

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO RIO GRANDE
DO NORTE

ANNE LOUISE E. DA SILVA
MÁRCIA EDUARDA DE S. CRUZ

**QUERO ME FORMAR: UMA PROPOSTA DE SOFTWARE PARA ORGANIZAÇÃO DE
FORMATURAS**

LAJES, RN

2019

ANNE LOUISE E. DA SILVA
MÁRCIA EDUARDA DE S. CRUZ

**QUERO ME FORMAR: UMA PROPOSTA DE SOFTWARE PARA ORGANIZAÇÃO DE
FORMATURAS**

Trabalho de Conclusão de Curso apresentado ao Curso Técnico de Nível Médio em Informática do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, em cumprimento às exigências legais como requisito parcial à obtenção do título de Técnico de Nível Médio em Informática.

Orientador: Prof. Me. Dannylo Johnathan Bernardino Egídio

Coorientador: Prof. Me. Mário Melo

LAJES, RN

2019

ANNE LOUISE E. DA SILVA
MÁRCIA EDUARDA DE S. CRUZ

**QUERO ME FORMAR: UMA PROPOSTA DE SOFTWARE PARA ORGANIZAÇÃO DE
FORMATURAS**

Trabalho de Conclusão de Curso apresentado ao Curso Técnico de Nível Médio em Informática do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, em cumprimento às exigências legais como requisito parcial à obtenção do título de Técnico de Nível Médio em Informática.

Trabalho de Conclusão de Curso apresentado e aprovado em ___/___/___, pela seguinte Banca Examinadora:

BANCA EXAMINADORA

Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte

Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte

Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte

AGRADECIMENTOS

Agradecemos a Deus por ter nos dado saúde e força para superar as dificuldades encontradas ao longo deste projeto.

A esta instituição, seu corpo docente, direção e administração que oportunizaram a janela que hoje nos permite vislumbrar um horizonte superior.

Ao professor orientador Dannylo Johnathan, e, também, ao coorientador Mário Melo, pelo suporte no pouco tempo que lhes coube, pelas suas correções e incentivos.

Aos nossos pais, pelo amor, incentivo e apoio incondicional que nos proporcionaram ao longo dos quatro anos de curso.

E, por fim, a todos que direta ou indiretamente fizeram parte da nossa formação, o nosso muito obrigada.

RESUMO

Grande parte de turmas concluintes planejam festas de formatura que precisam de organização por parte dos alunos para o desenvolver, principalmente financeiro, destas festas. O uso de um software para auxiliar os estudantes no que diz respeito à administração de dados não só financeiros, mas também pessoais dos próprios alunos participantes de um determinado evento de formatura faz total diferença. Este trabalho tem como objetivo desenvolver uma ferramenta de apoio aos alunos de turmas concluintes de forma que os mesmos possam registrar informações necessárias para a realização dos eventos de formatura.

Palavras-chave: Alunos; Dado Financeiro; Formatura; Eventos.

ABSTRACT

Most graduating classes plan graduation parties that need organization on the part of the students to develop, especially financial, these parties. The use of software to assist students in managing not only financial but also personal data of the students themselves participating in a given graduation event makes all the difference. This paper aims to develop a support tool for students in graduating classes so that they can record information needed to hold graduation events.

Keywords: Students; Financial data; Graduation; Events.

LISTA DE SIGLAS

QMF – Quero Me Formar.

UML – Unified Modeling Language (Linguagem de Modelagem Unificada).

URL – Uniform Resource Locator (Localizador Padrão de Recursos).

RF – Requisito Funcional.

RNF – Requisito Não Funcional.

LISTA DE FIGURAS

Figura 1 – Modelo de diagrama de caso de uso	5
Figura 2 – Modelo de diagrama de classe.....	8
Figura 3 – Modelo de diagrama de atividade	10
Figura 4 – Diagrama de caso de uso do QMF.....	21
Figura 5 – Diagrama de classe do QMF	23
Figura 6 – Diagrama de atividade Gerenciar Financeiro do QMF	25
Figura 7 - Diagrama de atividade Gerenciar Formandos do QMF	26
Figura 8 - Protótipo da tela inicial do QMF	28
Figura 9 - Protótipo da tela de login do QMF.....	29
Figura 10 - Protótipo da tela de gerenciamento do QMF	30
Figura 11 - Protótipo da tela de gerenciamento financeiro do QMF	31

LISTA DE TABELAS

Tabela 1 - Levantamento de Requisitos Funcionais	12
Tabela 2 – Levantamento de requisitos não funcionais	13
Tabela 3 – Identificação e descrição dos atores	13
Tabela 4 – Detalhamento do caso de uso Realizar Login.....	14
Tabela 5 - Detalhamento do caso de uso Gerenciar Financeiro.....	15
Tabela 6 – Detalhamento do caso de uso Gerenciar Empresas.....	16
Tabela 7 – Detalhamento do caso de uso Gerenciar Turmas.....	17
Tabela 8 - Detalhamento do caso de uso Gerenciar Formandos	18
Tabela 9 - Detalhamento do caso de uso Gerenciar Eventos.....	19

SUMÁRIO

1. INTRODUÇÃO	1
1.1 OBJETIVO GERAL.....	2
1.2 OBJETIVOS ESPECÍFICOS	2
2. REFERENCIAL TEÓRICO.....	2
2.1 Engenharia de Requisitos	2
2.1.1 Requisitos funcionais e não funcionais.....	3
2.2 UML	4
2.2.1 Diagrama de casos de uso	5
2.2.2 Diagrama de classe.....	7
2.2.3 Diagrama de Atividade	9
2.3 SOFTWARES UTILIZADOS	11
2.3.1 <i>Astah Community</i>	11
2.3.2 <i>Wireframe.cc</i>	11
3. DESENVOLVIMENTO	12
3.1 ESPECIFICAÇÃO DE REQUISITOS.....	12
3.1.1 Requisitos do sistema	12
3.1.2 Identificação dos Atores.....	13
3.1.3 Descrição Detalhada dos Casos de Uso	14
3.2 DETALHAMENTO.....	21
3.2.1 Diagrama de Caso de Uso.....	21
3.2.2 Realizar <i>Login</i>	21
3.2.3 Gerenciar Turma	22
3.2.4 Gerenciar Empresas	22
3.2.5 Gerenciar Eventos.....	22
3.2.6 Gerenciar Formandos	22
3.2.7 Diagrama de Classe.....	23
3.2.8 Diagramas de Atividade	24
3.2.9 Gerenciar Financeiro.....	25
3.2.10 Gerenciar Formandos	26
4. PROTOTIPAÇÃO	28
4.1 Tela Inicial.....	28
4.2 Tela de <i>Login</i>	29

4.3 Tela de Gerenciamento.....	30
4.4 Tela de Gerenciamento Financeiro.....	31
4.5 Tela de Gerenciamento Financeiro - Ver Caixa	32
5. CONCLUSÃO.....	33
6. TRABALHOS FUTUROS	33
7. REFERÊNCIAS BIBLIOGRÁFICAS	34

1. INTRODUÇÃO

Atualmente, diversas turmas concluintes do ensino médio e do ensino superior executam eventos de formatura. Para isso, são contratadas empresas responsáveis pela organização e realização destes eventos. Geralmente, nas turmas são criadas as comissões, que são os alunos encarregados por coordenar e serem intermediários entre alunos e empresa (s).

Não existe uma ferramenta acessível que possa ser manuseada facilmente pelos alunos. Dependendo do pacote oferecido pela empresa de formatura que foi escolhida, a comissão poderá precisar de outros fornecedores que podem ser contratados à parte, o que provocaria uma possível confusão no que diz respeito à organização de dados das empresas, já que em boa parte das turmas isso é feito sem uma forma sistemática.

Problemas para administrar o dinheiro, para registrar informações da formatura ou anotações que poderão ser importantes posteriormente, são adversidades corriqueiras para a comissão. Visto que, tudo isso, geralmente, é registrado em estruturas pouco seguras, como em cadernos ou, ainda, em celulares, caracterizando uma anotação menos acessível para os membros da comissão, uma vez que tudo é anotado no bloco de notas de um participante da comissão. Além disso, não existe um acompanhamento no que diz respeito à investimento, que mostre o que resta em caixa e que exiba o valor que foi retirado ou adicionado do mesmo.

Ainda no âmbito financeiro, são vistas dificuldades com relação à atualização mensal de gastos. Também são identificados impasses quanto à programação da turma, como definição de datas para eventos. Neste caso, é imprescindível que a comissão tenha total controle das informações relevantes, sempre cadastrando os movimentos financeiros, além da programação da turma e formatura, de acordo com suas necessidades. Diante disso, é de suma importância o desenvolvimento de um sistema que atenda às necessidades das comissões, procurando amenizar os problemas citados acima.

