



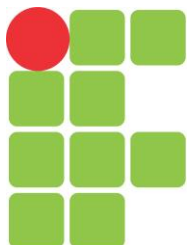
**DIRETORIA ACADÊMICA**  
**CURSO TÉCNICO INTEGRADO EM INFORMÁTICA**

**CLÁUDIA O'ARA AZEVEDO SANTIAGO**

**UMA MODELAGEM DE SOFTWARE PARA GERENCIAMENTO DE  
CEMITÉRIO**

São Gonçalo do Amarante – RN

2017



**INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA**  
**RIO GRANDE DO NORTE**  
Campus São Gonçalo do Amarante

---

### **DIREÇÃO GERAL**

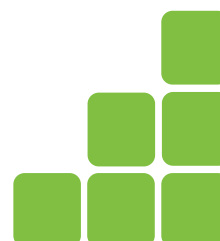
Prof. Dra. Luisa de Marilac de Castro Silva

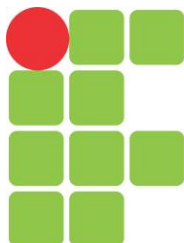
### **DIREÇÃO ACADÊMICA**

Prof. Me. Luiz Alberto Celestino Pessoa Pimentel

### **COORDENAÇÃO DO CURSO TÉCNICO EM INFORMÁTICA**

Prof. Me. Iria Caline Saraiva Cosme





INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
RIO GRANDE DO NORTE  
Campus São Gonçalo do Amarante

---

CLÁUDIA O'ARA AZEVEDO SANTIAGO

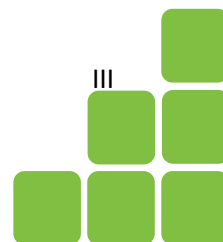
## UMA MODELAGEM DE SOFTWARE PARA GERENCIAMENTO DE CEMITÉRIOS

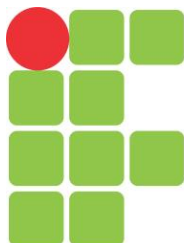
Trabalho de conclusão de curso apresentado à Diretoria Acadêmica do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte – Campus São Gonçalo do Amarante, como requisito básico necessário à obtenção do título de Técnico em Informática.

Orientador: Prof. Dr. Marcelo Damasceno de Melo

São Gonçalo do Amarante – RN

2017





INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
RIO GRANDE DO NORTE  
Campus São Gonçalo do Amarante


CLÁUDIA O'ARA AZEVEDO SANTIAGO

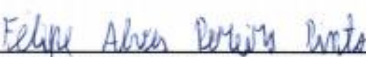
**UMA MODELAGEM DE SOFTWARE PARA GERENCIAMENTO DE  
CEMITÉRIOS: ESTUDOS E MELHORIAS DOS SERVIÇOS PRESTADOS À  
COMUNIDADE**

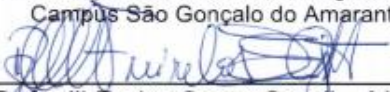
Trabalho de Conclusão de Curso apresentado ao Curso Técnico Integrado em Informática do Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, em cumprimento às exigências legais como requisito parcial à obtenção do título de Técnico em Informática.


Aprovado em: 03/01/2018

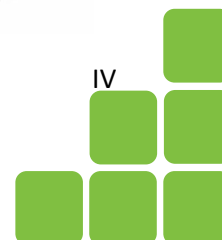
BANCA EXAMINADORA

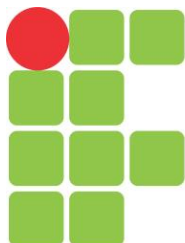
  
\_\_\_\_\_  
Prof. Dr. Marcelo Damasceno de Melo  
Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte  
Campus São Gonçalo do Amarante

  
\_\_\_\_\_  
Prof. Dr. Felipe Alves Pereira Pinto – Membro Interno  
Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte  
Campus São Gonçalo do Amarante

  
\_\_\_\_\_  
Prof. Me. Rafaelli Freire Costa Gentil – Membro Interno  
Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte  
Campus São Gonçalo do Amarante

  
\_\_\_\_\_  
Prof. Dr. Valério Gutemberg de Medeiros Junior – Membro Externo  
Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte  
Campus Parnamirim



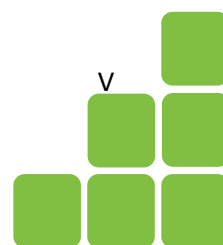


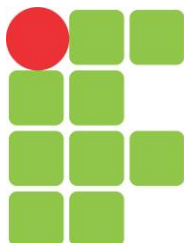
INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
RIO GRANDE DO NORTE  
Campus São Gonçalo do Amarante

