da instituição.

1. Atendente:

Após validação de *login* e senha o atendente será capaz de consultar cursos disponíveis e cancelamentos por período, cadastrar novo aluno nas áreas de informática e artes, com o código aluno consultar situação cadastral, status de pagamento, cursos em que o aluno está cadastrado e realizar o cancelamento de matrícula em cursos.

2. Aluno/Cliente:

Após validação de *login* e senha o usuário será capaz de consultar sua situação cadastral status de pagamento e cursos em que está cadastrado.

3. Sistema de acesso:

3.1. O sistema de acesso será o controlador responsável pela interface de comunicação do sistema com o banco de dados, controlando o fluxo das informações trafegadas.

3.2. O sistema de acesso, no momento de nova matrícula para alunos já cadastrados deverá conceder um desconto progressivo para alunos que já tenham se cadastrado em outros cursos da seguinte forma, 5% para alunos que já tenham sido matriculados em um curso, 10% para alunos que já tenham sido matriculados em dois cursos e 15% para alunos que já tenham sido matriculados em três cursos ou mais.

4. Sistema externo de pagamento:

O sistema externo de pagamento deverá validar os dados do cartão de crédito e retornar à confirmação de pagamento. Com a confirmação de pagamento o sistema gera um número de matrícula único para o aluno (*código_aluno*), reserva a vaga no curso selecionado, grava todas as informações e envia um *e-mail* de confirmação cadastral com todas as informações selecionadas e fornecidas pelo aluno.

5. Sistema financeiro externo:

Após o usuário ter efetuado a matrícula em um ou mais cursos será gerado um código único de controle, a data de matrícula, o valor da matrícula o *status* de pagamento e o *status* da matrícula e envia os dados ao sistema financeiro.

Especificações de usuários e contexto de uso

O sistema está destinado a pessoas com as seguintes características de usabilidade:

Atributos intelectuais

Habilidades	Requisitos
Experiência com similares	Sistemas ou <i>sites</i> voltados a compras e cadastros
Conhecimento do sistema	Não requerido
Experiência organizacional	Não requerido
Treinamento	Não requerido
Habilidade motora com Teclado	Uso de teclado do computador
Habilidade motora com <i>mouse</i>	Não requerido
Habilidade linguística	Idioma nativo (português)

Tabela 1 – Descrição de atributos intelectuais de usabilidade

Atributos físicos

Característica Física	Requisitos
Habilidade manual	O teclado pode ser operado com uma ou duas mãos
Visão	Teste padrão para visão normal ou corrigida
Audição	Não requerido

Tabela 2 – Descrição de atributos físicos de usabilidade

Especificação de equipamentos e ambientes

Os pontos de acesso serão computadores como terminais de autoatendimento espalhados pela instituição, estes terminais deverão estar conectados a um servidor por meio de uma rede interna para atualização em tempo real das informações acessadas.

Especificações de objetivos do sistema

Este sistema tem como objetivo principal manter o gerenciamento de alunos em cursos de curta duração. Para este objetivo o sistema inclui usuário, permite acesso por meio de *login* e senha, consulta disponibilidade e cancelamento de cursos, realiza controle e reserva de vagas, inclui usuários, cadastra usuário em até dois cursos, cria código único para cada aluno, acessa sistema externo de pagamento, tem comunicação com o sistema financeiro da instituição e envia comprovante de cadastramento por *e-mail*.

Modelagem e diagrama dos casos de uso

Casos de uso, como o próprio nome sugere, são descrições detalhadas e sequenciais em linguagem natural das atividades de interação com atores externos de um sistema sem que aja qualquer especificação técnica de como este funciona, estas interações podem ser do tipo *Include*, *Extend* e Generalização. Atores são iterações externas ao sistema que executam ações e esperam resultados, podendo ser desde um usuário, um sistema de pagamento, uma impressora, etc.

Diagrama de casos de uso

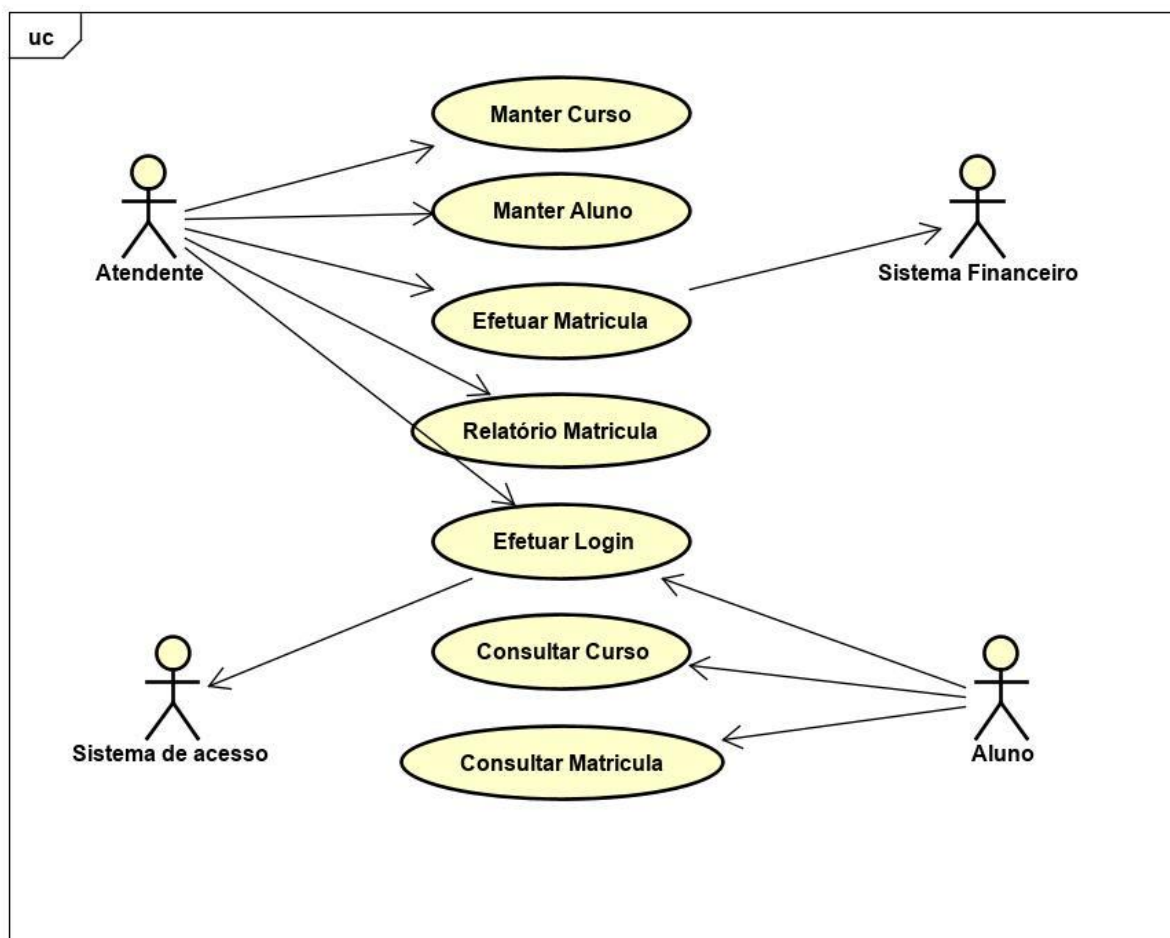


Figura 1 – Diagrama de casos de uso, desenvolvido em Astah UML / Fonte: O Autor, 2019

Caso de uso é a descrição de uma sequência de atividades executadas por um agente externo ao sistema sem que sejam revelados detalhes do funcionamento interno ao sistema, por isso dizemos que o caso de uso mostra a visão comportamental externa ao sistema (BEZERRA, 2006).

Quando bem descritos e definidos, os casos de uso espelham o problema e as funcionalidades do sistema, que pode ser utilizado por usuários analistas e desenvolvedores, como uma ferramenta de apoio para que sejam alcançados os objetivos do sistema.

Include:

Coloca a dependência entre o caso de uso incluído ao caso de uso que inclui, exemplo, caso um atendente queira realizar uma matricula deverá consultar a disponibilidade de vagas, quando um caso **A** inclui um caso **B**, sempre que o caso **A** for executado o caso **B** também será e a direção do relacionamento é sempre do lado que está incluindo para o que é incluído.

Extend:

Já neste caso o a extensão é condicional, o que significa que quando o caso **A** for executado o **B** poderá ou não ser executado também, ainda seguindo o exemplo acima, este aluno após ter acesso a informação de disponibilidade do curso poderá ou não concluir a matrícula. A direção do relacionamento é semelhante à do *include* e ocorre do extensor ao estendido.

Generalização:

A Generalização se dá quando reutilizamos o código para outros casos que extrapolam as margens do caso específico, como por exemplo emitir uma nota de compra, neste caso, a emissão é a mesma em vários campos do negócio, este aluno pode comprar além de cursos, materiais didáticos, atividades físicas e outros produtos que a instituição possa oferecer. Desta forma, o uso ideal é reaproveitar o que está pronto e simplesmente especializar para este contexto de uso específico.

A clareza na modelagem dos relacionamentos e o correto desenvolvimento dos cenários traz um melhor entendimento de quem os interpretará impactando diretamente na qualidade do *software*, viabilizando assim a produção com menos defeitos e maior velocidade, dentre outros benefícios.

Diagrama de Classe de Implementação

Os diagramas de implementação modelam a arquitetura física de um sistema. Na UML estes diagramas são reesponsáveis por mostrar os relacionamentos entre os componentes do *software e hardware* no sistema e a distribuição física do processamento. São desenvolvidos normalmente na fase de desenvolvimento de implementação e mostram a organização dos nós existentes em um sistema, os artefatos armazenados em cada nó e outros componentes e elementos que serão implementados. Os nós, representam na UML os recursos computacionais, dispositivos de *hardware*, como impressoras ou sensores e outros dispositivos que serão utilizados pelo sistema, estes nós, podem ser conectados por caminhos de comunicação para descrever estruturas de rede.

Os artefatos são elementos do modelo que representam as entidades físicas de execução em um sistema de *software* como por exemplo arquivos executáveis, bibliotecas, componentes de *software*, documentos e bancos de dados.

Uma especificação de implementação é basicamente um arquivo de configuração como por exemplo um arquivo de texto, que especifica como um artefato será implementado em um nó.

Como os diagramas de implementação se concentram na configuração dos nós de processamento de tempo de execução e de seus componentes e artefatos, é possível utilizar este tipo de diagrama para avaliar as implicações de distribuição e de alocações de recursos.

Os diagramas de implementação são distintos dos de componentes. Um diagrama de implementação mostra os componentes e artefatos em relação ao local em que eles serão utilizados no sistema em quanto o diagrama de componentes define a composição dos componentes no sistema.

Diagrama de Sequência de Implementação

Na *Unified Modeling Language* (UML), o diagrama de sequência, ilustra um grupo de objetos e as mensagens trocadas entre eles em uma interação, cada objeto possui uma linha de vida que fica ativa durante o tempo em que este objeto está trocando mensagens com outro e ilustra ainda a sequência das mensagens trocadas além de suas estruturas de controle entre os objetos.

O diagrama de sequência deve representar uma situação real ou um caso de uso do sistema, traduzindo resumidamente ao nosso contexto de sistema, um aluno e um atendente são objetos, o aluno solicita ao atendente que realize a sua matrícula, este atendente verifica a disponibilidade de vagas e caso haja vagas este realiza a matrícula e retorna ao aluno que sua matrícula foi realizada com sucesso, o diagrama deve representar esta troca de mensagens sem se preocupar com os métodos de cada objeto observando prioritariamente o caminho de sucesso.

O desenvolvimento do diagrama de sequência pode ser dividido nos estágios de análise, *design* e construção, para descrever as interações entre os objetos.

Em Análise, é possível utilizar diagramas de sequência para demonstrar as interações das instâncias de classe para realizar um caso de uso, assim eles ajudam a identificar as classes necessárias para o sistema e as responsabilidades dos objetos nestas interações.

Em *Design*, os diagramas de sequência explicam como o sistema se comporta durante as interações, é possível refinar o diagrama para mostrar como o sistema completa suas interações.

Durante a construção ou o desenvolvimento da arquitetura de um sistema, podemos utilizar os diagramas de sequência para demonstrar o comportamento de padrões e mecanismos de *design* que o sistema utiliza.

Na organização dos objetos, as instancias que participam da interação são posicionadas da esquerda para a direita sem ordem definida, esta disposição fica a critério do desenvolvedor que os posiciona para o melhor entendimento. Já as mensagens são posicionadas entre os participantes em ordem sequencial de cima para baixo e as especificações de execução e início e fim de fluxo de controle aparecem na linha de vida.

Uma mensagem é um elemento que define um tipo específico de comunicação entre as instâncias, ela leva informações de uma instância, representada por uma linha de vida, a outra, representada por outra linha de vida, em uma interação.

É o uso das interações que nos permitem referenciar outras interações existentes e criar uma sequência completa e complexa de interações mais simples e menores para resolver problemas mais complexos e maiores.

Model View Controller (MVC)

Para que a arquitetura seja feita e implantada de forma exata, primeiramente vamos saber o que é o MVC e sua importância para o sistema.

Como já mencionado o MVC É um padrão de arquitetura de aplicações que divide a aplicação em três camadas, sendo elas a visão (*view*), o modelo (*model*), e o controlador (*controller*), sendo uma arquitetura, ou seja, um modelo que define suas estruturas e tem algumas características como, componentes, propriedades dos componentes, relacionamento dos componentes e todos os elementos que fazem parte da estrutura básica padrão do *software* e como estes elementos interagem entre si.

Na prática o MVC se resume em dividir a aplicação em camadas, dessa forma pode-se criar a possibilidade de exibir uma mesma lógica de negócio através de várias interfaces e assim isolar a camada modelo (*model*) das demais camadas do sistema, dessa maneira a implementação do controlador deve permitir que esta camada receba os eventos da interface e os converta em ações no modelo.

A Camada *Model*

Essa camada é responsável pela lógica do sistema, ou seja, a regra de negócio, podendo-se dizer que a camada *Model* é o *software* propriamente dito.

O *Model* (Modelo) encapsula o estado e o comportamento da aplicação além de ser o único componente do MVC que faz interface da aplicação frente à fonte de dados, que normalmente é representada pelo banco de dados da aplicação.

Devido aos bancos serem, em sua esmagadora maioria, banco de dados relacionais, é necessário no modelo, existir um mapeamento dos objetos do *software* orientados a objeto, para as tabelas do banco de dados. Essa técnica é o Mapeamento Objeto-Relacional (ORM – *Object Relational Mapping*).

A camada *model* (Modelo), não fica apenas separada da apresentação, ou seja, essa camada não sabe da existência da camada de apresentação, sendo ela a combinação dos dados e dos métodos, assim o reuso característico da estrutura MVC se dá principalmente nesta camada, sendo o coração e alma do sistema.

A Camada *View*

Essa é a camada responsável pela apresentação, é a interface de representação do modelo, trata-se da fronteira entre usuário e o sistema. A *view* pode da forma mais conveniente, exibir alguns atributos e ocultar outros, atuando como um filtro para os dados do modelo, assim podemos dizer que este componente pode ser considerado o mais flexível do MVC, podendo ser facilmente alterado ou substituído, pois permitem que novas modificações sejam facilmente implementadas, sem afetar em nada a estrutura do sistema.

A Camada *Controller*

A camada *Controller* (controle), é a camada que responsável por controlar a comunicação entre as outras duas camadas, *Model* (modelo) e *View* (visão), podemos dizer que essa camada é a ponte entre as outras duas camadas do MVC, e seu trabalho é controlar as interações que ocorrem a partir do usuário (recebe o input) que trabalha sobre elementos na camada de visão e descobre o que essa entrada significará para o modelo

As entradas que esses componentes recebem são, normalmente, eventos de mouse, entradas de teclado, entre outras e estes eventos por sua vez serão traduzidos em requisições de serviços para outro componente que deverá tratar, assim existe um controlador para cada função da aplicação, mas algumas estratégias podem ser adotadas para que se desenvolva um controle central e se evite código duplicado.

Desenho da Arquitetura MVC

A arquitetura MVC é feita para separar a apresentação dos dados, da lógica de negócio. O desacoplamento desses componentes principais, permite uma reutilização eficiente de código e desenvolvimento paralelo, ou seja, O padrão MVC é um Padrão de Arquitetura, que visa a separação de responsabilidades e nada mais.