Sabendo disso, visando tirar vantagem desse contexto e auxiliar os estudantes concluintes, é proposto neste trabalho a especificação do projeto “Quero Me Formar”, um sistema que tem o objetivo de possibilitar o cadastro de dados dos alunos

formandos, das empresas contratadas, cadastro dos pacotes oferecidos por estas empresas e o controle da movimentação financeira, além de cadastrar eventos da formatura já confirmados e até possíveis eventos, com descrição e datas. Através desse sistema, os alunos participantes da comissão, poderão ter acesso às informações mais importantes de forma fácil.

1.1 OBJETIVO GERAL

Este trabalho tem como objetivo o desenvolvimento de um projeto de *software*, denominado “Quero Me Formar”, voltado para comissões de turmas concluintes. De modo que o sistema auxilie e melhore a organização da turma no que diz respeito ao controle de dados necessários dos alunos participantes, das empresas contratadas e, enfim, do capital financeiro.

1.2 OBJETIVOS ESPECÍFICOS

- Mapear os desejos e as dificuldades enfrentadas pelas comissões das turmas concluintes;
- Realizar pesquisas bibliográficas ligadas ao tema de estudo deste trabalho e das tecnologias usadas para o desenvolvimento;
- Identificar as funcionalidades que o sistema deve ter, a partir da elicitação de requisitos;
- Construir os protótipos das telas de gerenciamento financeiro;
- Desenvolver o projeto do sistema QMF;

2. REFERENCIAL TEÓRICO

Este capítulo apresenta bases tecnológicas, referências bibliográficas e todo embasamento teórico do projeto, além disso, também demonstra os programas usados que serviram como ferramentas para elaborar o trabalho. Entre elas, a Engenharia de Requisitos, *UML (Unified Modeling Language)*, *Astah Community*.

2.1 Engenharia de Requisitos

Segundo Sommerville, (2011, p 63)

“Os requisitos de um sistema são as descrições do que o sistema deve fazer, os serviços que oferece e as restrições a seu funcionamento. Esses requisitos

refletem as necessidades dos clientes para um sistema que serve a uma finalidade determinada, como controlar um dispositivo, colocar um pedido ou encontrar informações. O processo de descobrir, analisar, documentar e verificar esses serviços e restrições é chamado engenharia de requisitos (RE, do inglês *requirements engineering*)”.

De acordo com Sommerville (2011), o termo “requisito” não é fortemente usado pela indústria de desenvolvedores de software, pois “em alguns casos, o requisito é apenas uma declaração abstrata em alto nível de um serviço que o sistema deve oferecer ou uma restrição a um sistema. No outro extremo, é uma definição detalhada e formal de uma função do sistema”.

Para Ian Sommerville (2011), alguns problemas que surgem no decorrer do processo de engenharia de requisitos são as falhas em não separar claramente esses diferentes termos de descrição, a distinção usada por ele é:

- Requisitos de usuários: declaração em uma linguagem natural com diagramas, de quais funções o sistema deverá executar e quais suas restrições.
- Requisitos de sistema: são as descrições mais detalhadas das funções, serviços e restrições operacionais do *software*. Neste documento deve conter tudo o que deverá ser implementado.

Existem dois grupos de requisitos: os requisitos funcionais e os requisitos não funcionais, que serão descritos com mais detalhes nas seções seguintes.

2.1.1 Requisitos funcionais e não funcionais

Os requisitos funcionais de um sistema descrevem o que ele deve fazer. Estes requisitos dependem do tipo de *software* que será desenvolvido, de quem serão seus possíveis usuários e da abordagem geral utilizada pela organização ao descrever os requisitos. Esses requisitos funcionais do usuário definem os recursos específicos a serem fornecidos pelo sistema. Eles foram retirados do documento de requisitos do usuário e mostram que os requisitos funcionais podem ser escritos em diferentes níveis de detalhamento. (SOMMERVILLE, 2011, p. 74)

Os requisitos não funcionais, diferentemente dos funcionais, são requisitos que não estão diretamente ligados a seus usuários. Os requisitos não funcionais geralmente determinam ou restringem as características do sistema como um todo. Exemplos

desses requisitos são: proteção, desempenho, disponibilidade. Requisitos não funcionais são, em geral, mais críticos que requisitos funcionais individuais. Na realidade, a distinção entre diferentes tipos de requisitos não é tão clara como sugerem essas definições [...] quando desenvolvido em mais detalhes, esse requisito pode gerar outros requisitos. (SOMMERVILLE, 2011, p. 73)

2.2 UML

Nos últimos anos, a *UML* tornou-se a linguagem padrão de modelagem escolhida pela indústria de desenvolvedores de softwares, por ser uma linguagem de modelagem com finalidade geral, que pode ser utilizada em todos os domínios de aplicação.

De acordo com Guedes (2005, pág. 13)

A *UML* (Unified Modeling Language) é uma linguagem visual utilizada para modelar sistemas computacionais por meio do paradigma de Orientação a Objetos. Essa linguagem se tornou, nos últimos anos, a linguagem-padrão de modelagem de software adotada internacionalmente pela indústria de Engenharia de Software.

No entanto, a *UML* não é uma linguagem de programação, mas sim de modelagem e tem grande aceitação pela comunidade de desenvolvedores de software por oferecer melhor visualização, especificação, construção e documentação dos artefatos de um sistema. A *UML* é uma linguagem muito expressiva que abrange todas as visões necessárias para o desenvolvimento e implantação do sistema. Essas características são definidas através da *UML* antes do *software* começar, de fato, a ser desenvolvido.

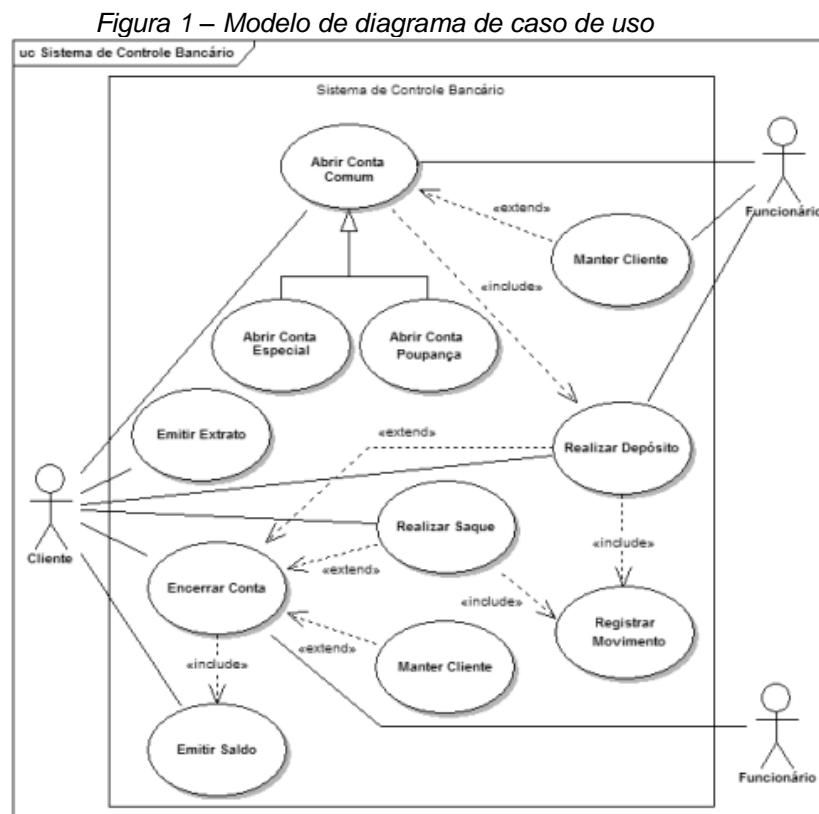
Tendo como objetivo propiciar múltiplas visões de um sistema a ser modelado, analisando-o e modelando-o sob diversos pontos, os diagramas analisam o sistema (ou parte dele), “[...] alguns diagramas enfocam o sistema de forma mais geral, apresentando a visão externa do sistema [...] enquanto outros oferecem uma visão de uma camada mais profunda do *software*” (GUEDES, 2011, p 30). Sabendo disso, os diagramas permitem que falhas sejam descobertas, minimizando a chance de erros futuros.

Nas seções seguintes, serão detalhados os diagramas utilizados no desenvolvimento do trabalho, a citar: diagrama de casos de uso, diagrama de classe,

diagramas de atividade e de sequência.

2.2.1 Diagrama de casos de uso

Guedes (2011) afirma que dentre os diagramas da *UML*, esse é considerado o mais geral e informal da *UML*. Este diagrama tem como objetivo demonstrar o comportamento externo do sistema, visando apresentar o software segundo a visão do usuário, ilustrando as funções e serviços oferecidos e quais usuários poderão utilizar cada funcionalidade. Ele apresenta uma linguagem simples e de fácil compreensão para que os usuários possam ter uma ideia geral de como o sistema irá de comportar. (GUEDES, 2011, p. 30). A seguir, na figura 1, um exemplo de diagrama *UML*:



Abaixo, segue uma breve explicação dos elementos utilizados no exemplo acima.