---

## Dedicatória

*Este trabalho é dedicado à minha mãe, pois sem a força e o apoio que ela me deu, eu não conseguiria chegar até aqui. Dedico também ao meu avô, que hoje está no céu, mas que sempre incentivou os meus estudos e sempre se orgulhou de mim.*





## Agradecimentos

Ser grato implica em não esquecer, jamais, o bem recebido ou aquele que o praticou. Eu sou grata a Deus por estar comigo e fortalecer a minha fé de acabar o curso cada dia que eu pensava em desistir. Eu sou grata a Ele, pois foi Ele que me manteve de pé durante esses quatro anos e me fez prosseguir. Obrigada, meu Deus.

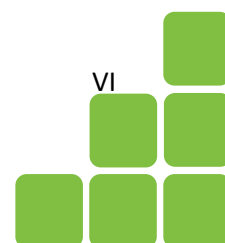
Agradeço também à minha família, que em todos os momentos estiveram presentes nessa caminhada em tudo que eu precisava. Foram eles que me apoiaram e que me ajudaram quando eu achava que nada ia dar certo. Quero agradecer, especialmente, à minha mãe, Denize do Nascimento Azevedo, que depositou e ainda deposita toda sua confiança em mim.

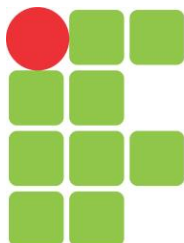
Quero agradecer também a todos os professores pela paciência que tiveram comigo, pois não foi fácil chegar até aqui e é a vocês que devo todo mérito. Se cheguei até aqui, vocês foram os culpados e agradeço por isso.

E, por último (mas não menos importante), quero agradecer a todos os meus colegas. Vocês foram muito especiais e se não fosse o apoio de vocês, eu com certeza não teria conseguido.

Sou imensamente e eternamente grata a todos citados!

**Meu muito obrigado**



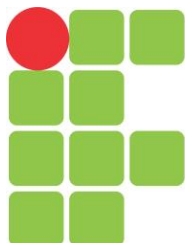


## Resumo

Tendo em vista a dificuldade de administração do cemitério público da comunidade de pescadores em ponta negra, foi feita uma modelagem de software para gerenciamento de cemitérios por meio da linguagem de modelagem unificada (UML) e seus diagramas (diagrama de classes e diagrama de casos de uso). Através dessas modelagens será possível ter uma visualização final completa do software e suas funcionalidades, que serão: cadastrar, listar e editar informações sobre os falecidos, túmulos e requerentes. O software propõe descomplicar as visitas dos familiares que desconhecem a localização do sepulcro, apoia no cadastramento dos finados e das criptas e auxilia o acesso aos documentos necessários.

**Palavras-chaves: Software; Cemitério; Sistema de Cadastro; Java; SQL.**





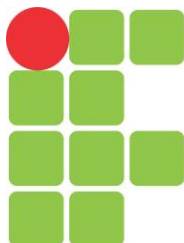
---

## ABSTRACT

In view of difficulties to manage a public cemetery at Ponta Negra, we decided to develop a software that registers, updates and lists the deceased and the tombs of a Ponta Negra necropolis. Thus, we facilitate the search and visualization of site address. The developed software proposes to assist relative visits who do not know the location of tomb, supports registration of the deceased and crypts and it helps to quick access the burial documents.

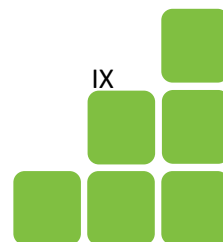
**Keywords:** Software; Cemetery; Registration System; Java; SQL.