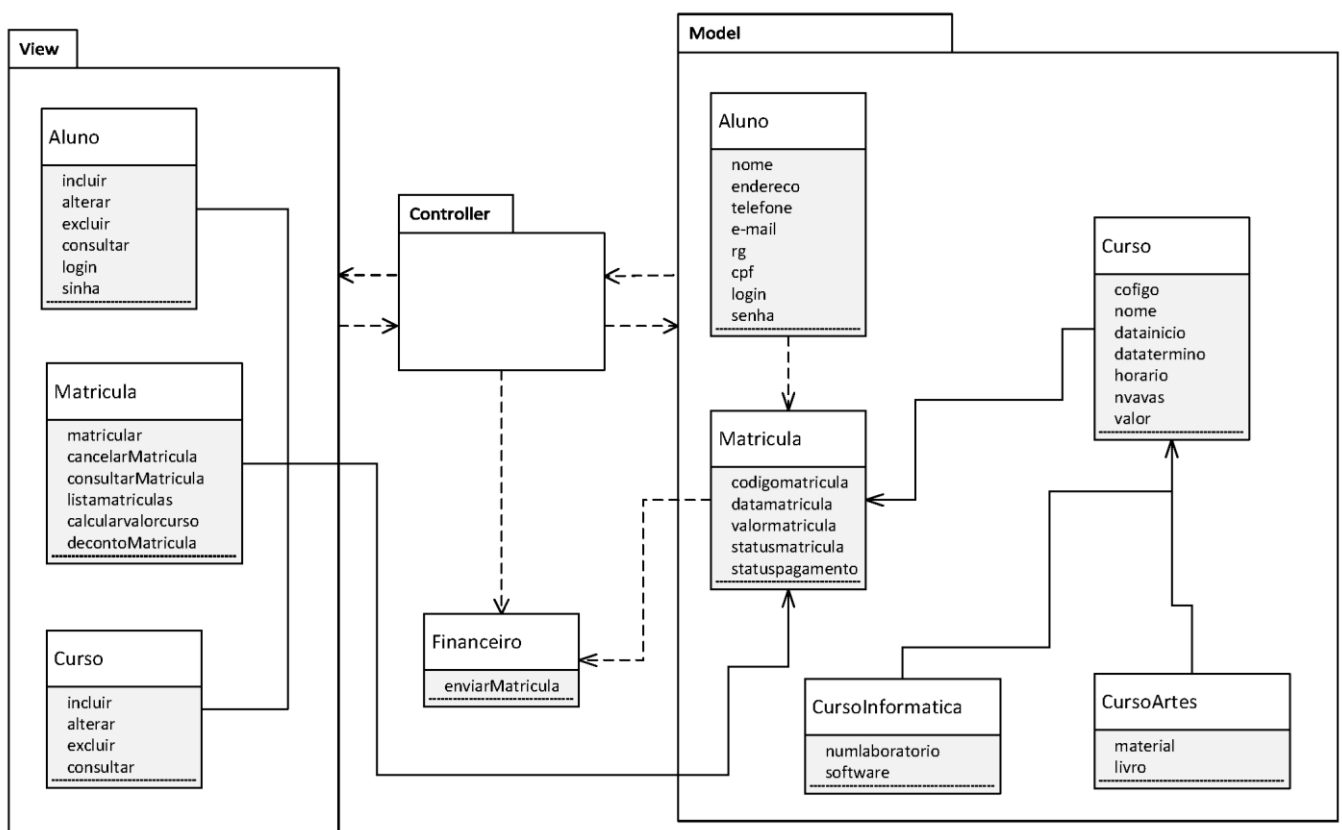


Figura 2 – Arquitetura MVC, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

A representação acima, mostra como é feito uma estruturação MVC, e sua separação, como vemos a camada *view* contém o que será solicitado dentro do sistema, são as interfaces gráficas que serão utilizados pelos alunos e atendentes podendo então cadastrar e incluir alunos.

Já a camada controle é a camada responsável por fazer a ponte entre as camadas *view* e a camada *model* dentro do sistema, como é a camada que faz o intermédio ela garante o envio das informações necessárias para o sistema financeiro, confirmando e efetivando a matrícula do aluno.

Dados de Cadastramento de Curso

Identificação	Especificações
Cd_Curso	Será gerado pelo sistema um código de controle de matrícula caracteres numéricos.
Nm_Curso	Preenchimento obrigatório, com até 50 caracteres alfanuméricos.
Dt_Inicio	Preenchimento obrigatório, com 8 caracteres numéricos no formato DD/MM/AAAA.
Dt_termino	Preenchimento obrigatório, com 8 caracteres numéricos no formato DD/MM/AAAA.
Horario	Preenchimento obrigatório, com 4 caracteres numéricos formato HH:MM
Vagas	Preenchimento obrigatório, com até 2 caracteres numéricos.
Valor	Preenchimento obrigatório, com até 6 caracteres numéricos.

Tabela 3 – Especificações de cadastramento de cursos

Dados de Cadastramento de Aluno

Identificação	Especificações
Nome	Preenchimento obrigatório, com até 100 caracteres e somente caracteres alfabéticos.
Endereço	Preenchimento obrigatório, com até 100 caracteres alfanuméricos. Alfanuméricos e símbolos.
Telefone	Preenchimento obrigatório, com 11 caracteres numéricos.
e-mail	Preenchimento obrigatório, com até 100 caracteres alfanuméricos e símbolos.
RG	Preenchimento obrigatório, com até 15 caracteres alfanuméricos.
CPF	Preenchimento obrigatório, com 11 caracteres numéricos.
<i>Login</i>	Preenchimento obrigatório, entre 6 e 12 caracteres alfanuméricos.
Senha	Preenchimento obrigatório, contendo no mínimo 6 caracteres.
Cd_Aluno	Será gerado pelo sistema um código numérico de controle de matrícula.

Tabela 4 – Especificações de cadastramento de alunos

Requisitos Funcionais

São as necessidades apontadas pelo cliente ou o que ele espera do sistema, por tanto, boa parte da qualidade de *software* está centrada em atender estes requisitos, de certa forma é óbvio descrever as funcionalidades do programa, mas não tão óbvio assim descrever o que o sistema não deve permitir. Pfleeger (2004), define requisitos funcionais como sendo o detalhamento da interação entre o sistema e o seu ambiente. Podemos considerar como parte importante de seu ambiente o usuário. Logo, podemos consideramos o requisito funcional ou RF é o detalhamento da interação do *software* e o usuário.

Uma instituição de ensino resolveu contratar uma empresa para construir um sistema para controlar as matrículas de cursos livres e tem como objetivo realizar o cadastro de alunos, cursos e matrículas de usuários para cursos de curta duração, este sistema deverá ser utilizado por atendentes e alunos matriculados.

Requisito de Usuário X Sistema

<p>Todo o acesso ao sistema é feito em terminais na escola por meio de <i>login</i> e senha.</p>	<p>1.1 O sistema deve permitir o usuário realizar <i>login</i> no sistema.</p> <p>1.2 O acesso ao sistema será feito por meio de <i>login</i> e senha do usuário.</p> <p>1.3 Cada usuário terá acesso apenas a seus dados pessoais.</p> <p>1.4 O sistema deve permitir o aluno cadastrar, consultar ou cancelar matrículas de cursos.</p> <p>1.5 O sistema, no momento do cancelamento o código da matrícula deverá ser enviado para o sistema financeiro.</p>
<p>O atendente pode cadastrar cursos das áreas de informática e artes.</p>	<p>2.1 O sistema deve permitir o atendente se logar ao sistema.</p> <p>2.2 O acesso ao sistema será feito por meio de <i>login</i> e senha do atendente.</p> <p>2.3 O sistema deve permitir o atendente cadastrar, consultar ou cancelar matrículas de cursos.</p> <p>2.4 Em cadastramento o sistema deve permitir cadastrar cursos nas áreas de artes e informática.</p> <p>2.5 Devem existir funções para o atendente consultar os cancelamentos por curso em um determinado período e listar informações de cursos disponíveis.</p>

<p>Todos os tipos de cursos possuem código, nome, data de início, data de término, horário, número de vagas, valor e cadastra alunos informando: nome, endereço, e-mail, RG, CPF, <i>login</i> e senha do aluno.</p>	<p>3.1 O sistema deve permitir selecionar data de início e fim de curso, área de curso e ainda selecionar um ou mais cursos.</p> <p>3.2 O sistema deve exibir tabela com todos os requisitos de cadastramento de novo aluno, nome, endereço, e-mail, RG, CPF.</p> <p>3.3 O sistema deve gerar um código de matrícula e senha de aluno único para cada CPF que deverão ser utilizados como <i>login</i> e senha de acesso. Caso já exista, utilizar <i>login</i> existente.</p> <p>3.4 O sistema deve informar o usuário sobre a disponibilidade, o número de vagas e se disponível o valor do curso selecionado.</p> <p>3.5 A matrícula do aluno em um ou mais cursos deve ser gerado um código único, a data da matrícula, o valor, o <i>status</i> de pagamento e <i>status de</i> matrícula.</p> <p>3.6 O sistema após o cadastro da matrícula, deverá enviar os dados para o sistema financeiro.</p>
--	--

Tabela 5 – Relação de requisitos funcionais

Requisitos não funcionais

Estes estão mais particularmente ligados ao uso do sistema, suas características internas envolvendo especificamente sua parte técnica de como ele será entregue ao usuário, estando relacionados ao desempenho, confiabilidade, usabilidade, segurança, disponibilidade, manutenção e tecnologias envolvidas.

Os requisitos não funcionais (RNF) descrevem restrições sobre os serviços oferecidos pelo sistema de *software* (SOMMERVILLE, 2010).

Tabela de requisitos não funcionais

Numero	Tipo	Identificação	Descrição
RNF 01	Usabilidade	Simplicidade de uso	Um novo aluno deve ser capaz de cadastrar, consultar ou cancelar curso de forma autônoma e intuitiva.
RNF 02	Usabilidade	Portabilidade	Não requerido, o sistema irá operar em terminais internos a instituição.
RNF 03	Usabilidade	Confiabilidade	O sistema deverá ter disponibilidade de 99% do período de aulas.
RNF 04	Ambiente Externo	Forma de pagamento	O sistema se comunicará com sistema externo de pagamentos por meio de cartões de crédito
RNF 05	Segurança	Controle de acesso	Usuários somente terão acesso por meio de <i>login</i> e senha cadastrados previamente.
RNF 06	Segurança	Controle de dados	Os usuários não terão acesso a dados de terceiros
RNF 07	Desempenho	Volume de acesso	O sistema deve ser capaz de suportar o acesso de todos os

			terminais da instituição ao mesmo tempo.
RNF 08	Desempenho	Velocidade	O tempo de execução dos processos de sistema deve ser curto.
RNF 09	Desempenho	Atualização de dados	A atualização do banco de dados deve acontecer em tempo real quando houver um novo cadastramento de aluno em curso.
RNF 10	Organizacionais	Tramites legais	Todos os dados inseridos no sistema não poderão ser fornecidos a terceiros e deverá seguir os padrões de dados já existentes na instituição.
RNF 11	Organizacionais	Desenvolvimento Linguagem	O sistema será desenvolvido na linguagem C#
RNF 12	Organizacionais	Banco de dados	O sistema deverá se comunicar com o banco de dados <i>SQL Server</i>

Tabela 6 – Relação e codificação de requisitos não funcionais

Diagrama de Classe de Implantação - Manter Curso

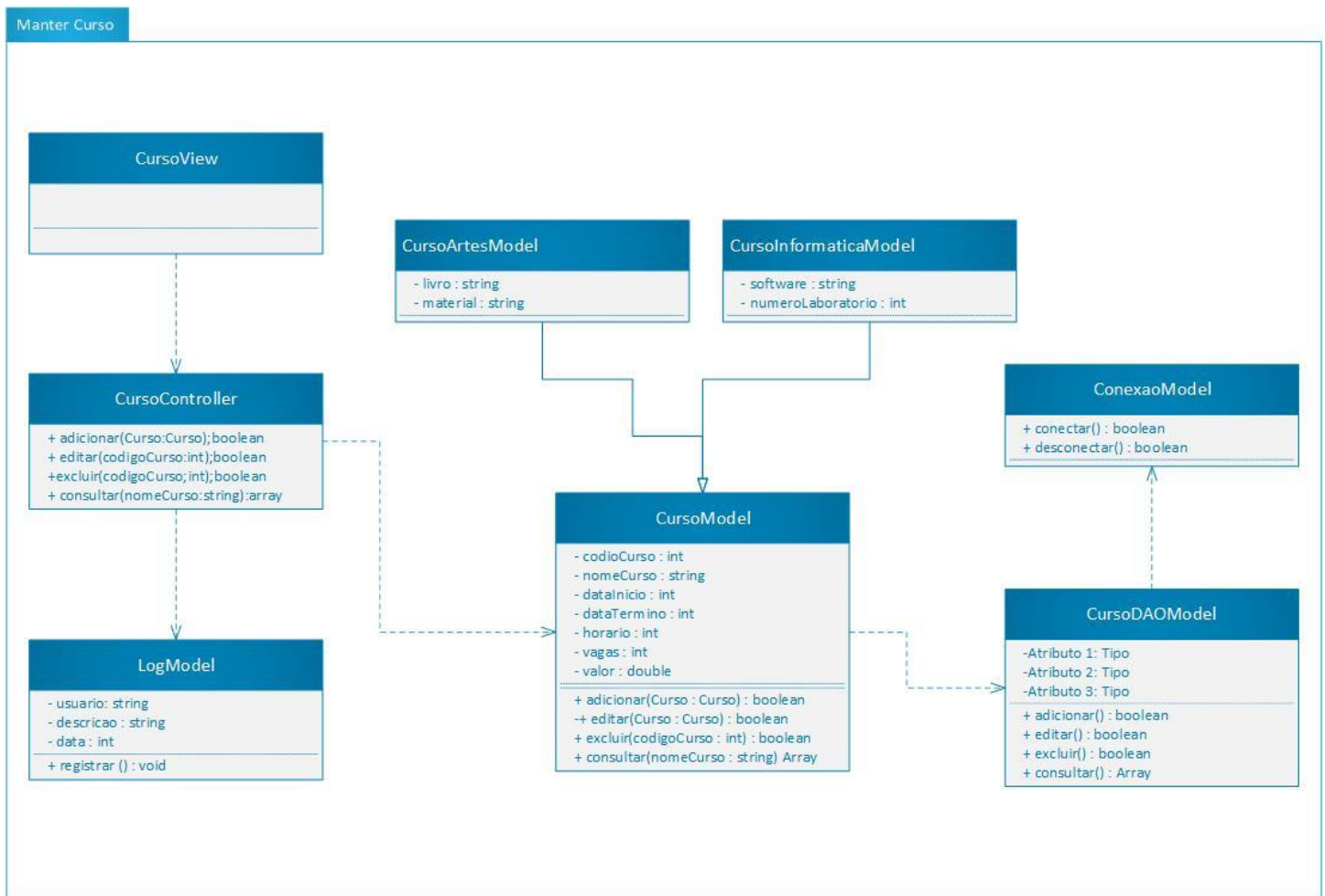


Figura 3 – Diagrama de classes manter curso, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

O diagrama, a seguir, corresponde a atividade de cadastramento de curso, realizada pelo atendente e que abrange duas modalidades distintas, sendo os cursos de informática e de artes, além de suas atribuições referentes ao número do laboratório e dos registros de *software* para o curso de informática e dos materiais e dos nomes dos livros para os cursos de artes.

Descrição dos casos de uso – Manter Curso

Caso de uso 1.1 – Manter curso – Inserir Curso

Identificação do caso de uso	Cadastrar novos cursos
Escopo	Cadastro de curso de curta duração
Objetivo do caso de uso	Permite o atendente inserir novos cursos ao sistema
Ator	Atendente
Interessados	Atendente / Instituição
Pré-condições	A atendente deve estar logado ao sistema
Pós-condições	O sistema recebe um novo curso cadastrado
<p>Fluxo normal:</p> <p>O usuário clica em novo e depois em curso em sua tela inicial do sistema.</p> <p>O sistema solicita o nome do curso</p> <p>O usuário informa o nome do curso e clica em próximo</p> <p>O sistema verifica o nome do curso e pede que o usuário preencha o restante do formulário</p> <p>O usuário fornece a data de início e fim, horário, número de vagas e valor do curso e clica em próximo</p> <p>O sistema exibe as informações do curso para conferência e pede confirmação.</p> <p>O usuário confere os dados e clica em confirma</p> <p>O sistema gera um número de controle de curso e exibe a mensagem de curso cadastrado com sucesso e disponibiliza para usuários matriculados.</p>	
<p>Fluxo Alternativo:</p> <p>1.1 Caso exista um curso com o mesmo nome, o sistema exibe a mensagem de curso já existente e retorna ao passo 3</p> <p>1.1 Caso a data de início seja anterior à data atual exibir a mensagem de data inválida</p> <p>1.2 Caso a data de término seja anterior à data de início do curso, exibir mensagens data de término inválida</p> <p>7.1 Caso o usuário não confirme os dados digitados o sistema retorna ao passo 5</p>	
Requisitos relacionados:	RNF – 03, RNF – 05, RNF – 07, RNF – 08, RNF – 09, RNF – 11, RNF – 12

Tabela 7 - Caso de uso incluir novo curso no sistema

Diagrama de sequência – Manter Curso - Inserir Curso

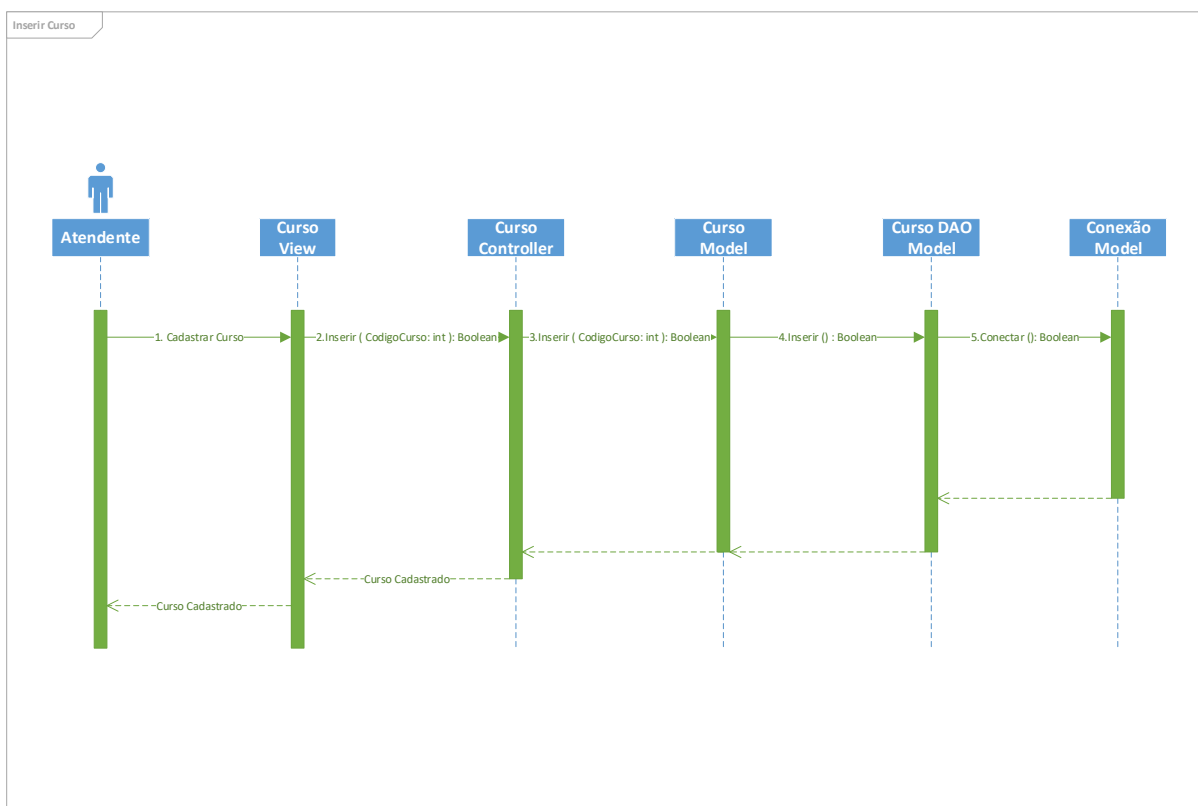


Figura 4 – Diagrama de sequência inserir curso no sistema, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

Caso de uso 1.2 – Manter curso – Consultar Curso

Identificação do caso de uso	Efetuar consulta de cursos disponíveis
Escopo	Cadastro de curso de curta duração
Objetivo do caso de uso	Permite o aluno realizar consulta de cursos disponíveis para matricula.
Ator	Aluno
Interessados	Aluno
Pré-condições	O aluno deve estar <i>logado</i> no sistema
Pós-condições	O sistema exibe na tela todas as informações dos cursos disponíveis para matricula.