Atores - representam os papéis executados pelos vários usuários que poderão usar de alguma forma, as funcionalidades e serviços do sistema. Os atores são representados por símbolos de “bonecos magros”, contendo uma breve descrição logo abaixo de seu símbolo, que identifica o papel que o ator em questão assume dentro do

diagrama. (GUEDES, 2011, p. 53)

Casos de uso - são usados para representar os requisitos do sistema, ou seja, as funções, serviços ou tarefas que o sistema oferece. Eles são usados para expressar e documentar os comportamentos desejados para as funções. Segundo Guedes (2011) os casos de uso costumam ser documentados, informando como será seu funcionamento, quais atividades deverão ser realizadas, qual evento obrigará sua execução, quais atores poderão utilizá-lo etc. Os casos de uso são representados por elipses que contém dentro de si uma descrição, exibindo a que funcionalidade aquele caso de uso se refere.

Associações (ator/caso de uso) - é representada por uma linha ligando o ator ao caso de uso. As linhas podem conter setas, que indicam o sentido em que as informações trafegam além de indicar quem inicia a comunicação. Se houver necessidade, as associações podem ter descrição para esclarecer a informação que está sendo transmitida (GUEDES, 2011).

Generalização/Especialização - Segundo Guedes (2011), a especialização é uma forma de associação entre casos de uso, que possuem características semelhantes, indicando pequenas diferenças entre si. “Costuma-se definir um caso de uso geral que descreve as características compartilhadas por todos os casos de uso em questão e então relacioná-lo com todos os casos de uso envolvidos” (GUEDES, p. 58, 2011). A associação/generalização é representada por uma linha com uma seta mais grossa, indicando qual o caso de uso geral (para onde a seta aponta) e quais os casos de uso especializados (de onde saem as setas apontando para o caso de uso geral).

Inclusão - Os relacionamentos de inclusão indicam obrigatoriedade, ou seja, quando um caso de uso está ligado a outro por meio de inclusão, a execução do primeiro implica na execução do segundo, afirma Guedes (2011). Esta associação é representada por uma linha tracejada com uma seta em uma de suas extremidades que aponta para o caso de uso incluído. O relacionamento de inclusão também costuma conter um estereótipo com o texto “include” entre os sinais de menor (<) e maior (>).

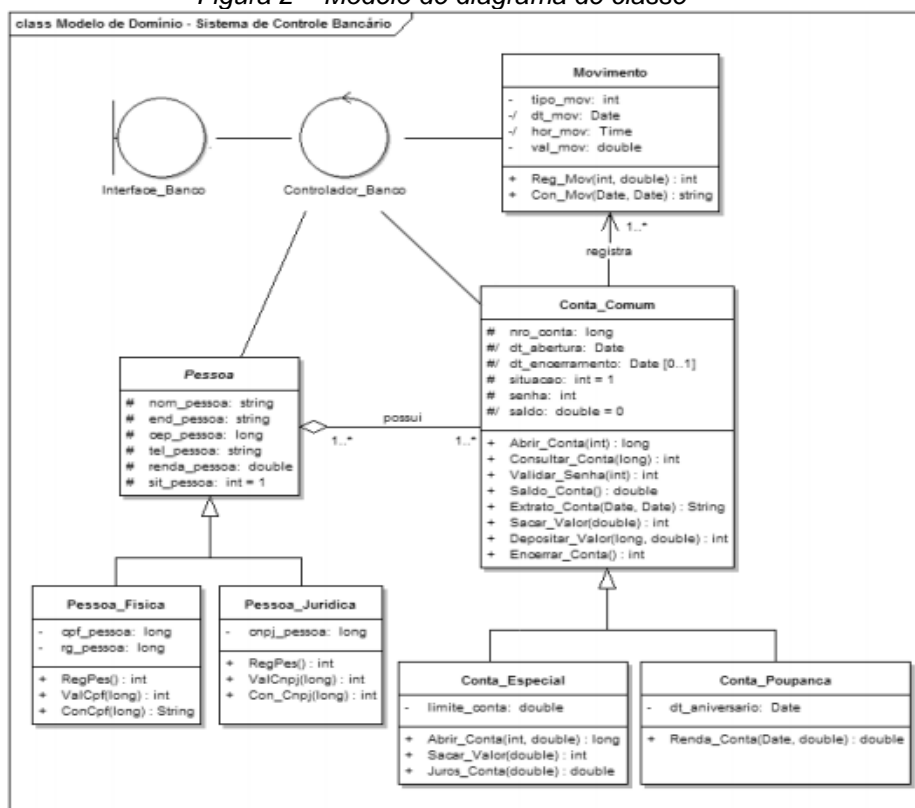
Extensão - São usados para descrever cenários opcionais de um caso de uso, ou seja, descrevem somente cenários que ocorrem em situações específicas, se uma determinada condição for satisfeita. Segundo Guedes (2011), relacionamentos de

extensão representam funções que não ocorrem sempre, no entanto, isso não significa que sejam incomuns. Como a inclusão, a representação da extensão é feita por uma linha tracejada, diferenciando apenas a direção para qual a seta aponta: ela deve apontar para o caso de uso que utiliza do caso de uso estendido, e ao invés do estereótipo conter o texto “*include*” contém “*extend*”.

2.2.2 Diagrama de classe

Sendo o diagrama de classe o mais utilizado na *UML*, seu principal objetivo é permitir a visualização das classes usadas pelo sistema e como elas se relacionam; sua função é definir uma estrutura lógica do sistema (GUEDES, 2011). Um diagrama de classe pode ser aplicado para definir o modelo de banco de dados, quando se assemelha aos antigos modelos entidade-relacionamento. Na verdade, segundo Guedes (2011), o diagrama de classes foi propositalmente projetado para ser uma evolução do modelo entidade-relacionamento. Vale ressaltar que o diagrama de classe não é usado exclusivamente para o projeto de modelos lógicos de bancos de dados. Abaixo, na figura 2, segue um exemplo de diagrama de classe a fim de facilitar a compreensão.

Figura 2 – Modelo de diagrama de classe



Fonte: UML 2 - Uma Abordagem Prática (2009)

Abaixo, segue uma breve explicação dos elementos utilizados no diagrama exposto como exemplo na figura 2.

Atributos e métodos - Geralmente, as classes possuem atributos, que armazenam os dados dos objetos de uma classe; além dos atributos, existem os métodos, que são as funções que uma instância de classe pode fazer. Os atributos e métodos são apresentados numa caixa contendo três divisões: a primeira contém a descrição ou nome da classe; a segunda guarda os atributos e seus respectivos tipos de dados; e a terceira, lista os métodos da classe. Segundo Guedes (2011), não é obrigatório uma classe possuir estas três divisões, pois, pode haver classe que não contenha atributos ou métodos. Os símbolos “#” e “+” na frente dos atributos e métodos representam sua visibilidade, que determinam quais objetos de quais classes podem usar o determinado atributo ou método.

Relacionamentos/Associações - Uma associação descreve uma conexão que ocorre entre os objetos de uma ou mais classes (GUEDES, 2011). As associações são representadas por linhas que ligam as classes envolvidas. É possível que essas linhas

tenham nomes ou títulos para facilitar o entendimento do tipo de relação que existe entre as classes envolvidas.

Agregação - De acordo com Guedes (2011), a agregação é um tipo especial de associação onde se tenta demonstrar que as informações de um objeto (objeto-todo) precisam ser complementadas por informações contidas em objetos de outra classe (objetos-parte). O símbolo de agregação é diferente do de associação por conter um losango na extremidade da classe que contém objetos-todo.

Generalização/Especialização - O objetivo dessa associação é representar a ocorrência de herança entre as classes, sendo possível identificar as classes-mãe e classes-filha, mostrando a hierarquia entre as classes.

Para evitar ter de declarar atributos e/ou métodos idênticos e como uma forma de reaproveitar código, cria-se uma classe geral em que são declarados os atributos e métodos comuns a todas as classes envolvidas no processo, e, então, declaram-se classes especializadas ligadas à classe geral. (GUEDES, 2011, P. 113).

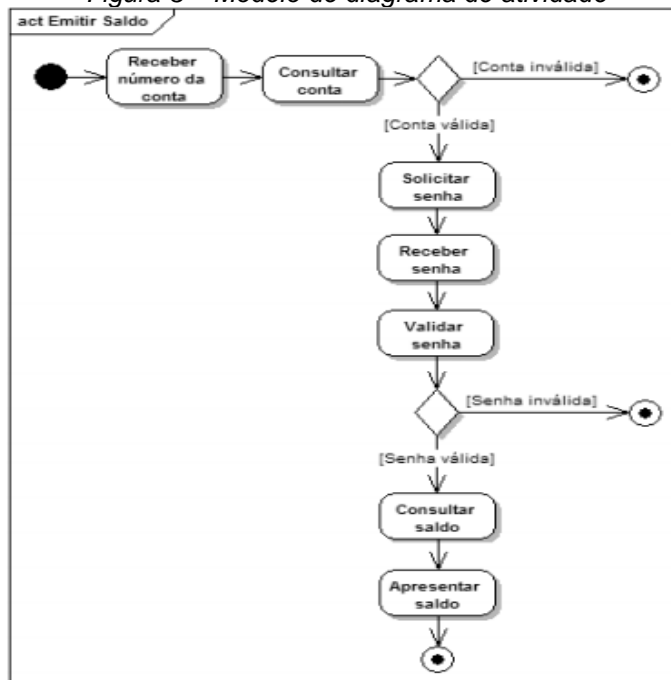
Sua representação é a mesma do diagrama de caso de uso.