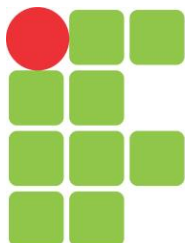




## Lista de Figuras

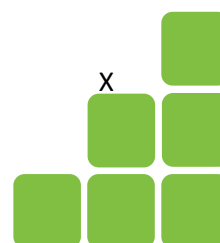
<i>Figura 1 – Modelo Incremental.....</i>	<i>18</i>
<i>Figura 2 – Diagrama de Classes.....</i>	<i>19</i>
<i>Figura 3 – Diagrama de Casos de Uso.....</i>	<i>21</i>
<i>Figura 4– Diagrama Entidade Relacionamento.....</i>	<i>22</i>
<i>Figura 5 – Fluxograma de Cadastro.....</i>	<i>35</i>
<i>Figura 6 – Fluxograma de Edição.....</i>	<i>36</i>
<i>Figura 7 – Fluxograma de Listagem.....</i>	<i>37</i>
<i>Figura 8 – Fluxograma de Caso de Uso: Buscar Falecido/Requerente.....</i>	<i>38</i>
<i>Figura 9 – Fluxograma de Caso de Uso: Buscar Falecido pela data de óbito.....</i>	<i>39</i>

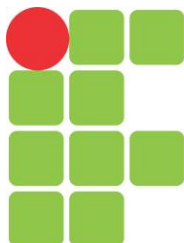




## Lista de Tabelas

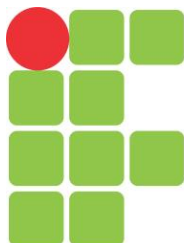
<i>Tabela 1 – Requisitos do Software</i> .....	17
--	----





## Sumário

<b>1 INTRODUÇÃO</b>	<b>13</b>
1.1 FINALIDADE.....	14
1.2 VISÃO GERAL.....	14
<b>2 FUNDAMENTAÇÃO TEÓRICA</b>	<b>14</b>
2.1 UML.....	14
2.2 PORQUE USAR A LINGUAGEM DE MODELAGEM UNIFICADA (UML). .....	15
<b>3 METODOLOGIA</b>	<b>16</b>
3.1 PROCESSO DE DESENVOLVIMENTO DE SOFTWARE.....	16
3.1.1 METODOLOGIAS ÁGEIS	17
3.1.2 MODELO DE PROCESSO DE DESENVOLVIMENTO DE SOFTWARE	17
3.2 DIAGRAMA DE CLASSES.....	19
3.3 DIAGRAMA DE CASOS DE USO .....	20
3.5 DIAGRAMA ENTIDADE RELACIONAMENTO .....	22
<b>4 REQUISITOS DO PROJETO E RECURSOS VISUAIS</b>	<b>23</b>
4.1 CASOS DE USO .....	23
4.1.1 CASO DE USO: CADASTRAR FALECIDO	23
4.1.2 CASO DE USO: CADASTRAR TÚMULO	24
4.1.3 CASO DE USO: EDITAR FALECIDO	25
4.1.4 CASO DE USO: EDITAR TÚMULO	26
4.1.5 CASO DE USO: CADASTRAR REQUERENTE	27
4.1.6 CASO DE USO: EDITAR REQUERENTE	29
4.1.7 CASO DE USO: LISTAR FALECIDO	30
4.1.8 CASO DE USO: LISTAR TÚMULO	30
4.1.9 CASO DE USO: LISTAR REQUERENTE	31
	XI



---

4.1.10 CASO DE USO: BUSCAR FALECIDO/REQUERENTE	32
4.1.11 CASO DE USO: BUSCAR FALECIDO PELA DATA DE ÓBITO	33
<b>4.2 RECURSOS VISUAIS - FLUXOGRAMAS.....</b>	<b>34</b>
4.2.1 CADASTRAR – FLUXOGRAMA	35
4.2.2 EDITAR – FLUXOGRAMA	36
4.2.7 LISTAR – FLUXOGRAMA	37
4.2.10 BUSCAR FALECIDO/REQUERENTE – FLUXOGRAMA	38
4.2.11 BUSCAR FALECIDO PELA DATA DE ÓBITO– FLUXOGRAMA	39
<b>5 CONSIDERAÇÕES FINAIS</b>	<b>39</b>
<b>6 REFERÊNCIAS BIBLIOGRÁFICAS</b>	<b>40</b>

## **1 INTRODUÇÃO**

Segundo Liebens (2003), mapas e bancos de dados precisos podem ser benéficos para pesquisa e gerenciamento de cemitérios. Tendo em vista essa perspectiva, foi criado um projeto para o desenvolvimento de software para o cemitério de Ponta Negra, cujo visa aprimorar a organização e administração do local através da modelagem de software pela linguagem padrão UML.

Será apresentado recursos da UML, tais como: Diagrama de Classes e Diagrama de Casos de Uso. Apresenta o Diagrama de Entidade-Relacionamento, que não faz parte da linguagem UML, mas se aplica à modelagem e também mostra os próprios Casos de Uso, que auxiliam a descrever o cenário que exhibe as funcionalidades do sistema do cemitério do ponto de vista, não só do usuário, mas também do desenvolvedor do sistema.