Fluxo normal:

O aluno clica na opção consultar cursos em sua tela inicial.

O sistema solicita o período em que o aluno deseja cursar

O aluno informa o período de início do curso

O sistema exibe os cursos disponíveis para o período

O aluno seleciona um curso para mais informações

O sistema exibe todos os detalhes do curso e informa ao aluno que caso deseje se cadastrar procure um atendente e retorna a tela inicial

Fluxo Alternativo:

2.1 Caso o período selecionado seja anterior a data de pesquisa, informar que o período selecionado é inválido.

Requisitos relacionados:

NRF 01 – Simplicidade, NRF 03 – Confiabilidade, NRF 05 – Controle de acesso, NRF 06 – Controle de dados, NRF 08 Velocidade, NRF 10 – Tramites legais.

Tabela 8 - Caso de uso manter aluno – Consultar curso pelo aluno

Diagrama de sequência – Manter Curso - Consulta de curso pelo aluno

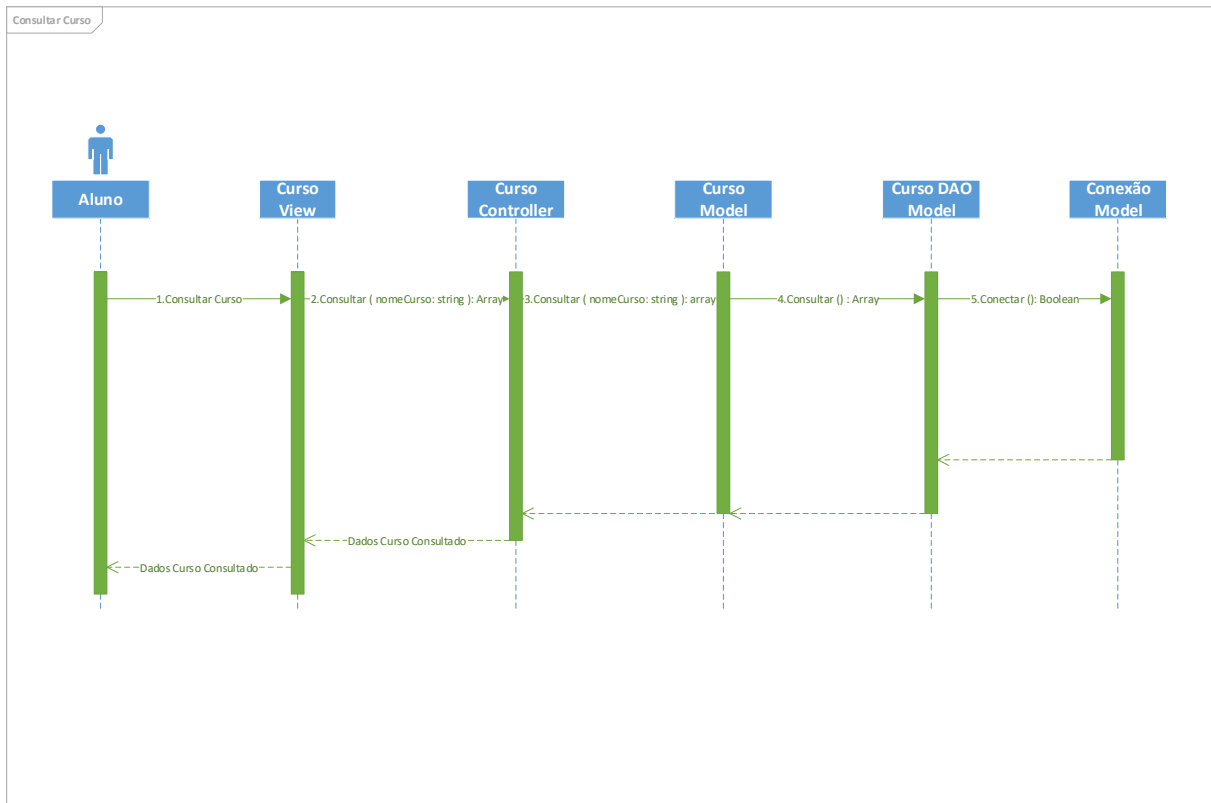


Figura 5 – Diagrama de sequência consultar curso pelo aluno, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

Caso de uso 1.3 – Manter curso – Editar Curso

Identificação do caso de uso	Edição de cursos disponíveis
Escopo	Cadastro de curso de curta duração
Objetivo do caso de uso	Permite ao atendente editar informações cadastradas no sistema
Ator	Atendente
Interessados	Atendente
Pré-condições	O atendente deve estar logado no sistema
Pós-condições	O sistema altera informações dos cursos com sucesso.

<p style="text-align: center;">Fluxo normal:</p> <p>O atendente clica na opção consultar cursos em sua tela inicial.</p> <p>O sistema solicita o código do curso desejado</p> <p>O atendente informa o código do curso</p> <p>O sistema exibe o curso</p> <p>O atendente seleciona a opção de edição de curso</p> <p>O sistema exibe todos os detalhes do curso e permite a edição</p> <p>O atendente edita os dados desejados e clica em salvar</p> <p>O sistema exibe as informações do curso e pede para que o atendente confirme</p> <p>O atendente confirma a edição</p>	
<p style="text-align: center;">Fluxo Alternativo:</p> <p>3.1 Caso o código informado não seja cadastrado, informar que o código selecionado é invalido.</p> <p>8.1 Caso o atendente não confirme a edição o sistema retorna ao passo 3.0</p>	
<p>Requisitos relacionados:</p>	<p>NRF 01 – Simplicidade, NRF 03 – Confiabilidade, NRF 05 – Controle de acesso, NRF 06 – Controle de dados, NRF 08 Velocidade, NRF 10 – Tramites legais.</p>

Tabela 9 - Caso de uso Editar Curso – Consulta pelo (a) Atendente

Diagrama de sequência – Manter Curso – Editar Curso

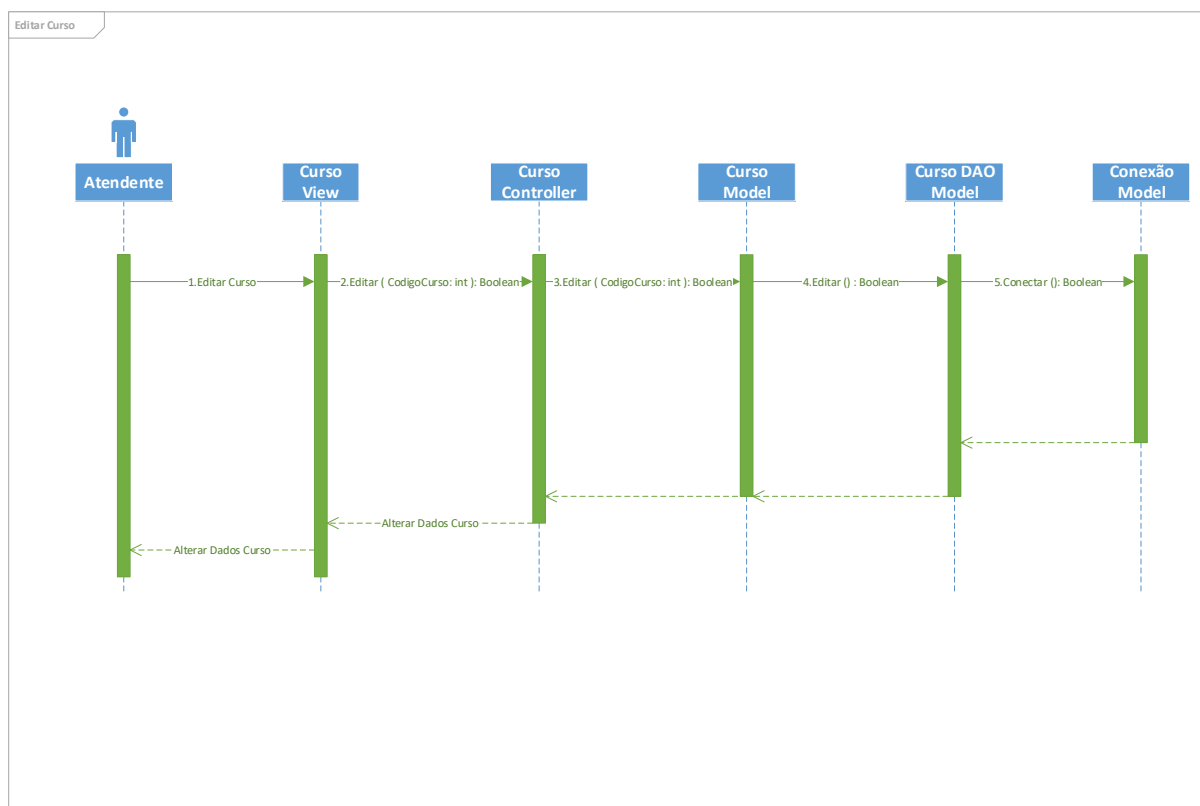


Figura 6 – Diagrama de sequência editar curso, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

Caso de uso 1.4 – Manter Curso – Excluir Curso

Identificação do caso de uso	Excluir cursos do sistema
Escopo	Cadastro de curso de curta duração
Objetivo do caso de uso	Permite o atendente excluir cursos do sistema
Ator	Atendente
Interessados	Atendente / Instituição
Pré-condições	O atendente deve estar logado ao sistema
Pós-condições	O sistema exclui curso cadastrado com sucesso

<p style="text-align: center;">Fluxo normal:</p> <p>O usuário clica na opção excluir cursos.</p> <p>O sistema exibe a relação de cursos ativos no período e pede que o usuário selecione o curso</p> <p>O usuário seleciona o curso que deseja excluir</p> <p>O sistema exibe as informações do curso selecionado e pede confirmação de cancelamento</p> <p>O usuário confere as informações do curso e clica em confirmar</p> <p>O sistema exibe a mensagem de curso excluído com sucesso e retorna ao menu principal</p>	
<p style="text-align: center;">Fluxo Alternativo:</p> <p>Caso aja alunos matriculados no curso selecionado, exibir mensagem de impossível excluir pois existem alunos matriculados</p> <p>5.1 Caso o usuário não confirme os dados o sistema retorna ao ponto 2 do fluxo normal</p>	
<p>Requisitos relacionados:</p>	<p>RNF – 03, RNF – 05, RNF – 07, RNF – 08, RNF – 09, RNF – 11, RNF – 12</p>

Tabela 10 - Caso de uso efetuar exclusão de curso no sistema

Diagrama de sequência – Manter Curso – Excluir curso no sistema

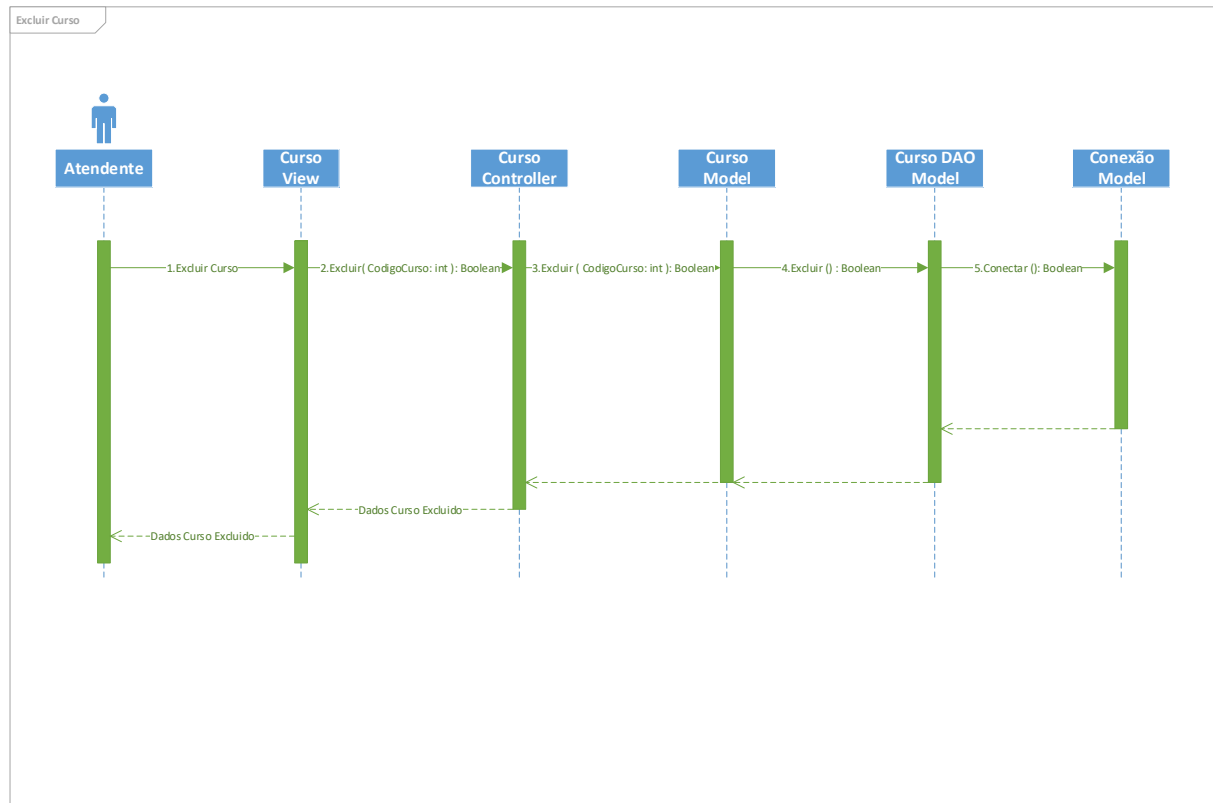


Figura 7 – Diagrama de sequência excluir curso, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

Diagrama de classe de implantação Manter Aluno

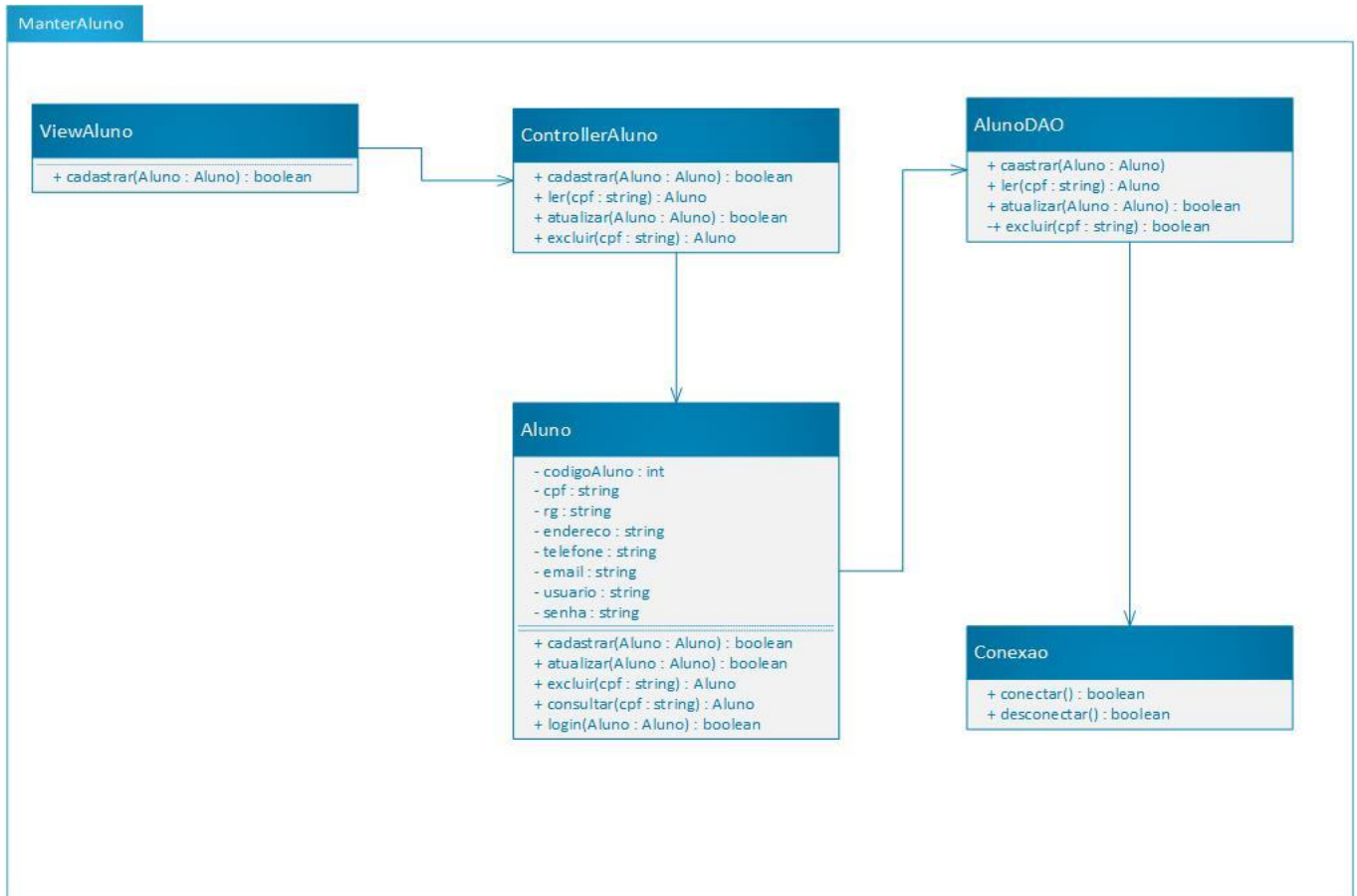


Figura 8 – Diagrama de classes manter aluno, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

Na imagem, temos um diagrama de classe do caso de uso respectivo a capacidade e atividade de cadastramento de alunos. Essa, é uma atividade realizada apenas pelo atendente, que inclui alguns dados pessoais do aluno.