Multiplicidade - Conforme Guedes (2011), a multiplicidade procura determinar o número máximo e mínimo de objetos envolvidos em cada extremidade da associação, além de possibilitar especificar o nível de dependência de um objeto para com os outros envolvidos na associação.

2.2.3 Diagrama de Atividade

“O diagrama de atividade é o diagrama com maior ênfase ao nível de algoritmo da UML, e provavelmente um dos mais detalhistas” (GUEDES, 2011, 277). O diagrama de atividade é muito similar aos antigos fluxogramas usados para desenvolver a lógica de programação e definir o fluxo de controle de um algoritmo. Esse diagrama é utilizado para modelar atividades que podem ser um método, um algoritmo ou um processo completo. A seguir, na figura 3 apresentamos um modelo de diagrama de atividade.

Figura 3 – Modelo de diagrama de atividade



Fonte: UML 2 - Uma Abordagem Prática (2009)

Atividade - Guedes (2011, p.278) afirma que “uma atividade específica a coordenação de execuções de comportamentos subordinados usando um modelo de fluxo de controle e dados” Uma atividade é simbolizada por um retângulo grande com as bordas arredondadas.

Nó de ação - Segundo Guedes, estes são os elementos mais simples de uma atividade (2011). Estes representam um passo que deve ser executado em uma atividade. Um nó de ação é representado por um pequeno retângulo com as bordas arredondadas, como uma atividade, porém o símbolo do nó de ação é menor.

Fluxo de controle - É um conector que liga dois nós, enviando sinais de controle. Pode possuir uma descrição, uma condição de guarda ou uma restrição, conhecida neste diagrama como peso. O fluxo de controle é representado por uma seta apontando para o novo nó e partindo do antigo. (GUEDES, 2011)

Nó inicial - Guedes (2011, p. 280) afirma que “esse componente pertence ao grupo de nós de controle utilizados para o controle de fluxo da atividade” Um nó inicial é representado por um círculo preenchido e é usado para representar o início do fluxo quando uma atividade é chamada.

Nó de final de atividade - Esse componente também é um nó de controle usado

para representar o fim de um fluxo de uma atividade. Um nó de final de atividade é representado por um círculo preenchido dentro de um círculo vazio.

Nó de decisão - Segundo Guedes (2011), um nó de decisão é usado para simbolizar uma escolha entre dois ou mais fluxos realizáveis, em um dos fluxos será escolhido em detrimento dos demais. Nós de decisão são representados entre colchetes, que determinam a condição para que um fluxo possa ser escolhido.

2.3 SOFTWARES UTILIZADOS

Este tópico apresenta os principais softwares utilizados para a elaboração do projeto.

2.3.1 Astah Community

O Astah Community é a versão gratuita da ferramenta *Astah*, ele é utilizado para modelagem de diagramas UML (*Unified Modeling Language* – Linguagem de Modelagem Unificada). O *Astah Community* possui uma interface de fácil utilização, sendo ele dividido em várias seções, cada uma com sua respectiva finalidade (LIMA, 2019).

Essa ferramenta é popular por sua simplicidade ao criar diagramas, como: diagramas de casos de uso, de sequência, de atividades, de classes, máquina de estado, entre outros. Esse sistema foi desenvolvido no Japão na plataforma Java, o que garante sua portabilidade em qualquer máquina que possua JVM (Máquina Virtual Java).

2.3.2 Wireframe.cc

Wireframe.cc é uma aplicação *web* que permite a criação de protótipos estáticos (rascunhos usados para exemplificar os relacionamentos entre páginas *web*) de forma simples e prática. O *Wireframe.cc* é gratuito e pode ser usado para a criação de protótipos de diferentes plataformas, como tablets e smartphones. Para criar um projeto basta acessar o site e iniciar a sua construção, após o término é necessário salvar para se obter uma *URL*, dessa forma, podemos compartilhar o *Wireframe* e modificá-lo posteriormente. (WIREFRAME.CC, 2019)

3. DESENVOLVIMENTO

Este capítulo irá tratar da seção do detalhamento do projeto, além de mostrar como deu-se a construção das tabelas e diagramas do sistema.

3.1 ESPECIFICAÇÃO DE REQUISITOS

Esse tópico essencialmente irá tratar de fase de levantamento de requisitos e funcionalidades em relação à documentação do sistema que previamente foram formuladas.

3.1.1 Requisitos do sistema

Tabela 1 - Levantamento de Requisitos Funcionais

Cod.	Nome	Descrição	Prioridade
RF01	Autenticar Usuários.	Possibilita que os usuários registrados no sistema tenham acesso à área reservada do mesmo. O usuário deverá informar seu e-mail e senha do registro para poder iniciar sessão.	Essencial.
RF02	Manter Financeiro.	Possibilita controlar gastos financeiros. O usuário deve informar a quantia em dinheiro que foi retirada ou adicionada do caixa da turma.	Importante.
RF03	Gerenciar Eventos.	Possibilita cadastrar, remover ou alterar os eventos que serão feitos para arrecadação de dinheiro.	Importante.
RF04	Gerenciar Formandos.	Possibilita cadastrar alunos que participaram de pelo menos um evento de formatura e mostrar valor final a ser pago, além da situação do aluno: se há parcelas em atraso, se está em dias.	Essencial.

RF05	Gerenciar Turmas.	Possibilita aos usuários realizar cadastro no sistema informando nome da turma, e-mail e uma senha para registro.	Essencial.
RF06	Gerenciar Empresas.	Possibilita cadastrar, remover e alterar as empresas contratadas além fornecer informações de cada uma, mostrando também o que cada empresa irá fazer, e por quais eventos está responsável.	Desejável.

Fonte: Elaborado pelo autor

Tabela 2 – Levantamento de requisitos não funcionais

Cod.	Nome	Descrição	Prioridade
RNF01	Usabilidade.	O sistema deve ser fácil de manusear, tornando simples e agradável realizar determinadas tarefas.	Essencial.
RNF02	Manutenibilidade.	Deve fazer reparações e/ou alterações no software após sua disponibilização para uso.	Importante.
RNF03	Segurança.	Acessos não autorizados ao sistema e dados associados não serão permitidos.	Essencial.

Fonte: Elaborado pelo autor

3.1.2 Identificação dos Atores

Tabela 3 – Identificação e descrição dos atores

Ator	Descrição
01 – Usuário.	Membro(s) que compõem a comissão de formatura.

Fonte: Elaborado pelo autor

3.1.3 Descrição Detalhada dos Casos de Uso

Nesta seção são apresentados os detalhes dos principais casos de uso.

Tabela 4 – Detalhamento do caso de uso Realizar Login

Documentação do caso de uso Realizar <i>Login</i>	
Nome do caso de uso	Realizar <i>login</i>
Atores:	Usuário, sistema.
Descrição	Possibilita aos usuários realizar <i>login</i> para acesso a área reservada do sistema. O usuário deverá informar seu e-mail e senha de registro para realizar seu <i>login</i> .
Pré-condições:	O usuário já deve ter cadastro no sistema.
Pós-condições:	<i>Login</i> realizado.
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1 – Usuário insere e-mail e senha.	2 – Sistema disponibiliza a tela inicial com menu e suas opções (que são os demais casos de uso).
5 – Fim	
Fluxo de Exceção	
[1] Caso o usuário insira e-mail ou senha incorreto ou que não exista, o sistema exibe a mensagem M1.	
Mensagens e exceções	
M1: E-mail ou senha incorreta!	

Fonte: Elaborado pelo autor

Tabela 5 - Detalhamento do caso de uso Gerenciar Financeiro

Documentação do caso de uso Gerenciar Financeiro	
Caso de Uso:	Gerenciar Financeiro
Atores:	Usuário, sistema.
Descrição	Possibilita controlar gastos financeiros. O usuário deve informar a quantia em dinheiro que foi retirada ou adicionada do caixa da turma.
Pré-condições:	O usuário deve estar logado no sistema.
Pós-condições:	Não existe.
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1 – Usuário seleciona a opção “Gerenciar Financeiro” do menu.	2 – Sistema disponibiliza a tela com os botões: Adicionar; para o usuário cadastrar um dado (financeiro). Remover; para o usuário remover algum dado cadastrado. Ver caixa; para o usuário consultar valor total disponível em caixa.
3 – Usuário informa uma das opções disponíveis e realiza a ação.	
4 – Fim.	
Fluxo de Exceção	
[3] Caso o usuário selecione cancelar, a tela aberta no momento deverá ser fechada e o caso de uso termina.	[2] Caso o caixa esteja zerado, exibir mensagem M1.
Mensagens e exceções	

M1: Caixa vazio!