O cemitério público em estudo é localizado em Ponta Negra no Rio Grande, ele abrange um total de 726 túmulos, sendo 197 sem identificação. Portanto, os aprimoramentos em questão são os cadastros dos sepultados, que será guardado em um banco de dados, assim, o risco de perder informações diminuirá. Como os dados serão guardados no computador também facilitará na hora de procurar algum dado sobre o falecido ou sobre o requerente que o enterrou. O software disponibilizará o cadastro do requerente, que é a pessoa responsável pelo túmulo e pelo falecido.

Haverá cadastro de túmulos, ou seja, será organizado de maneira que estará armazenado no sistema o número do túmulo, sua localização, quem está sepultado dentro, quem é o atual requerente, ou seja, quem possui a carta de aforamento e se o túmulo precisa ser exumado. Isso facilitará na manutenção do cemitério e também na organização e administração.

Além do cadastro, será possível atualizar algum dado sobre os falecidos, túmulos e requerentes, que por ventura, tenha sido cadastrado errado no sistema ou se tiver faltado essa informação na hora do cadastro. Como também, o funcionário poderá fazer buscas no que diz questão ao falecido no software.

## 1.1 FINALIDADE

A finalidade do software para o gerenciamento de cemitérios é reunir todas as informações necessárias dos falecidos, dos túmulos e dos requerentes para melhorar o atendimento ao cliente e auxiliar na organização do órgão público (Cemitério Público de Ponta Negra).

O gerenciador de cemitérios será usado pelos funcionários. Eles utilizarão o sistema para cadastrar os corpos e as necessidades de recursos e para acompanhar a situação do cemitério.

## 1.2 VISÃO GERAL

Essa modelagem de Software contém as seguintes informações:

- **Visão Geral do Projeto** — visa melhorar o controle de dados dentro da instituição pública (Cemitério Público de Ponta Negra).
- **Planos e Diretrizes aplicáveis** — Levantamento de requisitos do sistema de apoio a cemitérios; Modelagem do sistema de apoio a cemitérios; Planejamento de entrega do software de forma incremental; Desenvolvimento incremental de funcionalidades.

## 2 FUNDAMENTAÇÃO TEÓRICA

Esta seção apresenta conceitos sobre as áreas envolvidas no desenvolvimento deste trabalho, tais como a linguagem de modelagem unificada que serviu para o incremento do projeto do software.

### 2.1 UML

Segundo o livro “UML: guia do usuário” dos autores BOOCH, RUMBAUGH e JACOBSON (2006), a UML (Linguagem de Modelagem Unificada) é uma linguagem-padrão para a construção de sistemas de projetos de software. Ela possui diversos tipos de diagramas que são usados para documentar e modelar conceitos das estruturas.

Essa modelagem é considerada conceitual e ela tem sido executada com sucesso em sistemas com projetos de bancos de dados em geral. Os modelos conceituais permitem representar, de forma abstrata, a realidade da aplicação, facilitando a comunicação entre programadores e usuários.

A Linguagem de Modelagem Unificada proporciona uma maneira descomplicada e objetiva de documentação de um sistema, sendo assim de fácil entendimento para o desenvolvedor do software, facilitando a elaboração do sistema. Isso acontece através dos conceitos e diagramas que a linguagem de modelagem oferece. Os diagramas e arquivos que compõe um modelo UML podem ser utilizados para o incremento, exposição e manutenção durante todo o ciclo de vida dos sistemas. Conforme o autor GUEDES (2008), a UML tem em seu conjunto todos os recursos necessários para suprir as necessidades de todas as etapas que compõe um projeto, desde a definição, implementação, criação do modelo de banco de dados, distribuição, enfim, proporcionando sem qualquer outra ferramenta ou metodologia.

## **2.2 PORQUE USAR A LINGUAGEM DE MODELAGEM UNIFICADA (UML).**

Foi proposto que o documento do software fosse esquematizado com base em mapas conceituais, que são diagramas ou ferramentas gráficas que representam visualmente as relações entre conceitos e ideias, também conhecidos como uma interpelação que tem transparecido eficiência no fundamento à aprendizagem. A partir desses mapas, o desenvolvedor pode produzir uma descrição mais rigorosa do sistema utilizando a linguagem UML.