Caso de uso – Manter Aluno

Caso de uso 2.1 – Manter Aluno – Cadastrar aluno

Identificação do caso de uso	Efetuar cadastro de novo aluno
Escopo	Cadastro de curso de curta duração
Objetivo do caso de uso	Permite o atendente cadastrar novo aluno para posterior acesso ao sistema por meio de <i>login</i> e senha.
Ator	Atendente
Interessados	Aluno / Instituição
Pré-condições	Não possuir CPF cadastrado no sistema
Pós-condições	O sistema cadastra as informações do usuário e realiza o <i>login</i> com sucesso.
<p style="text-align: center;">Fluxo normal:</p> <p>O atendente clica no botão novo, depois no botão aluno na tela inicial do sistema.</p> <p>O sistema solicita ao atendente que preencha os campos de cadastro com CPF, RG, nome, data de nascimento, endereço, <i>e-mail</i>, <i>login</i> e senha.</p> <p>O aluno preenche os dados e clica em enviar.</p> <p>O sistema verifica se o CPF não está cadastrado, cria um código de aluno novo (cd_aluno), exibe mensagem de aluno cadastrado com sucesso e direciona o usuário para tela inicial.</p>	
<p style="text-align: center;">Fluxo Alternativo:</p> <p>Caso o cliente deixe campos de preenchimento obrigatório em branco ou preenchido de forma inadequada o sistema exibe mensagem de erro e retorna ao ponto 2 do fluxo normal.</p> <p>Caso o cliente informe um CPF já cadastrado o sistema exibe a mensagem de cliente já cadastrado e retorna a página de <i>login</i>.</p>	
Requisitos relacionados:	RNF – 01, RNF – 03, RNF – 05, RNF – 06, RNF – 07, RNF – 08, RNF – 09, RNF – 10, RNF – 11, RNF – 12

Tabela 11 - Caso de uso efetuar cadastro de novo aluno

Diagrama de sequência de implantação - Manter aluno - Cadastrar Aluno

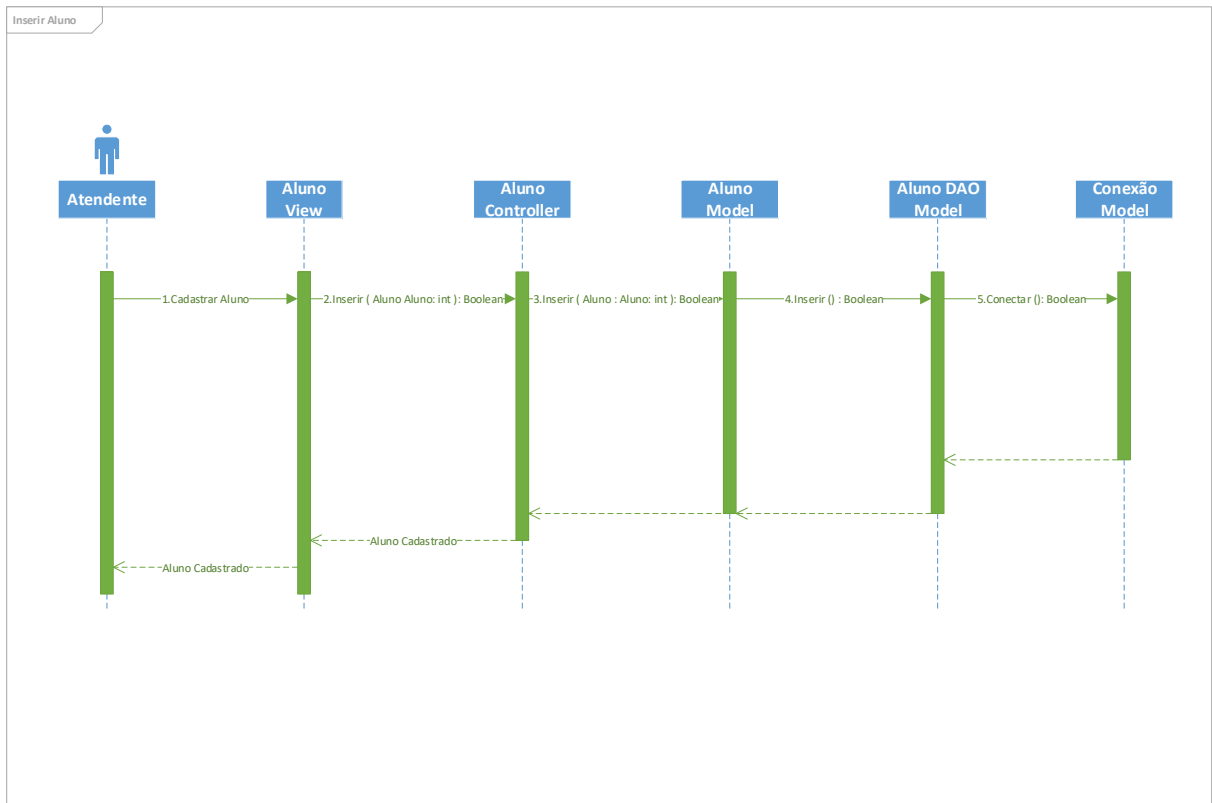


Figura 9 – Diagrama de sequência inserir aluno, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

Caso de uso 2.2 – Manter Aluno – Consultar aluno

Identificação do caso de uso	Efetuar consulta de <i>status</i> de matrícula
Escopo	Cadastro de curso de curta duração
Objetivo do caso de uso	Permite o atendente ter acesso às informações de cadastramento e de pagamento do aluno consultado.
Ator	Atendente
Interessados	Atendente
Pré-condições	O atendente deve estar logado ao sistema
Pós-condições	O sistema exibe na tela todas as informações do usuário consultado.

Fluxo normal:

- . O atendente clica na opção consultar em sua tela inicial.
- . O sistema solicita o CPF do aluno a ser consultado
- . O atendente informa o CPF
- . O sistema exibe as informações dos cursos em que o aluno está ativo e o *status* de pagamento e a opção de retornar a tela inicial.
- . O atendente clica em retornar a tela inicial
- . O sistema retorna a tela inicial

Fluxo Alternativo:

2.1 Caso o aluno não tenha se matriculado em nenhum curso o sistema exibe esta mensagem e retorna a tela inicial

Requisitos relacionados:

NRF 01 – Simplicidade, NRF 03 – Confiabilidade, NRF 05 – Controle de acesso, NRF 06 – Controle de dados, NRF 08 Velocidade, NRF 10 – Tramites legais.

Tabela 12 - Caso de uso consulta de aluno pelo atendente

Diagrama de sequência de implantação - Manter Aluno - Consultar Aluno

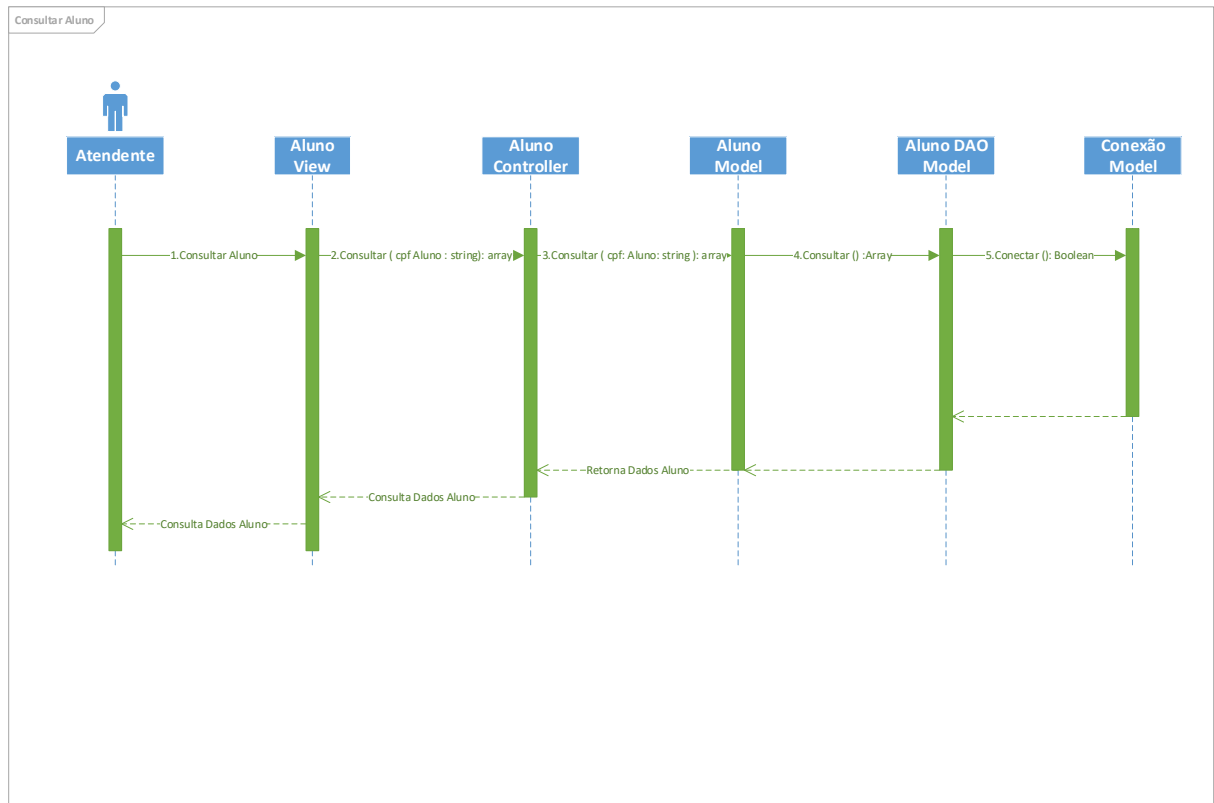


Figura 10 – Diagrama de sequência consultar aluno, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

Caso de uso 2.3 – Manter Aluno – Editar Aluno

Identificação do caso de uso	Efetuar Edição de cadastro de aluno
Escopo	Cadastro de curso de curta duração
Objetivo do caso de uso	Permite o atendente ter acesso às informações e editar cadastro do aluno
Ator	Atendente
Interessados	Atendente
Pré-condições	O atendente deve estar logado ao sistema
Pós-condições	O sistema exibe na tela todas as informações do aluno atualizadas.
Fluxo normal: O atendente clica na opção consultar em sua tela inicial. O sistema solicita o CPF do aluno a ser consultado O atendente informa o CPF O sistema exibe as informações dos cursos em que o aluno está ativo e o <i>status</i> de pagamento e a opção de retornar a tela inicial. O atendente clica em editar cadastro O sistema retorna a tabela com as informações cadastrais do aluno O atendente realiza as alterações necessárias e clica em salvar O sistema exibe todas as informações cadastradas do aluno e solicita a confirmação da edição. O atendente confirma a edição.	
Fluxo Alternativo: 3.1 Caso o aluno não seja cadastrado, exibir mensagem de aluno inexistente 8.1 Caso o atendente não confirme a edição o sistema retorna ao ponto 2.0.	
Requisitos relacionados:	NRF 01 – Simplicidade, NRF 03 – Confiabilidade, NRF 05 – Controle de acesso, NRF 06 – Controle de dados, NRF 08 Velocidade, NRF 10 – Tramites legais.

Tabela 13 – Caso de uso edição de cadastro de aluno pelo atendente

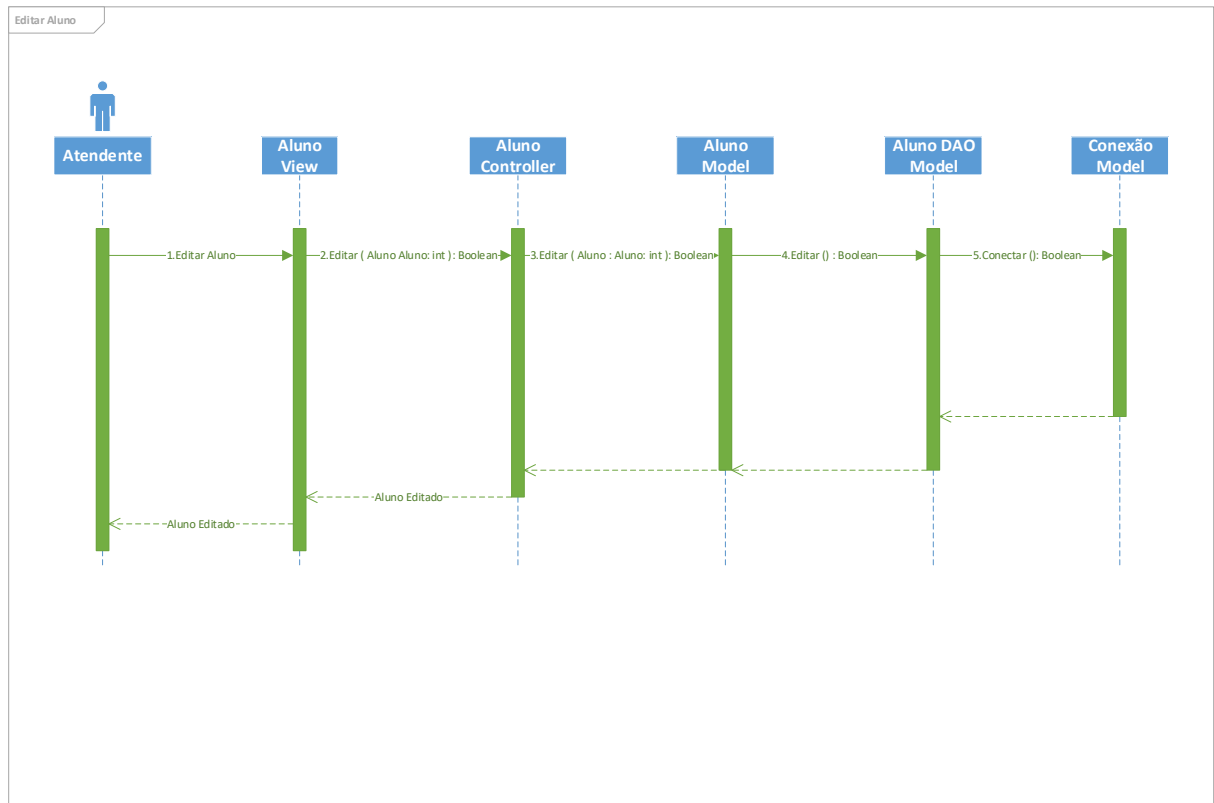
Diagrama de sequência de implantação – Manter aluno – Editar cadastro de Aluno

Figura 11 – Diagrama de sequência editar cadastro aluno, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

Caso de uso 2.4 – Manter Aluno – Excluir Aluno

Identificação do caso de uso	Excluir aluno do sistema
Escopo	Cadastro de curso de curta duração
Objetivo do caso de uso	Permite o atendente excluir alunos do sistema
Ator	Atendente
Interessados	Atendente / Instituição
Pré-condições	O atendente deve estar logado ao sistema
Pós-condições	O sistema exclui aluno cadastrado com sucesso
Fluxo normal: 1. O usuário clica na opção excluir e seleciona a opção aluno em seu menu principal. 2. O sistema solicita o CPF do aluno a ser excluído do sistema 3. O Atendente informa o CPF do aluno a ser deletado 4. O sistema exibe as informações do aluno selecionado e pede confirmação da exclusão 5. O atendente confere as informações do aluno e clica em confirmar 6. O sistema exibe a mensagem de aluno excluído com sucesso e retorna ao menu principal	
Fluxo Alternativo: 3.2 Caso o CPF informado não esteja cadastrado exibir a mensagem de CPF inválido 5.1 Caso o usuário não confirme os dados o sistema retorna ao ponto 2 do fluxo normal	
Requisitos relacionados:	RNF – 03, RNF – 05, RNF – 07, RNF – 08, RNF – 09, RNF – 11, RNF – 12