Fonte: Elaborado pelo autor

Tabela 6 – Detalhamento do caso de uso Gerenciar Empresas

Documentação do caso de uso Gerenciar Empresas	
Caso de Uso:	Gerenciar Empresas
Atores:	Usuário, sistema.
Descrição	Possibilita adicionar, remover ou alterar empresas e dados das empresas cadastradas.
Pré-condições:	Estar logado no sistema.
Pós-condições:	Não existe.
Sequência Típica de Eventos	
Ação do Ator	Resposta do Sistema
1 – O usuário seleciona a opção “Gerenciar Empresas” do menu.	2 – A tela deve exibir uma tabela com as empresas já cadastradas, se for o caso. Além dos botões: Adicionar; para o usuário cadastrar alguma nova empresa. Remover; para o usuário remover alguma empresa já cadastrada. Alterar; para o usuário alterar informações da empresa já cadastrada.
3 – O usuário deve selecionar alguma opção disponibilizada pelo sistema, se assim desejar.	4 – Se for o caso, o sistema deve mostrar a tela do caso de uso que foi selecionado pelo usuário.
5 - Fim.	
Eventos Alternativos	

Não existe.
Mensagens e exceções
Não existe.

Fonte: Elaborado pelo autor

Tabela 7 – Detalhamento do caso de uso Gerenciar Turmas

Documentação do caso de uso Gerenciar Turmas	
Caso de Uso:	Gerenciar Turmas
Atores:	Usuário, sistema.
Descrição	Possibilita cadastrar uma nova turma.
Pré-condições:	Não existe.
Pós-condições:	Turma cadastrada com sucesso.
Sequência Típica de Eventos	
Ação do Ator	Resposta do Sistema
1 – O usuário deve selecionar a opção “Cadastrar Turma” na tela de <i>login</i> do sistema.	2 – O sistema deve disponibilizar na tela os campos de texto: Nome da turma; para o usuário informar o nome da turma. E-mail para cadastro; para o usuário informar um e-mail para que possa realizar o <i>login</i> ; Senha para cadastro; para o usuário informar uma senha para <i>login</i> .
3 – O usuário deve preencher os campos e confirmar.	4 – O sistema verifica e valida as informações inseridas.

5 – O sistema exibe a mensagem M1.	
6 – O sistema carrega a tela principal.	
7 – Fim.	
Eventos Alternativos	
[3] Caso o usuário selecione cancelar, o sistema deve voltar à tela de <i>login</i> .	[4] Caso o sistema encontre algum cadastro com o mesmo nome de turma ou mesmo e-mail, exibir mensagem M2.
Mensagens e exceções	
M1: Cadastro realizado com sucesso! M2: Nome ou e-mail não disponível!	

Fonte: Elaborado pelo autor

Tabela 8 - Detalhamento do caso de uso Gerenciar Formandos

Documentação do caso de uso Gerenciar Formandos	
Caso de Uso:	Gerenciar Formandos
Atores:	Usuário, sistema.
Descrição	Possibilita adicionar, remover ou alterar cadastros e dados dos alunos.
Pré-condições:	Estar logado no sistema.
Pós-condições:	Não existe.
Sequência Típica de Eventos	
Ação do Ator	Resposta do Sistema

1 – O usuário seleciona a opção “Gerenciar Formandos” do menu.	2 – A tela deve exibir uma tabela com os formandos já cadastrados, se for o caso. Além dos botões: Adicionar; para o usuário cadastrar um novo formando. Remover; para o usuário remover algum formando já cadastrado. Alterar; para o usuário alterar informações de um formando já cadastrado.
3 – O usuário deve selecionar alguma opção disponibilizada pelo sistema, se assim desejar.	4 – Se for o caso, o sistema deve mostrar a tela do caso de uso que foi selecionado pelo usuário.
5 – Fim.	
Eventos Alternativos	
[4] Caso o usuário selecione cancelar, a tela aberta no momento deverá ser fechada e o caso de uso termina.	
Mensagens e exceções	
Não existe.	

Fonte: Elaborado pelo autor

Tabela 9 - Detalhamento do caso de uso Gerenciar Eventos

Documentação do caso de uso Gerenciar Eventos	
Caso de Uso:	Gerenciar Eventos
Atores:	Usuário, sistema.
Descrição	Possibilita adicionar, remover ou alterar cadastros e dados dos eventos realizados para arrecadação de dinheiro.
Pré-condições:	Estar logado no sistema.

Pós-condições:	Não existe.
Sequência Típica de Eventos	
Ação do Ator	Resposta do Sistema
1 – O usuário seleciona a opção “Gerenciar Eventos” do menu.	2 – A tela deve exibir uma tabela com os eventos já cadastrados, se for o caso. Além dos botões: Adicionar; para o usuário cadastrar um novo evento. Remover; para o usuário remover algum evento já cadastrado. Alterar; para o usuário alterar informações de algum evento.
3 – O usuário deve selecionar alguma opção disponibilizada pelo sistema, se assim desejar.	4 – Se for o caso, o sistema deve mostrar a tela do caso de uso que foi selecionado pelo usuário.
5 – Fim.	
Eventos Alternativos	
[4] Caso o usuário selecione cancelar, a tela aberta no momento deverá ser fechada e o caso de uso termina.	
Mensagens e exceções	
Não existe.	

Fonte: Elaborado pelo autor

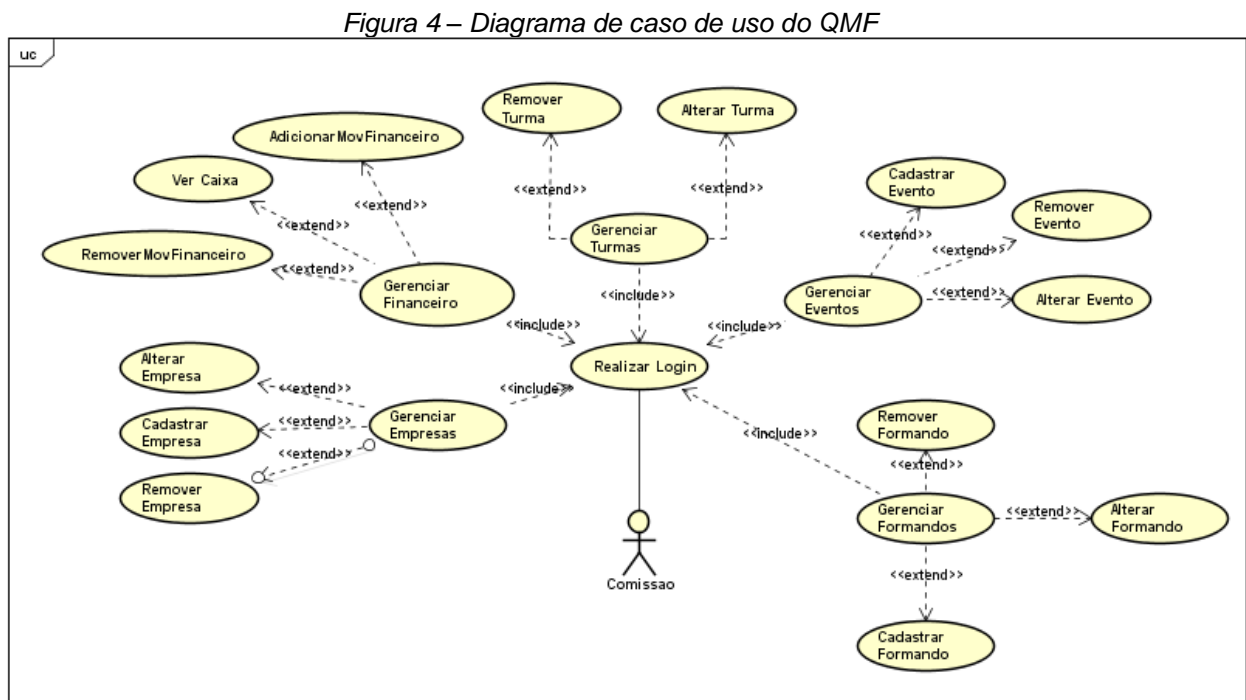
3.2 DETALHAMENTO

O detalhamento do módulo está subdividido em diagrama de casos de uso, diagrama de classe e diagramas de atividade. A partir desses diagramas padrões da UML, é possível explicar de forma clara como o sistema deverá se comportar.

3.2.1 Diagrama de Caso de Uso

Os diagramas de caso de uso têm um papel de bastante relevância para a modelagem do comportamento de um sistema, subsistema, ou ainda, de uma classe. Além disso, a técnica de casos de uso é uma ferramenta interessante para abstrair os requisitos funcionais e atores dos sistemas, mostrando quem são e como interagem.