A UML propicia que você esboce uma icnografia do seu sistema. O autor GUEDES (2008) usa a comparação de um construtor que vai realizar um projeto sem antes ter toda a planta que defina estrutura a ser construída. De acordo com o literato, a experiência do construtor garante, até certo ponto, o sucesso do projeto, mas, com certeza, uma vez feito o planejamento, o “cálculo estrutural”, o desenho da planta, a garantia de sucesso antes, durante e depois da efetivação da construção é incomparavelmente maior. O mesmo acontece com um projeto de software, a experiência do desenvolvedor ou analista, não pode substituir a necessidade de um projeto que defina uma “planta” da solução como um todo. Esta “planta” garante, em

todas as fases do projeto, seja na definição, desenvolvimento, homologação, distribuição, utilização e manutenção do mesmo, uma maior clareza e objetividade para execução de cada ação, e, com certeza, quanto maior a solução, maior a necessidade de um projeto definido adequadamente.

Desta forma, a UML é uma linguagem padrão para visualização, especificação, construção e documentação de um aplicativo ou projeto de software, e objetiva aumentar a produtividade, otimizar as etapas que envolvem o desenvolvimento de um sistema, aumentando assim a qualidade do produto a ser implementado. Ela independe da ferramenta em que o aplicativo será desenvolvido. A ideia é prover uma visão lógica de todo o processo de forma a facilitar a sua implementação física.

### **3 METODOLOGIA**

Esta seção apresenta os processos de desenvolvimento de software que foram usados para construir o projeto, tais quais como a metodologia ágil, e o modelo de processo de desenvolvimento de software.

#### **3.1 PROCESSO DE DESENVOLVIMENTO DE SOFTWARE**

Processos de desenvolvimento ágeis são projetados para criar um software útil rapidamente. Geralmente, eles são processos iterativos nos quais a especificação, o projeto, o desenvolvimento e o teste são intercalados. O software não é desenvolvido e disponibilizado integralmente, mas em uma série de incrementos e, cada incremento inclui uma nova funcionalidade do sistema (SOMMERVILLE, 2007).

O Sistema deverá efetuar cadastro de túmulos e falecidos e também editar os cadastros feitos. A funcionalidade do software, em geral, é armazenar os dados dos falecidos, dos túmulos e dos requerentes através do cadastro com inclusão, alteração, listagem e pesquisa:



Tabela 1 – Requisitos do Software

Cadastro de	Dados necessários
<b>Falecido</b>	Número de processos de óbito, órgão emissor, data de óbito, nome, nome do pai, nome da mãe, idade, sexo, estado civil, causa da morte, nome do médico, CRM do médico, número do túmulo.
<b>Túmulo</b>	Número do túmulo, rua, bloco, CPF do requerente.
<b>Requerente</b>	Nome, endereço, telefone 01, telefone 02, CPF, número do túmulo.

Essa tabela descreve os atributos necessários para cadastrar cada entidade (falecido, túmulo, requerente) no sistema. Para um falecido ser sepultado, tem que ter cadastrado pelo menos um requerente e um túmulo.

### 3.1.1 METODOLOGIAS ÁGEIS

Dos Santos Soares (2004) relacionou diversos estudos acerca de metodologias ágeis, expondo que, apesar do surgimento das mesmas ser recente, a maioria dos resultados iniciais, no que diz respeito à qualidade, confiabilidade, prazos de entrega e custos, são muito superiores aos das metodologias tradicionais.

Entre estes estudos foram apresentados dois quadros com curvas de tempo e valores no desenvolvimento de software, percebendo-se uma grande diferença na curva de aumento de valores, sendo que nas metodologias ágeis os gastos foram reduzidos em comparação à outra metodologia devido ao tempo de desenvolvimento e aplicação.

Para Conceição e Silveira, a justificativa para esta redução é porque as metodologias ágeis incentivam as mudanças no decorrer do processo, objetivando fornecer ao cliente o produto que ele realmente precisa.

### 3.1.2 MODELO DE PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

O modelo de processo de desenvolvimento de software escolhido para esse projeto foi o modelo incremental. Para Ferreira (2017), o modelo incremental aplica sequências lineares, de forma escalonada, à medida que o tempo vai avançando. Cada sequência linear gera “incrementos” (entregáveis/aprovados/liberados) do software de maneira similar aos incrementos gerados por um fluxo de processos evolucionários.