Tabela 14 - Caso de uso efetuar exclusão de aluno no sistema

Diagrama de sequência de implantação - Manter aluno – Excluir Aluno

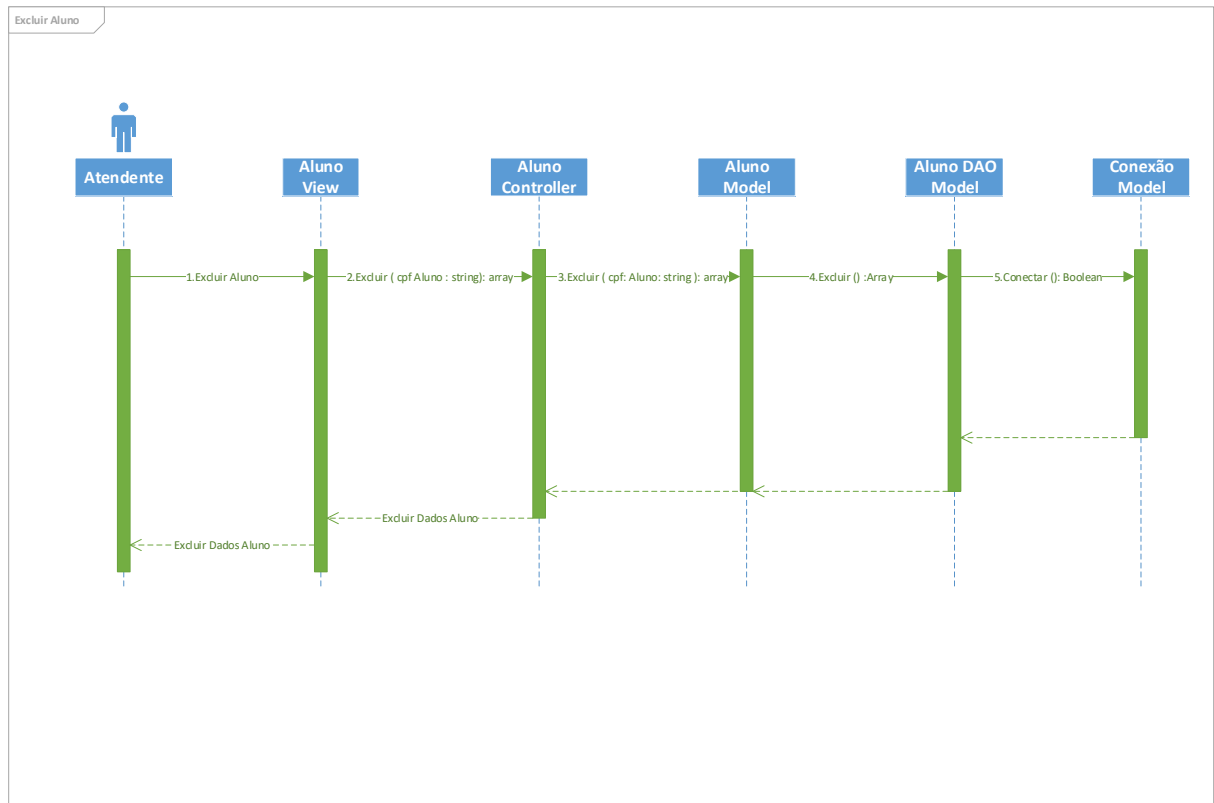


Figura 12 – Diagrama de sequência excluir aluno do sistema, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

Diagrama de classe de implantação manter matricula

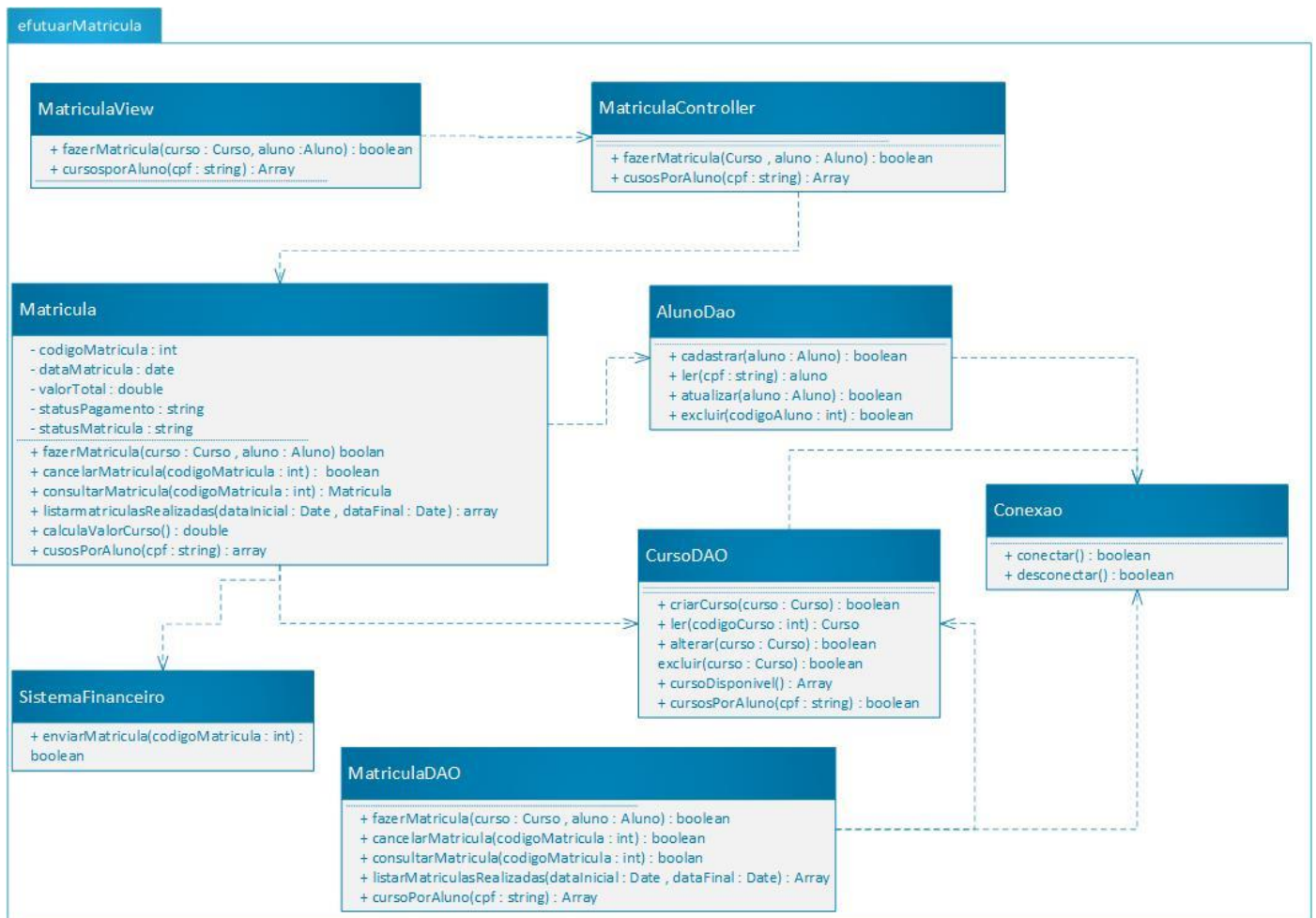


Figura 13 – Diagrama de classes manter matricula, desenvolvido em MySQL / Fonte: O Autor, 2019

Na imagem acima, temos o diagrama de classe de implementação, relacionado com o caso de uso responsável pela ação de efetuar matrícula, em que o atendente poderá matricular o aluno em quantos cursos desejar.

Caso de uso – Manter Matricula

Caso de uso 3.1 – Manter matricula – Cadastrar aluno em curso

Identificação do caso de uso	Cadastrar aluno em curso
Escopo	Cadastro de curso de curta duração
Objetivo do caso de uso	Permite o usuário se matricular em até dois cursos de curta duração.
Ator	Atendente
Interessados	Aluno / Instituição / Sistema de Pagamento / Sistema financeiro
Pré-condições	O usuário deve estar logado ao sistema
Pós-condições	O sistema matricula o aluno nos cursos de seu interesse
Fluxo normal: O usuário clica na opção cursos. O sistema verifica se o usuário já está cadastrado em cursos, consulta os cursos disponíveis e pede ao usuário que selecione a data de interesse. O usuário informa a data de interesse O sistema relaciona os cursos disponíveis para o período e pede ao usuário que selecione seu curso de interesse. O usuário seleciona seu curso O sistema informa ao usuário que ele pode selecionar dois cursos e pergunta se ele deseja selecionar mais um curso ou finalizar matricula O sistema exibe as informações selecionados pelo aluno e pede sua confirmação O sistema direciona o aluno ao sistema externo de pagamento O sistema externo de pagamento retorna com confirmação de pagamento O sistema reserva a vaga do aluno no curso gera um número de controle e envia as informações da matricula ao sistema financeiro externo, exibe uma mensagem de boas-vindas ao aluno e envia uma mensagem com suas informações ao <i>e-mail</i> cadastrado	
Fluxo Alternativo: 2.1 Caso não aja cursos cadastrados o sistema exibe a mensagem	

<p>2.2 Caso o aluno já esteja cadastrado em dois cursos, exibe a mensagem de opção indisponível</p> <p>6.1 Caso o aluno selecione mais cursos, o sistema contabiliza sua seleção e retorna ao passo 4</p> <p>6.2 Caso o aluno já tenha escolhido dois cursos o sistema somente exibe a mensagem finalizar matricula</p> <p>6.3 O sistema, se o aluno já tiver se cadastrado em cursos anteriormente, contabiliza um desconto de 5% para 1 curso, 10% para 2 cursos e 15% para 3 cursos ou mais.</p> <p>7.1 Caso o aluno não confirme suas escolhas o sistema apaga todas as escolhas e retorna ao passo 6.</p> <p>9.1 Caso o sistema externo não confirme o pagamento, o sistema exibe a mensagem de erro ao realizar ao confirmar pagamento e apaga todas as opções selecionadas</p>	
Requisitos relacionados:	<p>RNF – 01, RNF – 03, RNF – 04, RNF – 05, RNF – 06,</p> <p>RNF – 07, RNF – 08, RNF – 09, RNF – 10, RNF – 11,</p> <p>RNF – 12</p>

Tabela 15 - Caso de uso efetuar matricula em cursos

Diagrama de sequência de implantação - Efetuar matrícula - Cadastrar aluno em curso

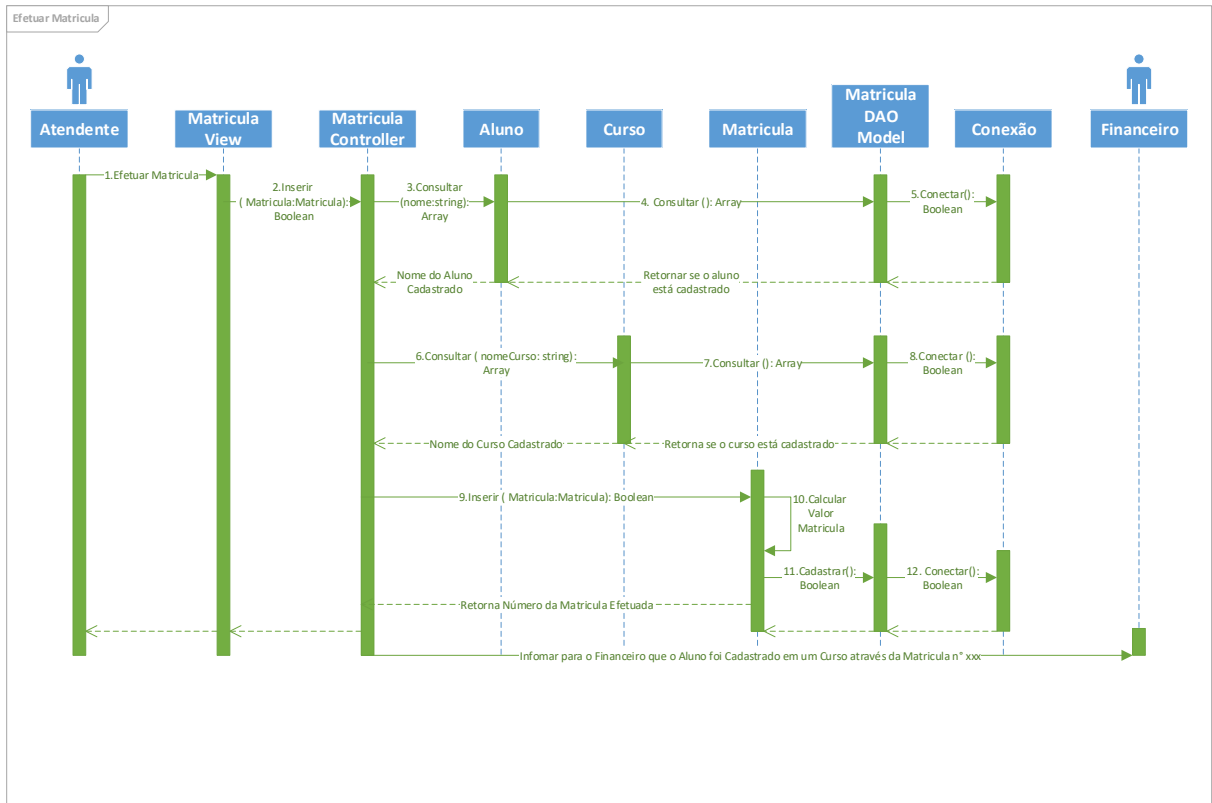


Figura 14 – Diagrama de sequência efetuar matrícula de aluno em curso, desenvolvido em Microsoft Visio /

Fonte: O Autor, 2019

Caso de uso 3.2 – Manter Matrículas – Consulta de matrícula

Identificação do caso de uso	Efetuar consulta de <i>status</i> de matrícula
Escopo	Cadastro de curso de curta duração
Objetivo do caso de uso	Permite o aluno ter acesso às informações de cadastramento, de pagamento e <i>status</i> dos cursos em que está matriculado
Ator	Aluno
Interessados	Aluno
Pré-condições	O aluno deve estar logado ao sistema
Pós-condições	O sistema exibe na tela todas as informações do usuário consultado.
Fluxo normal: O aluno clica na opção consultar <i>status</i> de matrícula em sua tela inicial. O sistema verifica o CPF do aluno a logado, e exibe todas as informações cadastrais, cursos em que o aluno está matriculado ou mesmo já tenha concluído e um botão de retornar a tela inicial. O aluno clica em retornar a tela inicial O sistema retorna a tela inicial	
Fluxo Alternativo: 2.1 Caso o aluno não tenha se matriculado em nenhum curso o sistema exibe esta mensagem e retorna a tela inicial	
Requisitos relacionados:	NRF 01 – Simplicidade, NRF 03 – Confiabilidade, NRF 05 – Controle de acesso, NRF 06 – Controle de dados, NRF 08 Velocidade, NRF 10 – Tramites legais.