Sendo assim, levando em consideração os atores e os requisitos funcionais deste trabalho, foram definidos dezoito casos de uso, como mostrado na figura 4.



Fonte: Elaborado pelo autor

3.2.2 Realizar *Login*

Para realizar qualquer ação do sistema, o usuário precisa ter um cadastro da turma, correspondente ao caso de uso Gerenciar Turmas, e estar logado no sistema. Todas as demais funções só poderão ser realizadas quando o usuário fizer *login*.

3.2.3 Gerenciar Turma

Este caso de uso é equivalente ao cadastro da turma e será a partir da ação deste caso de uso que será possível realizar as demais funções. Para cadastrar uma turma, basta fornecer os dados que o sistema solicitará.

3.2.4 Gerenciar Empresas

O caso de uso gerenciar empresa estende mais 3 (três) casos de uso opcionais (assim como os casos de uso que serão citados abaixo) que são eles: Adicionar Empresa; Remover Empresa e Alterar Empresa, nos quais o usuário poderá cadastrar alguma empresa contratada, adicionando as informações necessárias. Remover empresa, como o próprio nome diz, exclui o cadastro de uma empresa, e, por fim, alterar deve permitir que o usuário altere dados de uma empresa cadastrada.

3.2.5 Gerenciar Eventos

Assim como o caso de uso “Gerenciar Empresas”, este também contém 3 (três) casos de usos opcionais: Adicionar Evento; Remover Evento e Alterar Evento. Deverá ser permitido que o usuário possa realizar o cadastro de um evento futuro; remova um evento já cadastrado no sistema por um usuário e, ainda, possa alterar alguma informação contida no cadastro de um determinado evento.

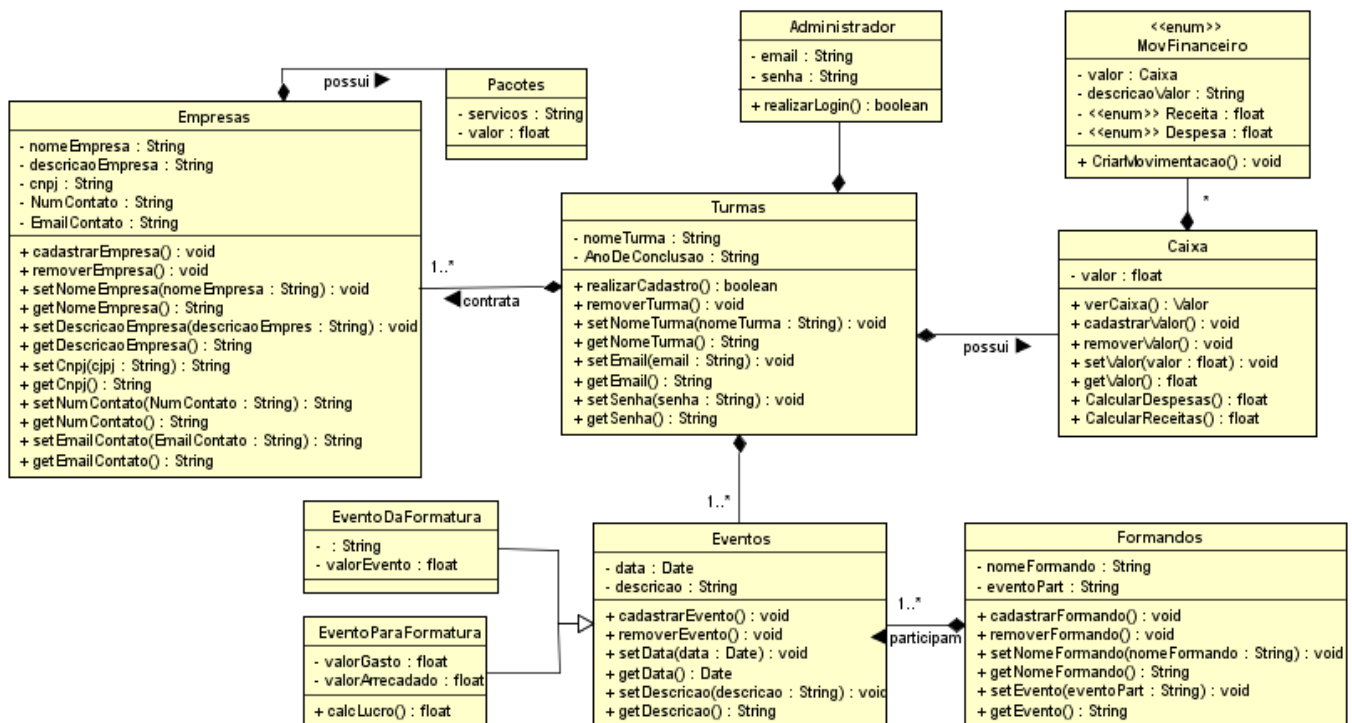
3.2.6 Gerenciar Formandos

Semelhante aos casos de uso descritos acima, o caso de uso “Gerenciar Formandos” possui também os seguintes casos de uso opcionais: Cadastrar, Remover e Alterar Formandos. O usuário deverá cadastrar alunos (formandos) que participarão de pelo menos um evento de formatura; deverá ser permitido que ele possa remover este cadastro, caso queira, e, por fim, deverá poder alterar alguma informação de um cadastro de formando.

3.2.7 Diagrama de Classe

O diagrama de classe tem como objetivo descrever como as classes se relacionam entre si, além de determinar os atributos e métodos contidos nas mesmas. Desta forma, foram definidas dez (10) classes para definir a estrutura do sistema QMF, sendo expostas na figura 5:

Figura 5 – Diagrama de classe do QMF



Fonte: Elaborado pelo autor

Inicialmente, a classe **Empresas** está sendo relacionada à outras duas classes: **Pacotes** e **Turmas**, nas quais pode-se perceber que é possível uma empresa possuir um pacote com todos os serviços, e, respectivamente, seus valores, que serão oferecidos. Além disso, uma turma poderá contratar uma empresa, que será responsável, de fato, pelo desenvolvimento de eventos da formatura.

Por conseguinte, a classe **Turmas** se relacionará com as demais classes: a classe **Administrador** tem como objetivo validar o *login* do usuário, ou seja, se as informações inseridas pelo usuário forem verdadeiras, ele tem acesso à área restrita do sistema, caso sejam falsas, uma mensagem de erro aparecerá, por isso o método **realizarLogin()** é do tipo *boolean*, que retorna verdadeiro ou falso. A classe Turma

possui, também, relacionamento com a classe **Caixa** (toda turma deve possuir um caixa), que é responsável por cadastrar dados financeiros de uma determinada turma e exibir as movimentações feitas, por isso a classe **MovFinanceiro**, que exibirá as movimentações feitas por uma determinada turma. Ademais, a classe turma terá relacionamento com a classe **Eventos**, responsável pelo cadastro de eventos da turma em questão. Estes eventos podem ser divididos em **EventosDaFormatura** (que são eventos realizados no final do curso) ou **EventosParaFormatura** (que são eventos feitos ao longo do curso para arrecadação de dinheiro), como estão nos próprios nomes das classes. Por fim, a classe **Formandos** se relacionará com a classe **Eventos**, permitindo que um formando, que foi cadastrado naquela turma específica, possa participar dos eventos de formatura da mesma.

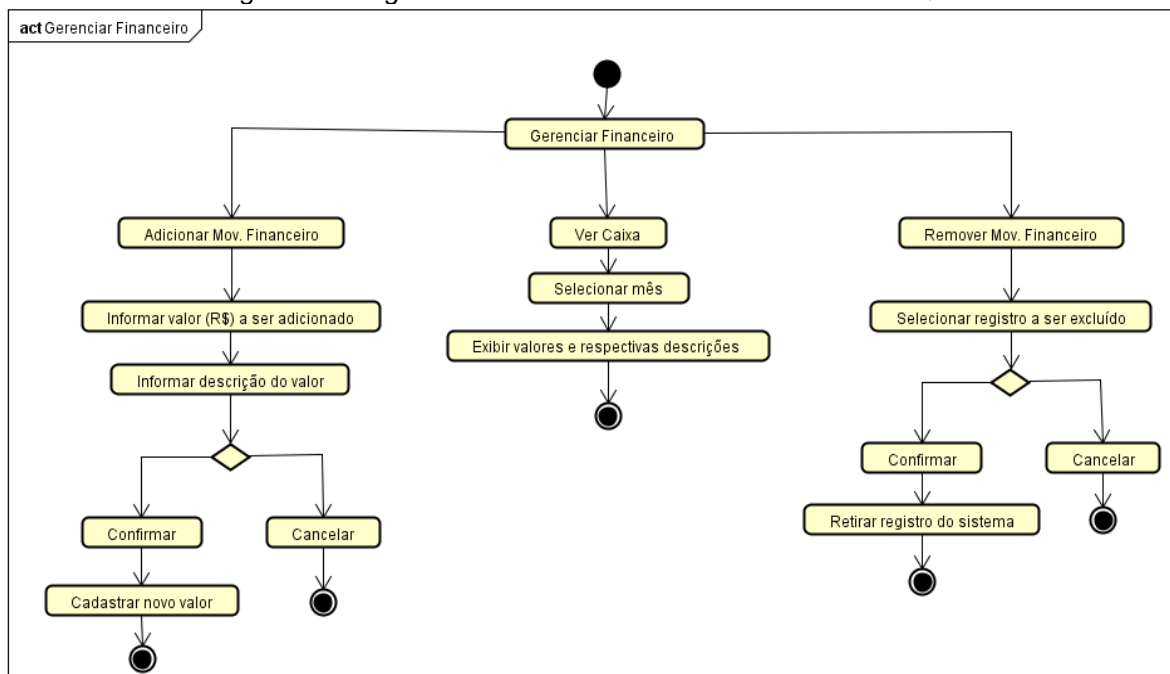
3.2.8 Diagramas de Atividade

Nesta seção, estarão dispostos os diagramas de atividade, responsáveis pelo fluxo que ocorre de uma atividade para outra.