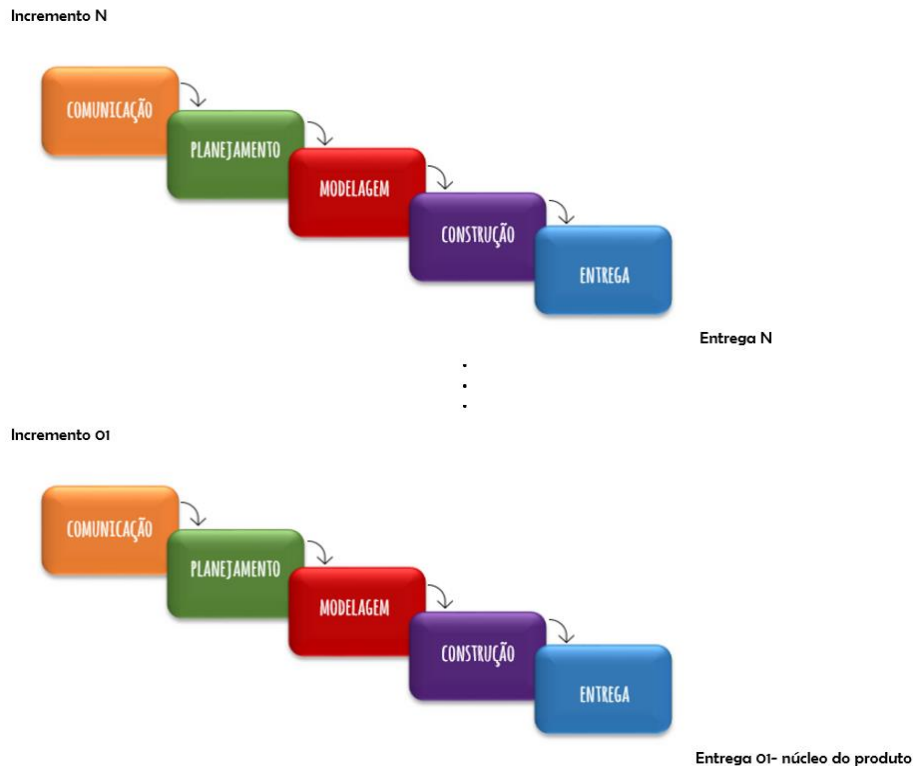


Figura 1 – Modelo Incremental

A figura 1 apresenta o modelo de processo incremental. Verifica-se que ele aplica sequências lineares de forma ordenada, à medida que o tempo for avançando. Cada uma das sequencias gera um incremento no sistema. Esses incrementos são apresentados prontos para o cliente.

Uma das vantagens do modelo incremental para o desenvolvedor é que, o cliente informa seus requisitos e suas preferências para os próximos incrementos de acordo com a apresentação da entrega 01. A implementação torna-se menos complicada tendo em vista que a construção de um sistema menor é sempre menos arriscada que a construção de um grande.

A utilização do modelo incremental reduz o tempo de desenvolvimento de um sistema. Sendo assim, as chances de mudanças nos requisitos do usuário durante o desenvolvimento são menores. Conseqüentemente, se um grande erro é cometido, apenas o último incremento é descartado.

### 3.2 DIAGRAMA DE CLASSES

Um diagrama de classes é uma explanação da estrutura e relações das classes que servem de modelo para objetos. Na Unified Modeling Language (UML) em diagrama de classe, uma classe é representada por um retângulo com três divisões, são elas: O nome da classe, seus atributos e por fim os métodos. Vejam abaixo na Figura 2 sua representação:

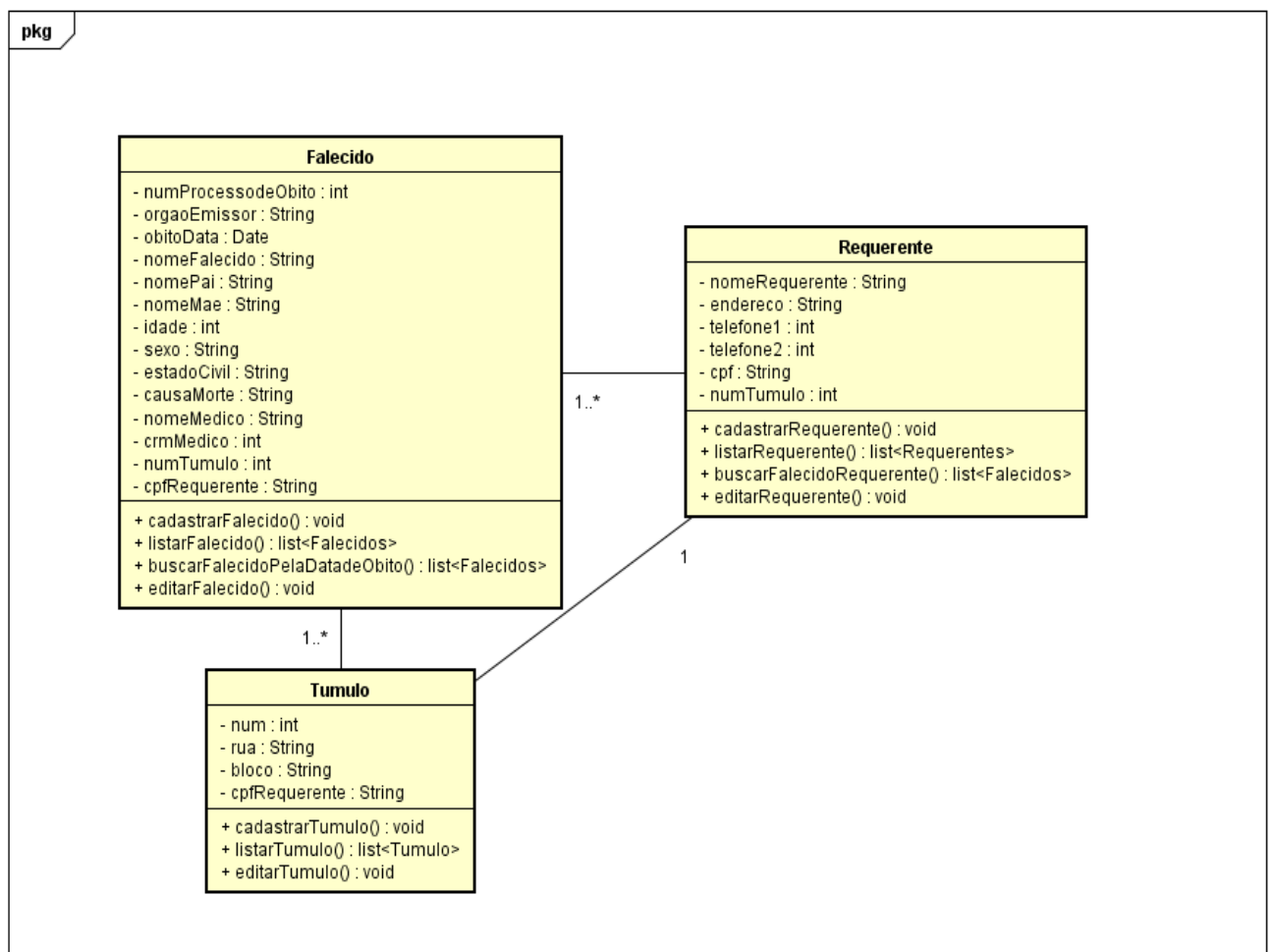


Figura 2 – Diagrama de Classes

O sistema foi modelado a partir de três classes: FALECIDO, REQUERENTE e TÚMULO. A classe 'FALECIDO' tem quatro métodos: cadastrar falecido, listar falecido, buscar falecido pela data de óbito e editar falecido. A classe 'REQUERENTE' tem quatro métodos: cadastrar requerente, listar requerente, buscar requerente e falecido e editar requerente. E a classe 'TÚMULO' tem três métodos: cadastrar túmulo, listar túmulo e editar túmulo.

A multiplicidade de classes diz respeito a quantidade de exigências de um objeto que uma classe pode ter em relação a outra. A relação entre as classes FALECIDO - TUMULO e FALECIDO-REQUERENTE é igual. É uma relação de multiplicidade obrigatória de UM para N, ou seja, um requerente pode ser responsável por N falecidos, mas o falecido só pode ser responsabilidade de um requerente. Assim como para túmulo, ou seja, um túmulo pode ter N falecidos, mas o falecido só pode estar em um túmulo. Entretanto a relação das classes TUMULO e REQUERENTE é diferente. É uma relação de multiplicidade obrigatória um para um. Isso implica em dizer que um requerente pode ser responsável apenas por um túmulo e o túmulo é responsabilidade só de um requerente.

### **3.3 DIAGRAMA DE CASOS DE USO**

Esse diagrama documenta o que o sistema faz do ponto de vista do usuário. Em outras palavras, ele descreve as principais funcionalidades do sistema e a interação dessas funcionalidades com os usuários do mesmo sistema. Um diagrama de Caso de Uso descreve um cenário que mostra as funcionalidades do sistema do ponto de vista do usuário.