Tabela 16 - Caso de uso consulta status de matrícula pelo aluno

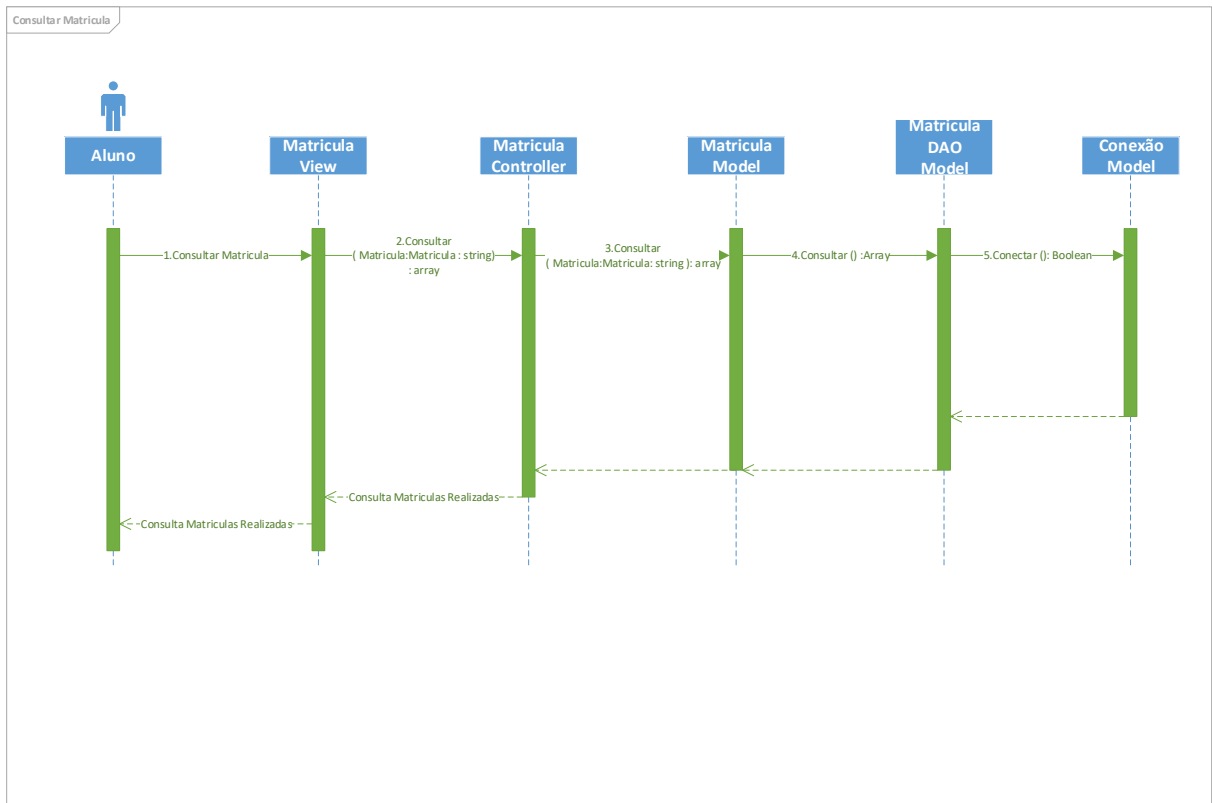
Diagrama de sequência de implantação - Efetuar matricula - Consulta de Matricula

Figura 15 – Diagrama de sequência consultar *status* da matricula, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

Caso de uso 3.3 – Manter Matrículas – Relatório de matrículas por período

Identificação do caso de uso	Consulta de matrículas ativas e inativas por período
Escopo	Cadastro de curso de curta duração
Objetivo do caso de uso	Permite gerenciar as matrículas ativas e inativas por período
Ator	Atendente
Interessados	Atendente / Instituição
Pré-condições	O atendente deve estar logado ao sistema
Pós-condições	O sistema exibe a relação de matrículas ativas e cancelados por um determinado período
Fluxo normal: O atendente clica em consultar <i>status</i> de matrículas por período O sistema solicita o período desejado no formato MM/AAA ao MM/AAA O atendente informa o período desejado O sistema relaciona os números de matrícula, nome dos alunos cursos ativos ou cancelados no período selecionado e exibe a opção de imprimir relatório e retornar ao menu principal	
Fluxo Alternativo: 3.1 Caso o período selecionado seja posterior a data atual, exibir mensagem de período inválido 3.2 Caso o período de início seja posterior ao de fim, exibir mensagem de período inválido 4.1 Caso não haja cancelamentos, exibir mensagem de não existem cancelamentos neste período	
Requisitos relacionados:	RNF – 03, RNF – 05, RNF – 07, RNF – 08, RNF – 09, RNF – 11, RNF – 12

Tabela 17 - Caso de uso consultar matrículas por período

Diagrama de sequência de implantação - Manter matricula – Relatório de matrículas por período

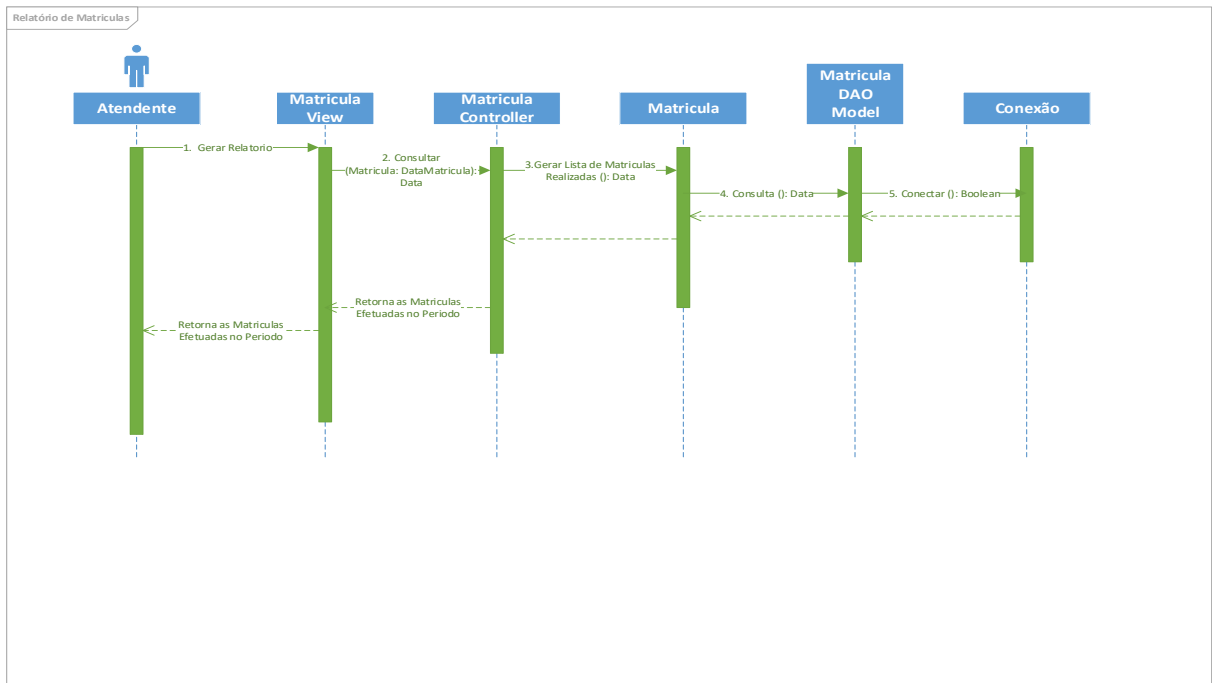


Figura 16 – Diagrama de sequência exibir relatório de matrículas, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

Caso de uso 3.4 – Manter Matrículas – Excluir Matrícula

Identificação do caso de uso	Efetuar cancelamento em curso
Escopo	Cadastro de curso de curta duração
Objetivo do caso de uso	Permite o usuário cancelar suas matrículas ativas em cursos
Ator	Atendente
Interessados	Aluno / Instituição / Sistema financeiro
Pré-condições	O atendente informa o CPF do aluno que deseja cancelar
Pós-condições	O sistema cancela matrícula e envia estas informações ao sistema financeiro externo
Fluxo normal: O atendente clica na opção cancelamento. O sistema verifica e exibe os cursos em que o aluno está com matrícula ativa O atendente seleciona os cursos a serem cancelados e clica em cancelar O sistema exibe as informações dos cursos a serem cancelados e pergunta se deseja realmente cancelar O atendente confirma o cancelamento O sistema exibe uma mensagem de curso cancelado com sucesso, envia o código de cancelamento da matrícula ao sistema externo financeiro, envia um <i>e-mail</i> com as informações de cancelamento ao aluno e retorna a tela inicial	
Fluxo Alternativo: 2.1 Caso o aluno não esteja cadastrado em nenhum curso, exibir a mensagem de não existem cursos para esta matrícula 3.1 Caso o aluno não selecione cursos, exibir a mensagem, favor selecione um curso ou retorne a tela inicial 5.1 Caso o aluno não confirme o cancelamento o sistema não grava as informações e retorna a tela inicial	
Requisitos relacionados:	RNF – 01, RNF – 03, RNF – 05, RNF – 06, RNF – 07, RNF – 08, RNF – 09, RNF – 10, RNF – 11, RNF – 12

Tabela 18 - Caso de uso efetuar cancelamento de matrícula em curso

Diagrama de sequência de implantação - Manter matricula – Cancelamento de Matricula

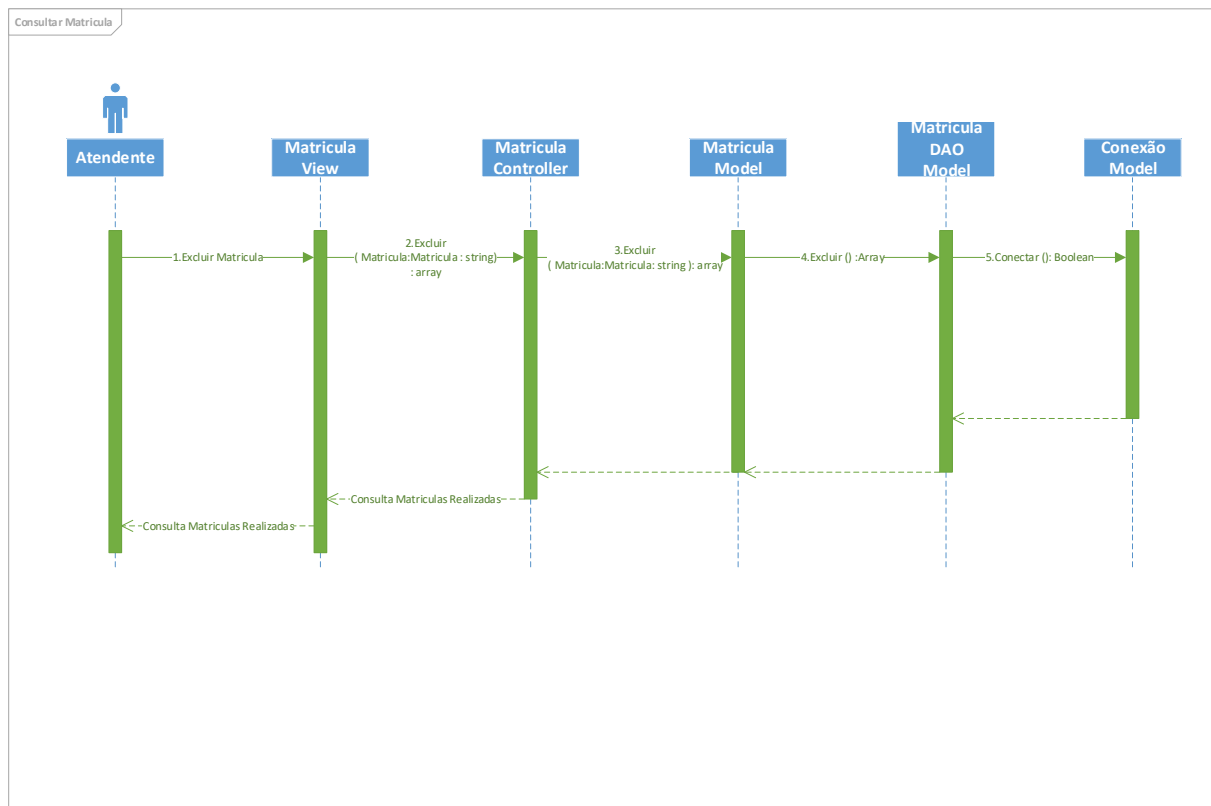


Figura 17 – Diagrama de sequência excluir matricula, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

Diagrama de classe de implantação efetuar login ao sistema

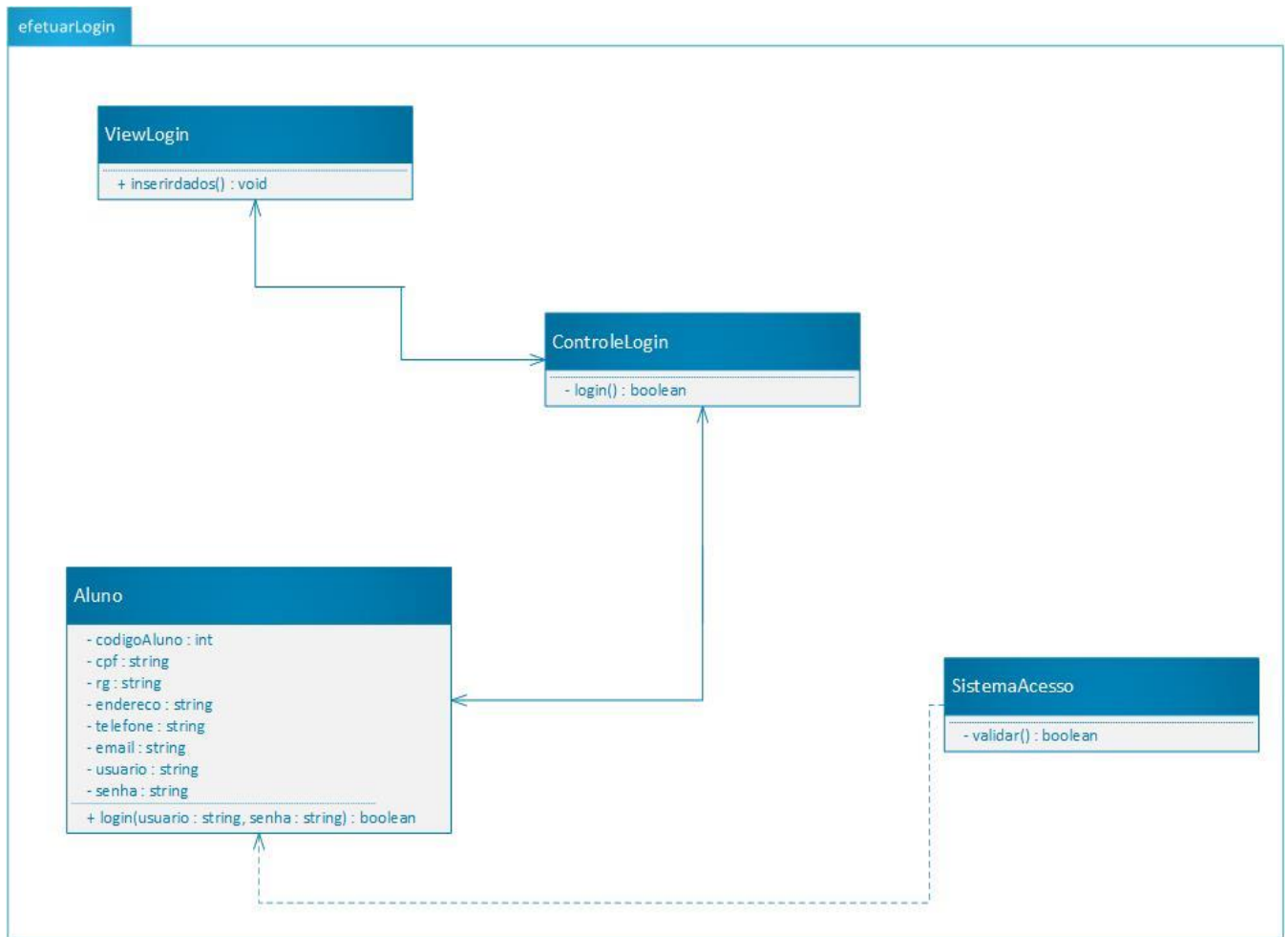


Figura 18 – Diagrama de classes efetuar login, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

Podemos ver o Diagrama de Classe do caso de uso Efetuar Login, nele podemos ver todos as classes e objetos envolvidos no sistema assim como seu sistema de acesso para validar o acesso do usuário no sistema, de acordo com a arquitetura MVC.

Caso de uso – Efetuar *Login*

Caso de uso 4 – Efetuar *Login* – Efetuar acesso ao sistema

Identificação do caso de uso	Efetuar <i>login</i> com senha
Escopo	Cadastro de curso de curta duração
Objetivo do caso de uso	Permite o usuário ter acesso aos dados para o qual é cadastrado dentro do sistema conforme perfil do usuário.
Ator	Atendente / Aluno
Interessados	Atendente / Aluno
Pré-condições	Nenhum usuário deve estar logado ao sistema
Pós-condições	O sistema permite o acesso a conta do usuário exibe uma mensagem de boas-vindas e o direciona a página inicial com acesso aos <i>links</i> conforme o perfil do usuário.
Fluxo normal: O usuário seleciona a opção já sou cadastrado O sistema solicita os dados de acesso do cliente O usuário informa seu <i>login</i> e senha de acesso e clica em entrar O sistema exibe mensagem de boas-vindas, o direciona a tela inicial e exibe as opções de menu do usuário.	
Fluxo Alternativo: 3.1 Caso o usuário insira dados de acesso inválido, o sistema exibe uma mensagem de <i>login</i> invalido e caso não tenha um nome de usuário, clique em não sou cadastrado e retorna ao campo de <i>login</i> e senha	
Requisitos relacionados:	RNF – 01, RNF – 03, RNF – 04, RNF – 05, RNF – 06, RNF – 07, RNF – 08, RNF – 09, RNF – 10, RNF – 11, RNF – 12

Tabela 19 - Caso de uso efetuar *login* com senha

Diagrama de sequência de implantação efetuar *Login* ao sistema

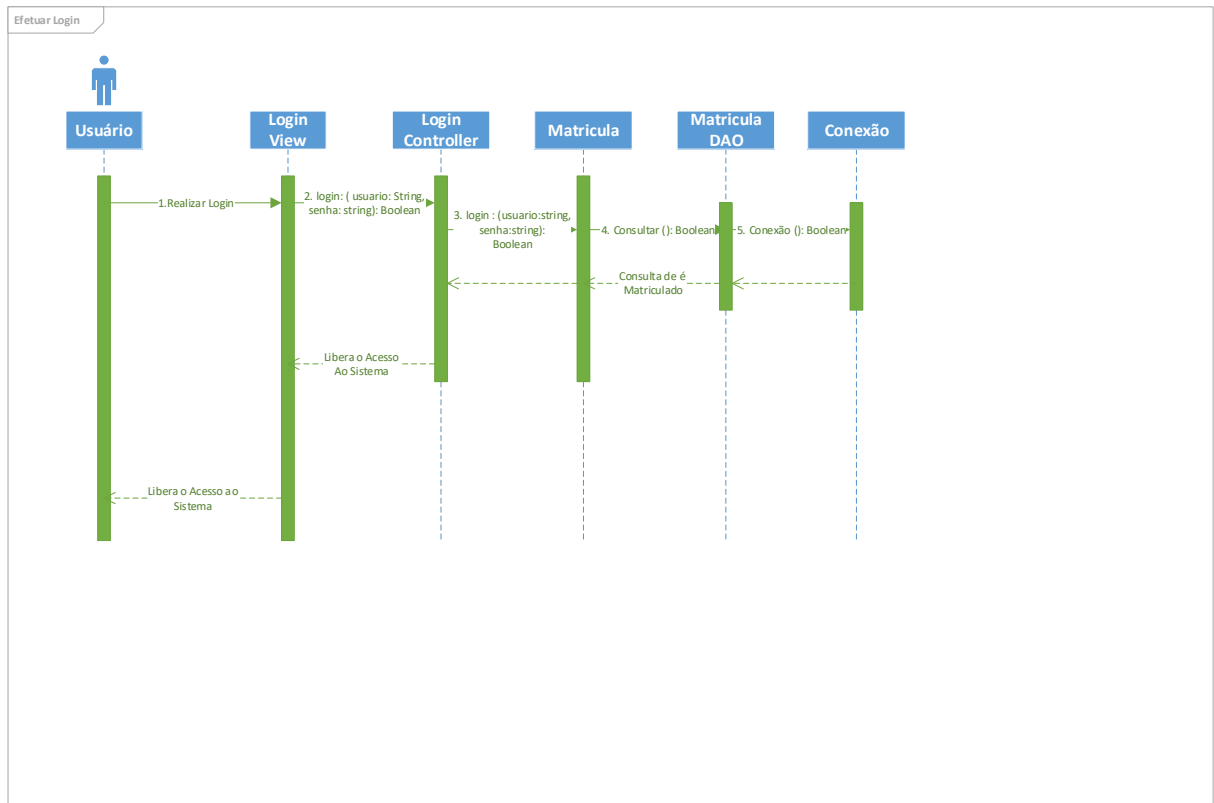


Figura 19 – Diagrama de sequência realizar *login* no sistema, desenvolvido em Microsoft Visio / Fonte: O Autor,