3.2.9 Gerenciar Financeiro

Na figura 6, está sendo exibido o diagrama de atividade Gerenciar Financeiro, que está relacionado com o caso de uso Gerenciar Financeiro, como caso de uso principal, estendendo três (3) outros casos de usos, como mostrado a seguir:

Figura 6 – Diagrama de atividade Gerenciar Financeiro do QMF



Fonte: Elaborado pelo autor

Neste diagrama, estão disponíveis três (3) opções de fluxos, no qual são:

Adicionar Mov. Financeiro, que permite ao usuário adicionar um dado financeiro. Para isso, o usuário deverá inserir o valor e descrição, respectivamente, que deseja cadastrar. Após isso, ele deve confirmar ou cancelar o cadastro. Se confirmado, o cadastro deverá ser salvo, se a opção “cancelar” for selecionado, o fluxo é encerrado.

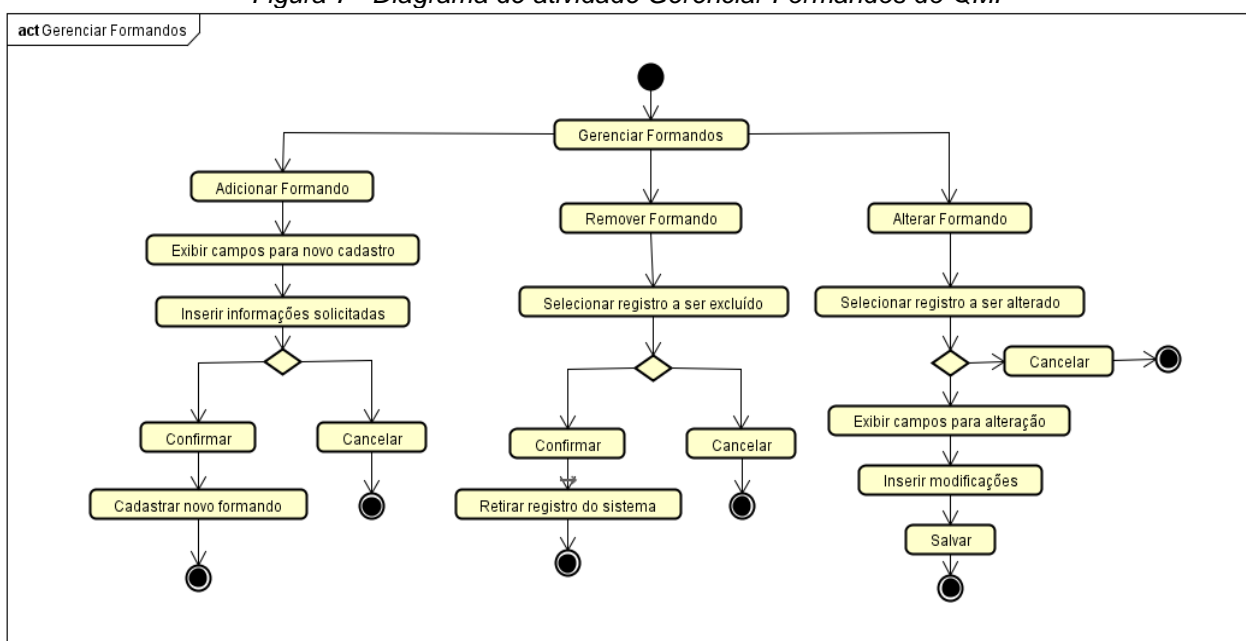
Além disso, o fluxo **Remover Mov. Financeiro**, permitirá que o usuário exclua um dado financeiro já cadastrado no sistema. Ele deverá selecionar o registro que deseja remover e confirmar ou cancelar a ação. Se confirmar, o registro será retirado do sistema, caso contrário, o fluxo será encerrado e o registro continuará cadastrado.

Por fim, o fluxo **Ver Caixa**, possibilita ao usuário a exibição dos movimentos feitos pela turma em um determinado mês. Ele só precisará selecionar o mês que deseja, e o sistema exibirá valores e descrições correspondentes ao mês que foi selecionado.

3.2.10 Gerenciar Formandos

Este diagrama está relacionado aos diagramas de caso de uso Gerenciar Formandos, como caso de uso principal, estendendo a outros três (3) casos, como mostra a figura 7 a seguir:

Figura 7 - Diagrama de atividade Gerenciar Formandos do QMF



Fonte: Elaborado pelo autor

Similar ao diagrama de atividade Gerenciar Financeiro, o diagrama Gerenciar Formandos também possui três (3) opções de fluxos:

Adicionar Formandos, que possibilita ao usuário cadastrar um novo formando, inserindo os dados solicitados pelo sistema. Para efetivar o cadastro, basta o usuário selecionar a opção “confirmar” que aparecerá após as informações de novo cadastro serem inseridas, no entanto, o usuário poderá cancelar o cadastro, selecionando a opção “cancelar”, e o sistema encerrará o fluxo sem que o novo registro seja cadastrado.

O fluxo **Remover Formandos** permitirá ao usuário eliminar um formando já cadastrado, selecionando o registro que deseja remover, após isso, o usuário precisará confirmar a exclusão ou cancelar. Se cancelada a ação, o sistema encerrará o fluxo sem excluir o registro que foi selecionado, caso contrário, o sistema excluirá o registro e encerrará o fluxo.

Por fim, o fluxo **Alterar Formandos** permitirá que o usuário altere dados de um formando já cadastrado, selecionando o registro que deve ser alterado, e, posteriormente, inserindo os dados que deseja mudar. Após isso, o usuário deverá confirmar ou cancelar a ação. Se confirmar, o sistema atualizará os dados e encerrará o fluxo, se cancelar, o sistema encerra o fluxo sem que os dados sejam atualizados.

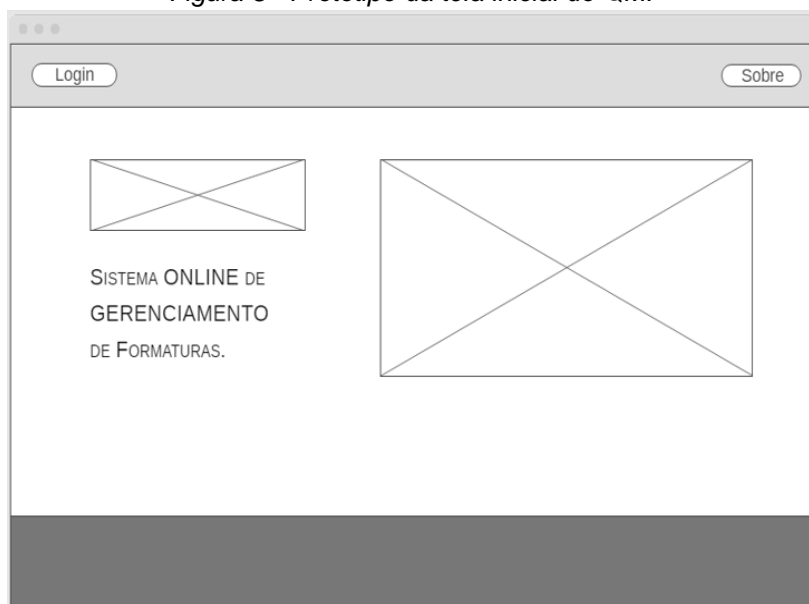
4. PROTOTIPAÇÃO

Nesta seção serão expostas as os protótipos das telas do sistema.

4.1 Tela Inicial

Na tela inicial, mostrada na figura 8, serão exibidos os botões de início, *login* e sobre, no qual o usuário poderá entrar de fato no sistema e conhecer mais informações do software. Também são apresentadas imagens relacionadas a logo e a proposta que o sistema irá oferecer.