2019

Diagrama de Classe de Implementação – Consultar Matricula

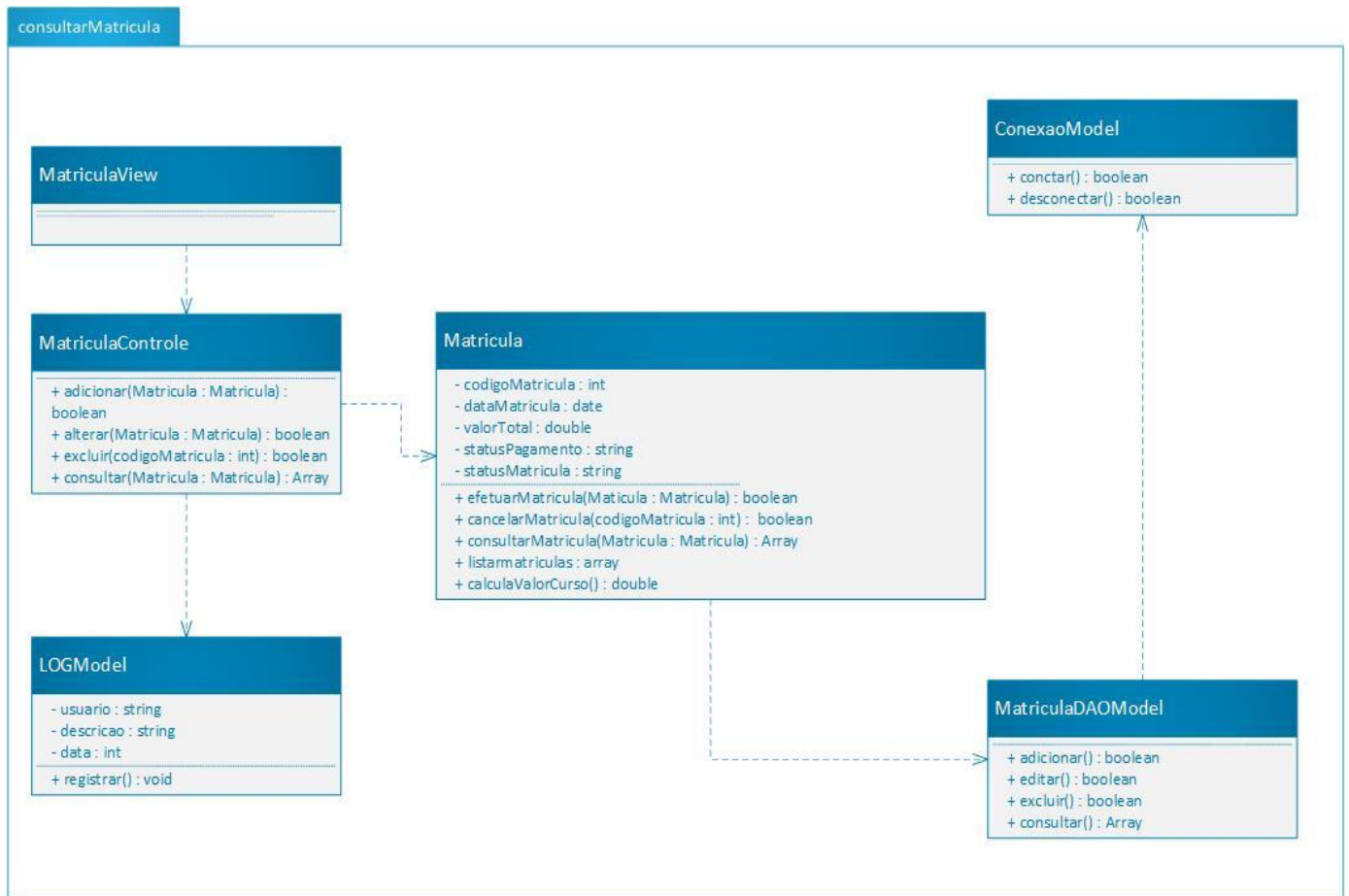


Figura 20 – Diagrama de classes consultar matricula, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

O diagrama de classe, compreende a ação realizada pelo aluno. Com ela, o usuário poderá realizar a consulta de matrículas de cursos que já foram realizados ou que estão em curso, permitindo uma maior capacidade de informação obtida.

Diagrama de Sequência – Consultar matricula

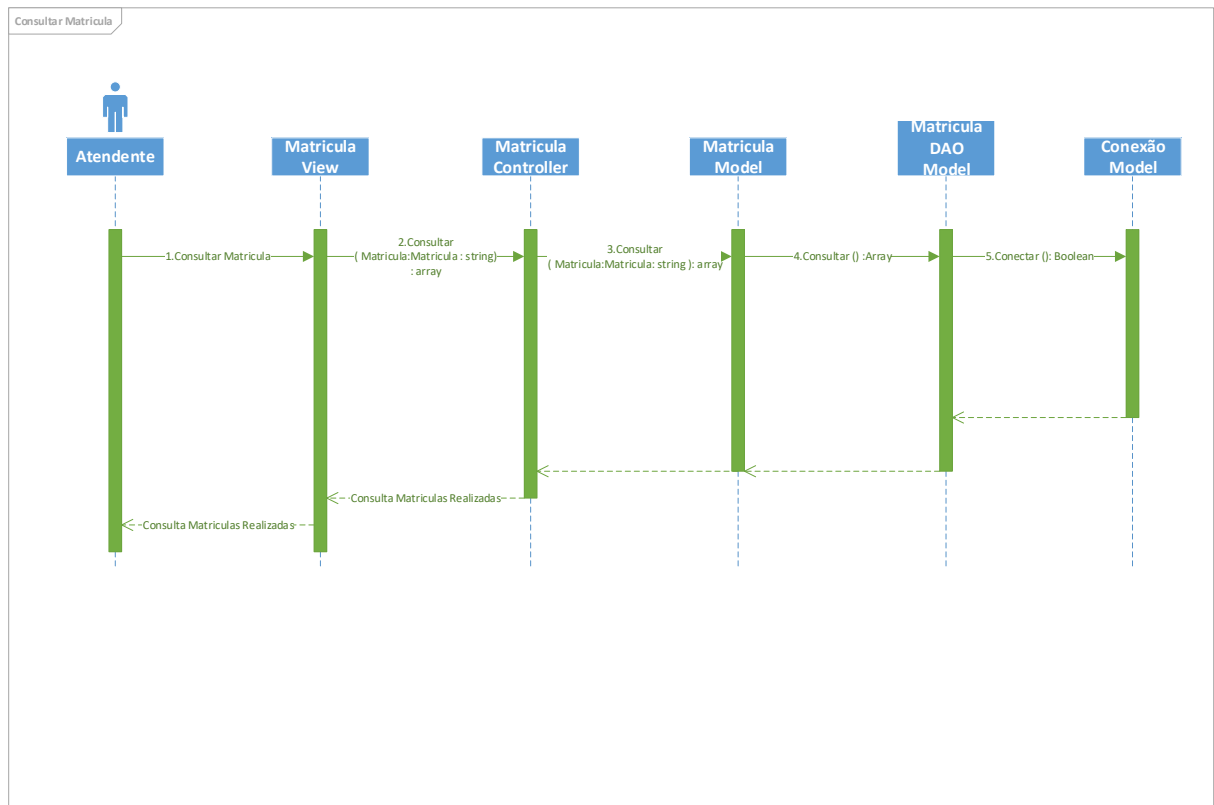


Figura 21 – Diagrama de sequência consultar matricula, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

Diagrama de Classe de Implementação - Gerar Relatório de Matricula

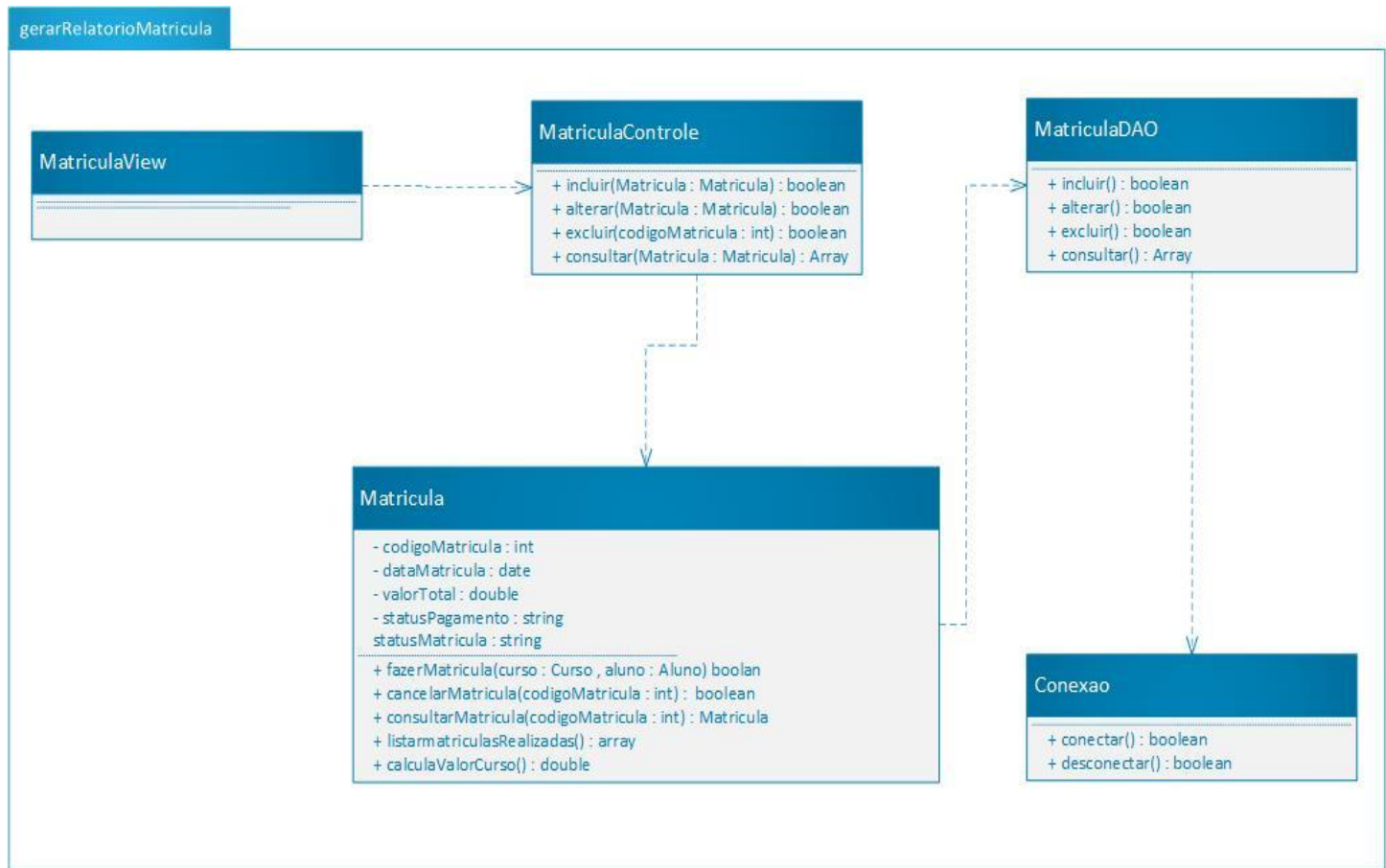


Figura 22 – Diagrama Gerar relatório de matricula, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

O diagrama de classe de implementação para o caso de uso “Gerar relatoria de matricula”, onde é descrito que o atendente emite relatórios em tela da quantidade de matrículas por curso em um determinado período. Podemos ver as classes, métodos e atributos no diagrama em questão.

Diagrama de sequência – Gerar Relatório de matrícula

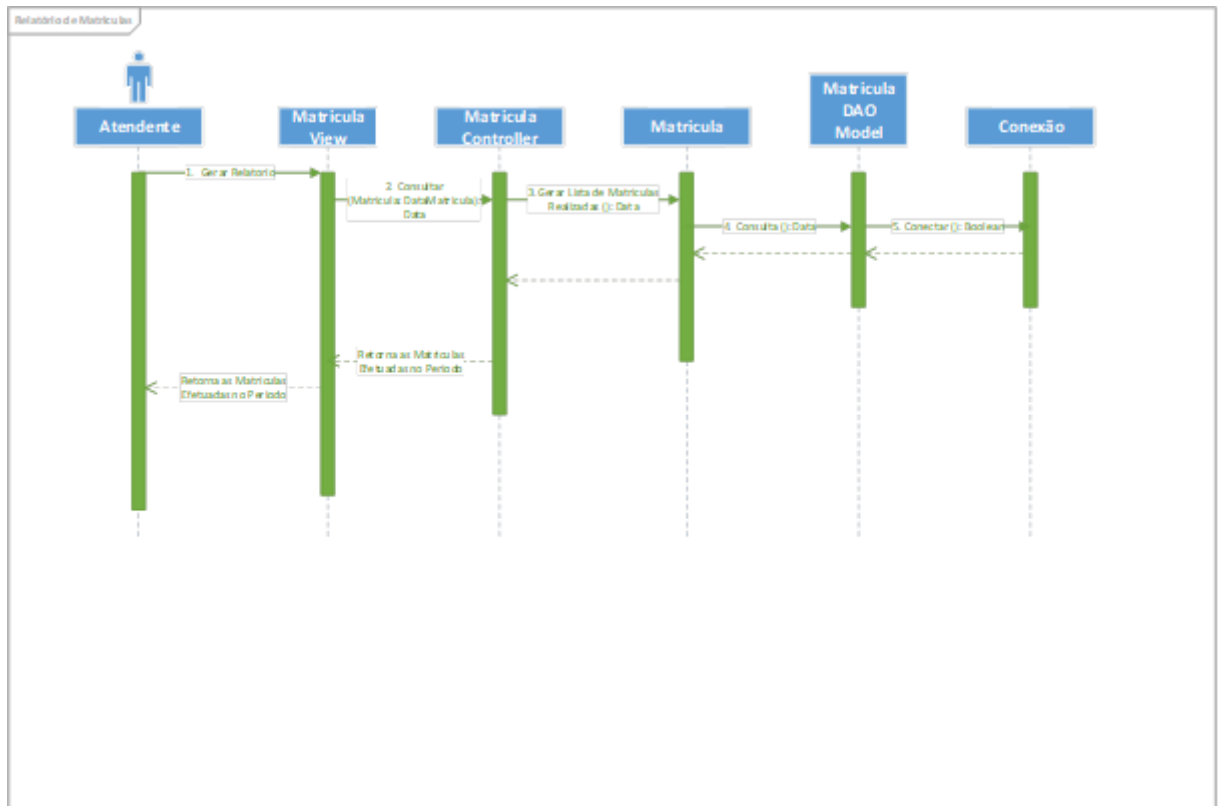


Figura 23 – Diagrama de sequência manter curso, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