Figura 8 - Protótipo da tela inicial do QMF

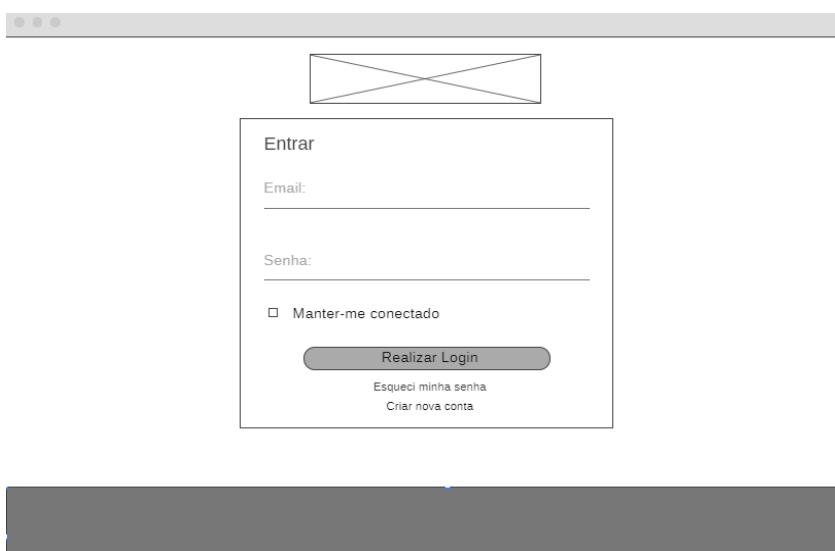


Fonte: Elaborado pelo autor

4.2 Tela de *Login*

Na tela de *login* serão exibidos os campos de texto para o preenchimento dos dados, como *e-mail* e senha, a fim de possibilitar ao usuário a permissão para acessar o painel principal do sistema, além do botão para de fato realizar *login*. Ademais, existem dois links “Esqueci minha senha” e “Criar nova senha” para recuperar a senha e criar uma nova respectivamente. Abaixo, na figura 9, segue foto do protótipo da tela de *login*.

Figura 9 - Protótipo da tela de login do QMF



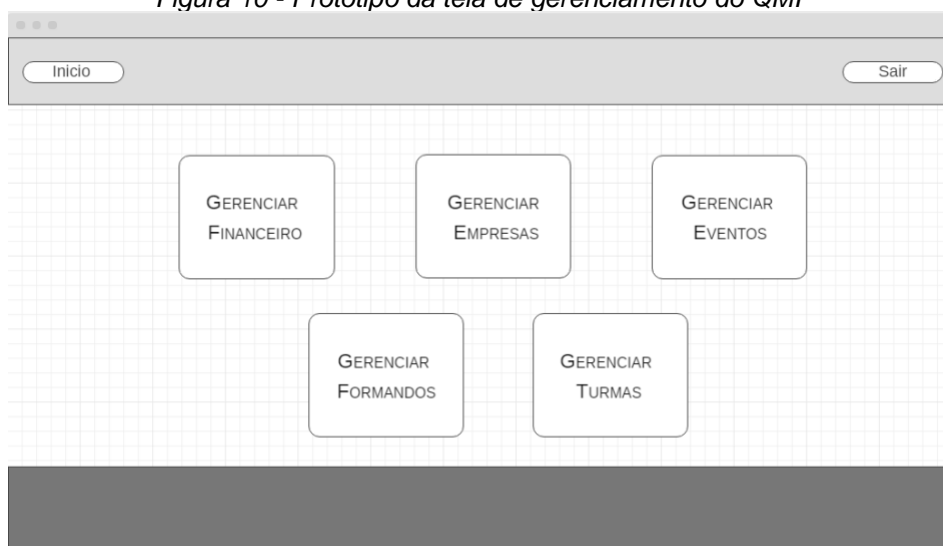
O protótipo da tela de login do QMF apresenta uma interface simples e funcional. No topo, há uma barra cinza com três pontos de menu. Abaixo, um retângulo com uma 'X' indica uma imagem ou logo que não está presente. O formulário principal, intitulado 'Entrar', contém os seguintes elementos: um campo de texto para 'Email:', um campo de texto para 'Senha:', uma opção de checkbox rotulada 'Manter-me conectado', um botão arredondado com o texto 'Realizar Login', e dois links de texto: 'Esqueci minha senha' e 'Criar nova conta'. A base da tela é ocupada por uma barra cinza sólida.

Fonte: Elaborado pelo autor

4.3 Tela de Gerenciamento

Na tela de gerenciamento, conforme é exposto na figura 10, serão exibidos *cards* com as opções de menu que o usuário poderá acessar. Também serão exibidos os botões de “Início” e “Sair”, para voltar ao início e sair da área restrita, respectivamente. Vale ressaltar que este menu superior será fixo a partir do momento em que o usuário inserir e-mail e senha corretos.

Figura 10 - Protótipo da tela de gerenciamento do QMF



Fonte: Elaborado pelo autor

4.4 Tela de Gerenciamento Financeiro

Na figura 11, é demonstrada a tela de Gerenciamento Financeiro, na qual será possível o usuário adicionar algum dado financeiro, clicando no botão “Adicionar” ou ainda remover um dado já cadastrado, clicando no botão “Remover”.

Figura 11 - Protótipo da tela de gerenciamento financeiro do QMF

Descrição	Valor
Livro de Ouro	R\$ 200,00

Fonte: Elaborado pelo autor

4.5 Tela de Gerenciamento Financeiro - Ver Caixa

Nesta tela, como mostra a figura 12, o usuário poderá visualizar as movimentações feitas no mês selecionado por ele. Em “Entradas e Saídas do Mês”, será possível ver as movimentações de forma individual, já em “Balança Geral”, o usuário terá acesso aos movimentos de forma geral.

É importante mencionar que “entradas” é todo dinheiro arrecadado, todo dinheiro que entrou na conta da turma, enquanto “saídas”, como o próprio nome diz, é todo o dinheiro que foi retirado do caixa.

Figura 12 - Protótipo da tela Ver Caixa do QMF

Protótipo da tela Ver Caixa do QMF. A interface apresenta um menu de meses (Jan a Dez) com 'Mar' selecionado. Abaixo, há duas tabelas: 'ENTRADAS E SAÍDAS DO MÊS' e 'BALANÇO GERAL'. A primeira tabela mostra Entradas de R\$ 20,00, Saídas de R\$ 10,00 e Resultado de R\$ 10,00. A segunda tabela mostra Entradas de R\$ 1000,00, Saídas de R\$ 200,00 e Resultado de R\$ 800,00. Botões 'Início', 'Sair' e 'VOLTAR' estão presentes.

ENTRADAS E SAÍDAS DO MÊS	
Entradas:	R\$ 20,00
Saídas:	R\$ 10,00
Resultado:	R\$ 10,00

BALANÇO GERAL	
Entradas:	R\$ 1000,00
Saídas:	R\$ 200,00
Resultado:	R\$ 800,00

Fonte: Elaborado pelo autor

5. CONCLUSÃO

A finalidade deste projeto foi propor e especificar a criação de um sistema web que auxilie alunos que estão a se formar, com a capacidade de controlar informações pessoais (dos alunos que irão se formar) e financeiras (referentes aos gastos e receitas da formatura em geral), de forma eficaz e segura, de modo que os problemas citados pelos componentes das comissões possam ser amenizados. O QMF (sistema proposto) foi projetado para cadastrar, alterar, excluir e exibir informações inseridas no sistema a partir do usuário.

6. TRABALHOS FUTUROS

Os trabalhos futuros serão o desenvolvimento das demais telas de gerenciamento do sistema e o modelo conceitual, assim como a construção de fato da aplicação, para que o sistema possa ser utilizado pelos alunos que formam as comissões de formatura. Além disso, outras funções também foram pensadas para que o sistema possa atender às necessidades não só dos alunos que participam da comissão, mas dos demais estudantes também. Para isso, foram idealizadas funcionalidades como: área restrita específica para alunos que não são da comissão, possibilitando que os mesmos tenham acesso às informações sobre os pacotes oferecidos pelas empresas e ver detalhamento das movimentações do caixa da turma como proposta para melhorar a transparência do que é feito com o dinheiro que entra e sai do caixa. Ademais, também é sugerida a construção de um sistema mobile que permite ao usuário acesso *offline* ao *software*.

7. REFERÊNCIAS BIBLIOGRÁFICAS

GUEDES, Gilleanes Thorwald Araujo. **UML 2 Uma Abordagem Prática**: 2ª Edição. São Paulo: Novatec, 2011.

SOMMERVILLE, Ian. **Engenharia de software**. 9ª Edição. São Paulo: Pearson Prentice Hall, 2011.

LIMA, Davi de. **Modele softwares com Astah Community**. Disponível em:<<https://www.techtudo.com.br/tudo-sobre/astah-commmunity.html>> Acessado em: 10 julho de 2019.

WIREFRAME.CC. **About**. Disponível em:<<https://wireframe.cc>> Acessado em: 20 de julho de 2019.