Diagrama de Classe de implementação - Consultar Curso

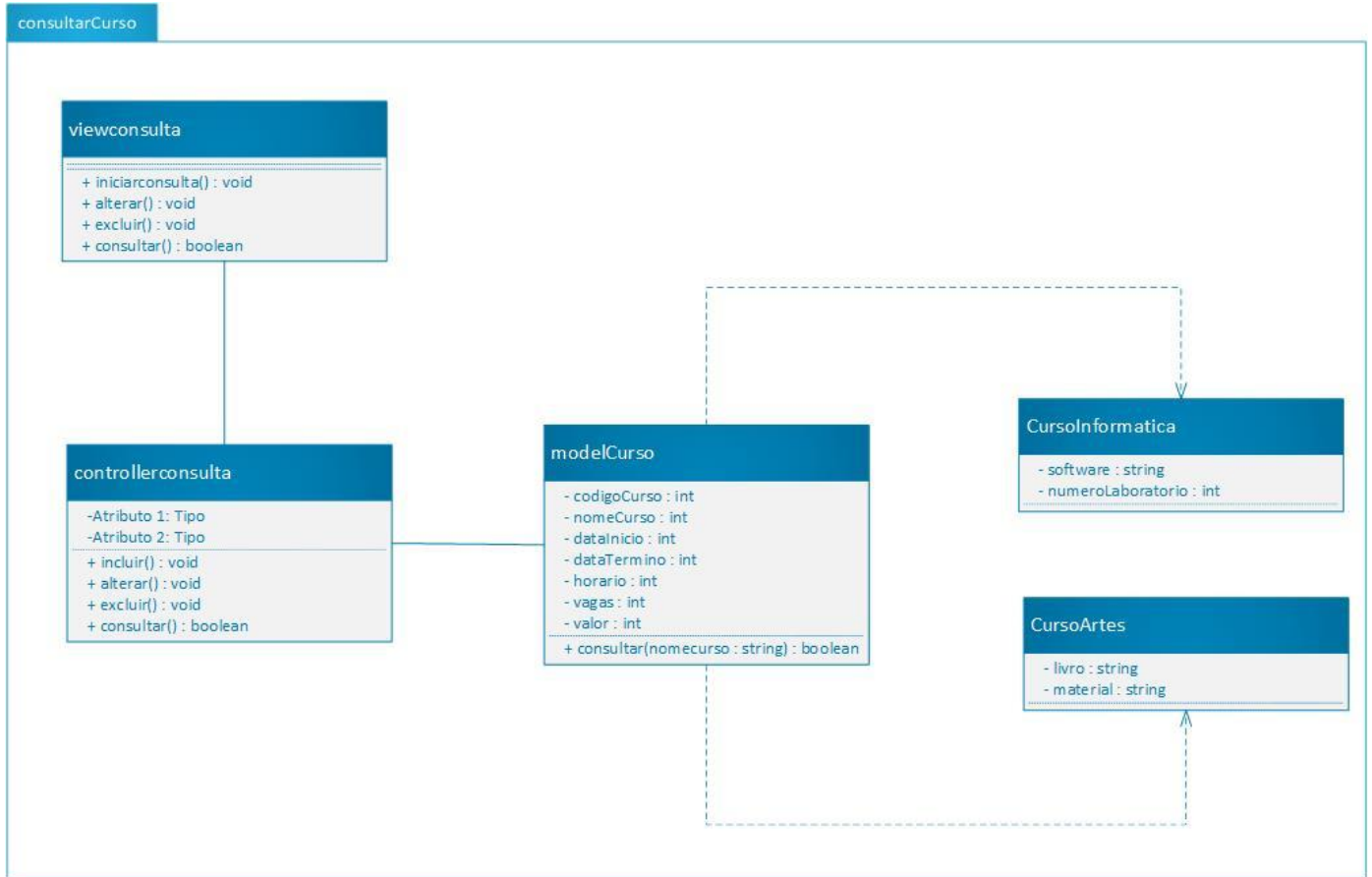


Figura 24 – Diagrama consultar curso, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

O diagrama de classe de implementação correspondente com a atividade apresentada pelo caso de uso consultar curso. Nele, é possível observar todas as classes e objetos envolvidos na estrutura de atuação apresentada pelo sistema, sobre essa atividade, de acordo com a arquitetura MVC.

Diagrama de Atividades

O diagrama de Atividade nos fornece uma visualização do comportamento ou aspecto dinâmico de um sistema descrevendo a sequência de ações em um processo. Sendo semelhante a fluxogramas, os diagramas de atividade mostram, além do fluxo principal, fluxos paralelos, simultâneos e alternativos. Os nós de atividades são utilizados para modelar o fluxo de controle e os dados entre as ações.

Os tópicos a seguir, segundo a gigante da tecnologia IBM (*International Business Machines*), descrevem os elementos de modelos de diagramas de atividade nos padrões da UML.

Atividades

As atividades são elementos de contêiner que descrevem o nível mais alto do comportamento em um diagrama de atividades. Contendo diversos nós de atividade e linhas de atividade que representam a sequência de tarefas em um fluxo de trabalho que resulta em um comportamento.

Ações

Na UML, uma ação representa uma unidade discreta de funcionalidade em uma atividade.

Nós de controle

Um nó de controle em um diagrama de atividade, é um nó de atividade abstrata que coordena o fluxo de controle em uma atividade.

Nós de objeto

Um nó de objeto é um nó de atividade abstrato que ajuda a definir o fluxo de objetos em uma atividade, ele indica que uma instancia de um classificador pode estar disponível em um ponto específico na atividade.

Linhas de atividade

Uma extremidade de atividade é uma conexão direcionada entre dois nós de atividade, quando uma ação é concluída, a linha de atividade segue seu fluxo para a próxima ação da sequência.

Os diagramas de atividade são também bastante utilizados para modelar processos de negócios, *workflows* além de sistemas, então, mesmo antes de se iniciar um projeto, ou antes ainda da fase de requisitos é possível que seja desenvolvido um diagrama de atividades para que possamos definir os fluxos de trabalho mais importantes, durante a fase de requisitos este diagrama é de suma importância para o entendimento do fluxo de eventos descritos nos casos de uso e já na fase de análise e *design*, utilizamos os diagramas de atividade para elucidar e definir o comportamento das operações.

Diagrama de atividade – calcular valor de curso

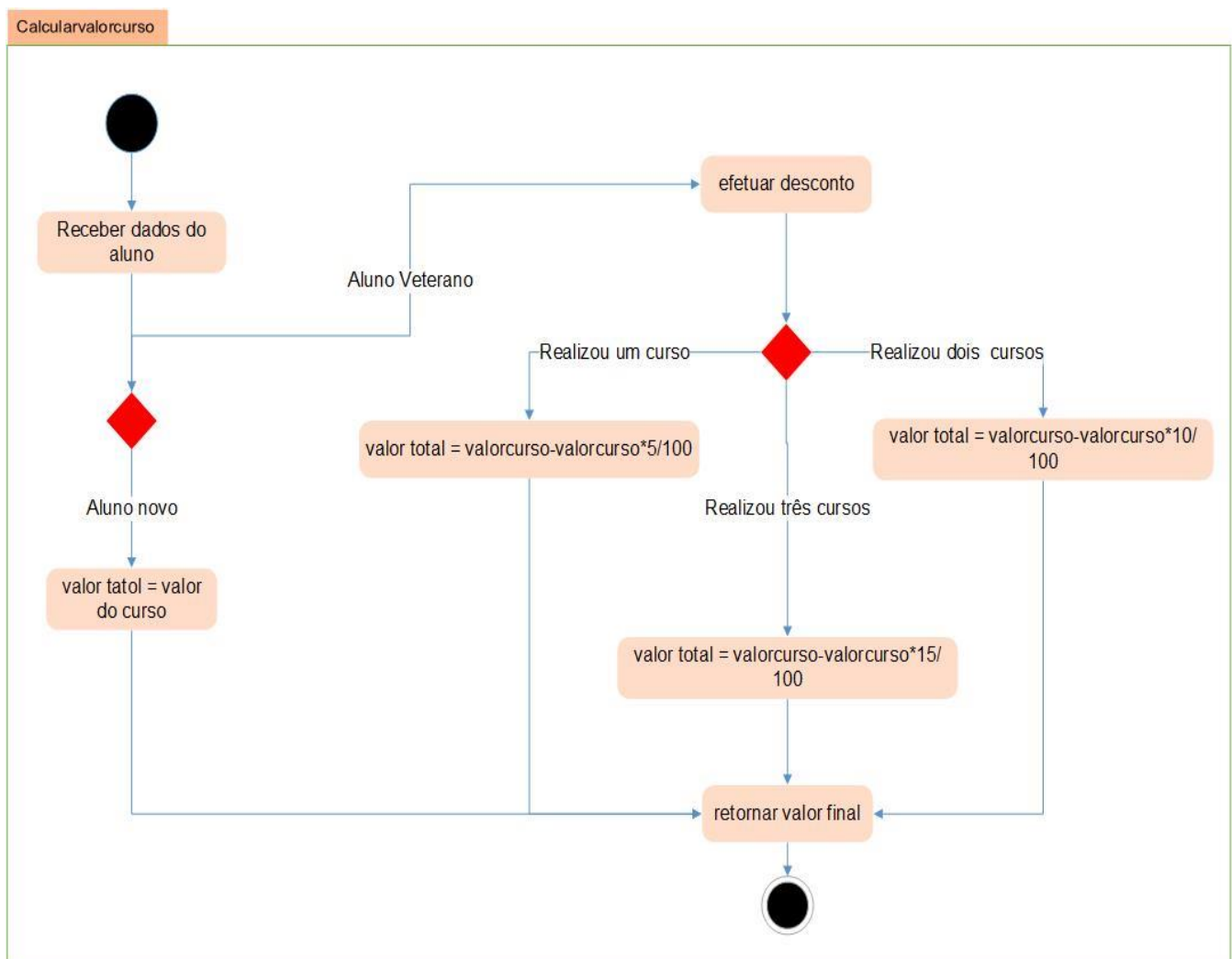


Figura 25 – Diagrama de atividades consultar curso, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

Diagrama de distribuição

Os componentes representam uma visão mais física do sistema de *software*, as partes deste sistema são apresentados sob uma perspectiva mais concreta, ou seja, podemos empacotar estes componentes e distribuí-los, mas, como representar, modelar e especificar questões importantes de configuração destes componentes?

O diagrama de distribuição tem por finalidade esclarecer a configuração física dos componentes de *hardware* do sistema em que serão executados os componentes de *software* (LARMAN, 2007). No diagrama de implantação estes componentes são nós e estes “nós” devem possuir um nome e um tipo e com a visão que nos é fornecida, poderemos perceber as dependências entre os nós e como se dá a comunicação entre eles.

É importante frisar que além de deixar clara a dependência entre os nós precisamos especificar o protocolo de comunicação entre eles, pois estes detalhes são fundamentais para dimensionarmos a infraestrutura necessária ao *software*.

Um nó de processamento é um recurso computacional do sistema que permite e execução de um *software*, ou de parte dele, sendo um componente como um computador, um dispositivo móvel, uma estrutura de memória ou até mesmo um dispositivo periférico, assim pensando em uma estrutura física, podemos imaginar que podemos distribuir esses componentes em diversas plataformas e que poderemos ter parte do sistema de *software* operando em um sistema operacional ou plataforma, enquanto outra parte poderá ser executado em outro.

Diagrama de distribuição ou implantação do sistema

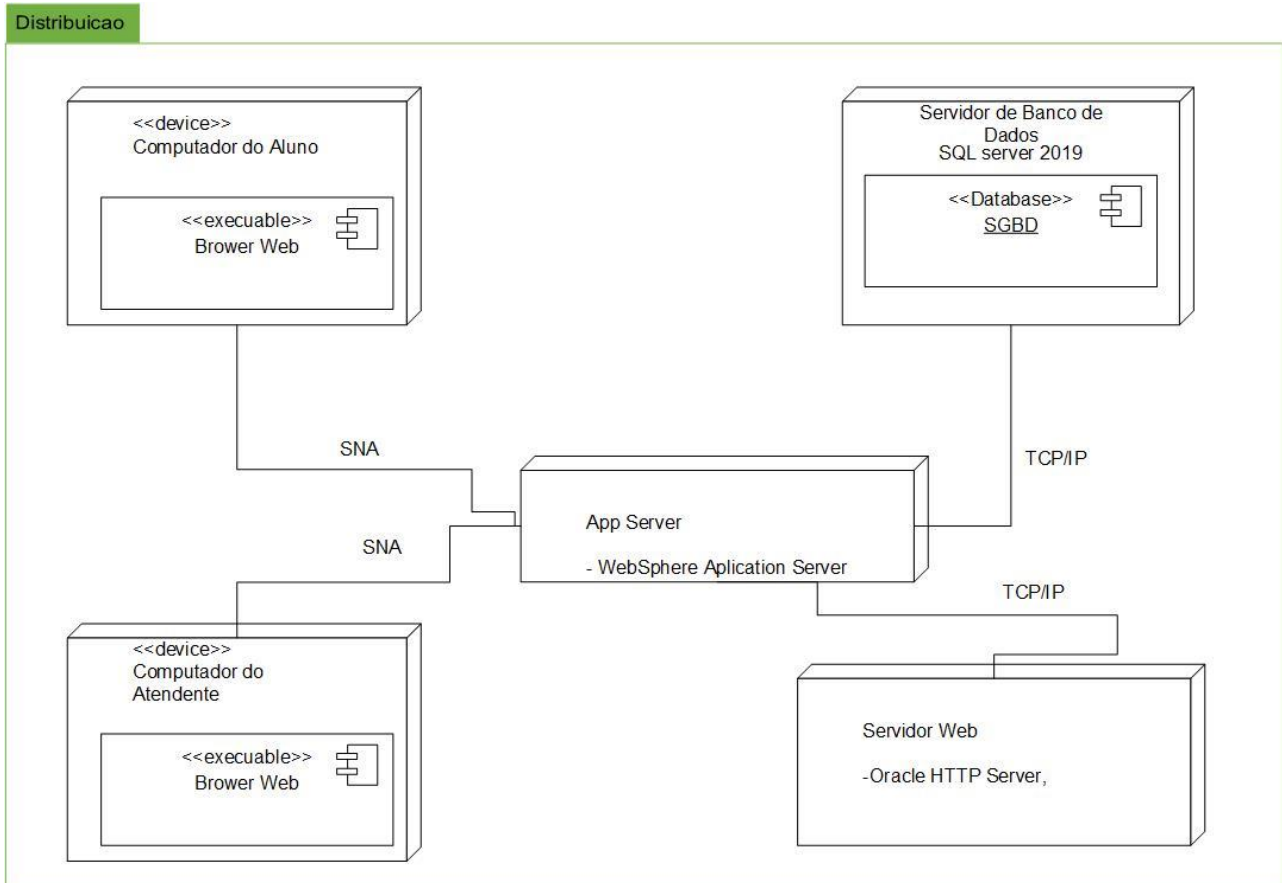


Figura 26 – Diagrama de distribuição, desenvolvido em Microsoft Visio / Fonte: O Autor, 2019

Conclusão

Concluimos que a fase de *Design* de um projeto de *software* tem seu foco na elucidação de todas as funcionalidades potenciais que o cliente e todos os *stakeholders* esperam do sistema.

Assim o padrão arquitetural juntamente com os diagramas de classe e sequências de ações, mostram a seriedade do projeto no nível arquitetural e estão diretamente relacionados com a organização do sistema e portanto, afetam os atributos de qualidade assim como desempenho, portabilidade, confiabilidade, disponibilidade. Sem um padrão de arquitetura todas as implantações a serem realizadas fossem apenas codificadas seriam falhas, trazendo caos para o projeto provocando retrabalho e aumento de custos. É percebida a relevância da fase de *design* e a importância de um arquiteto de *software* na realização de projetos de tecnologia, pois auxilia o programador sobre as principais decisões e técnicas necessárias para um projeto bem estruturado, o que por consequência, esse cuidado na estruturação garante que o sistema funcione de forma eficiente pois a divisão das responsabilidades em camadas torna o projeto mais sólido e de fácil entendimento, aumenta o reuso de código e facilita a manutenção do sistema. Desta forma podemos dizer que o projeto de *design* juntamente com o arquitetural é um fator determinante para o sucesso do sistema.

Referências bibliográficas

BEZERRA, E. Princípios de análise e projeto de sistemas com UML: Um guia prático para modelagem de sistemas orientados a objetos através da linguagem de modelagem unificada. Rio de Janeiro: Campus, 2006. P.99.

BOOCH, G.; Rumbaugh, J.; Jacobson, I.; *The Unified Modeling Language User Guide*. 2nd ed. Addison - Wesley

GUEDES, Gilleanes T. A.; 2011. UML 2 : Uma abordagem prática. 2nd ed. São Paulo : Novatec Editora.

LARMAN, C. Utilizando UML e padrões: Uma introdução à análise e ao projeto orientado a objetos e ao processo unificado. 2. Ed. Porto Alegre: Bookman, 2007.

PRESSMAN, R.S. Engenharia de *Software* 6ª. Edição, Editora *McGraw – Hill*, 2006, ISBN: 8586804576

Sites

Caso de uso Include, Extend e Generalização: Plinio Ventura – 28 Dez, 2014, *Updated* 10 Fev, 2019. Disponível em: <<https://www.ateomomento.com.br/caso-de-uso-include-extend-e-generalizacao/>> Acesso em: 23 de setembro de 2019.

Conceitos de diagramas de classe de implementação. Disponível em: <https://www.ibm.com/support/knowledgecenter/pt-br/SS4JE2_7.5.5/com.ibm.xtools.modeler.doc/topics/cdepd.html> Acesso em: 30 de setembro de 2019.

Desenvolvimento Orientado para Arquitetura. Disponível em: <<https://www.devmedia.com.br/arquitetura-de-software-desenvolvimento-orientado-para-arquitetura/8033>> Acesso em: 23 de setembro de 2019.

Diagramas de sequência de implementação IBM. Disponível em: <https://www.ibm.com/support/knowledgecenter/pt-br/SS4JE2_7.5.5/com.ibm.xtools.sequence.doc/topics/cseqd_v.html> Acesso em: 23 de setembro de 2019.

Diagramas de sequência de implementação IBM. Disponível em:

<https://www.ibm.com/support/knowledgecenter/pt-br/SS4JE2_7.5.5/com.ibm.xtools.sequence.doc/topics/cseqd_v.html> Acesso em: 30 de setembro de 2019.

Diagramas de sequência de implementação IBM. Disponível em:

<https://www.ibm.com/support/knowledgecenter/pt-br/SS4JE2_7.5.5/com.ibm.xtools.sequence.doc/topics/cseqd_v.html> Acesso em: 30 de setembro de 2019.

Suporte ao desenvolvimento de diagramas de atividade IBM. Disponível em:

<https://www.ibm.com/support/knowledgecenter/pt-br/SS8PJ7_9.6.0/com.ibm.xtools.modeler.doc/topics/cactd.html> Acesso em: 30 de setembro de 2019.