Ele tem o objetivo de auxiliar a comunicação entre os analistas e o cliente. O cliente deve ver no diagrama de Casos de Uso as principais funcionalidades de seu sistema.

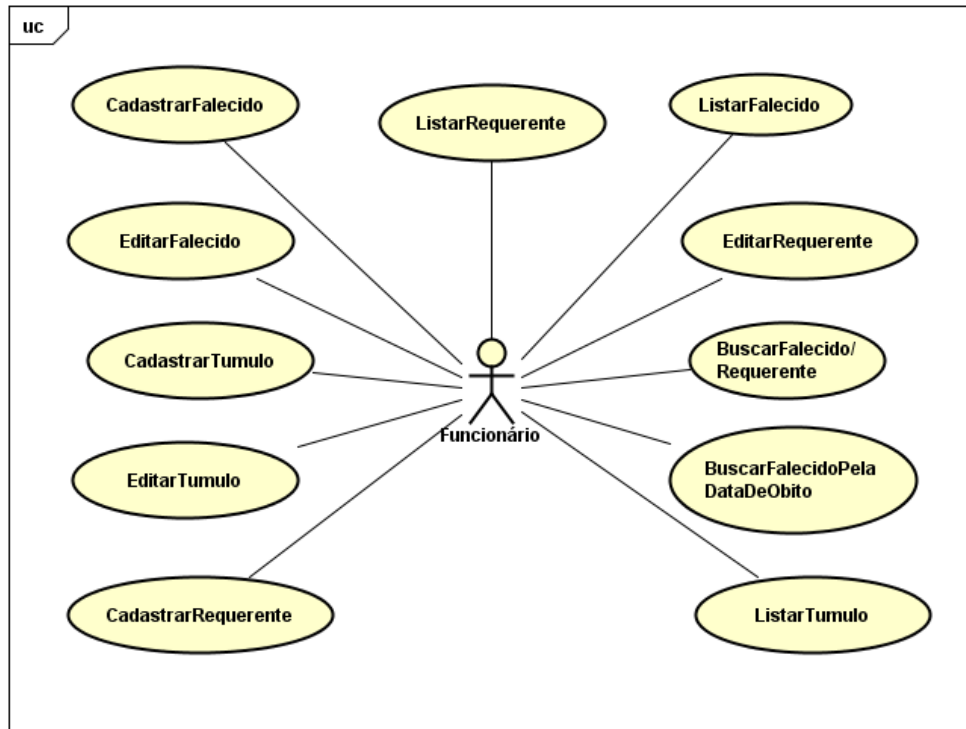


Figura 3: Diagrama de Casos de Uso

Esse diagrama representa as entidades e quem é responsável por administrá-las. As classes e suas funcionalidades são:

- **Cadastrar Falecido:** cadastra o falecido que chegar ao cemitério para ser sepultado.
- **Editar Falecido:** edita algum dado do falecido que por ventura cadastraram errado.
- **Cadastrar Túmulo:** cadastra os túmulos do cemitério.
- **Editar Túmulo:** edita dados cadastrados dos túmulos.
- **Cadastrar Requerente:** cadastra os requerentes que forem sepultar os falecidos.
- **Editar Requerente:** edita dados do requerente que foram fornecidos errado.
- **Listar Falecido:** faz uma lista de todos os falecidos que estão cadastrados no sistema.
- **Listar Túmulo:** faz uma lista de todos os túmulos que estão cadastrados no sistema.
- **Listar Requerente:** faz uma lista de todos os requerentes que estão cadastrados no sistema.

- **Buscar Falecido/Requerente:** faz uma busca utilizando o CPF do requerente para achar o requerente e o (s) falecido (s) que ele é responsável.
- **Buscar Falecido pela data de óbito:** busca pela data de óbito todos os falecidos que morreram nesse dia.

### 3.5 DIAGRAMA ENTIDADE RELACIONAMENTO

Segundo RODRIGUES (2017), o diagrama Entidade Relacionamento (Diagrama ER) é a representação gráfica e principal ferramenta do modelo conceitual (modelo Entidade Relacionamento), que serve para descrever os objetos (entidades) envolvidos em um domínio de negócios, com suas características (atributos) e como elas se relacionam entre si (relacionamentos).

O diagrama facilita ainda a comunicação entre os integrantes da equipe, pois oferece uma linguagem comum utilizada tanto pelo analista, responsável por levantar os requisitos, e os desenvolvedores, responsáveis por implementar aquilo que foi modelado.

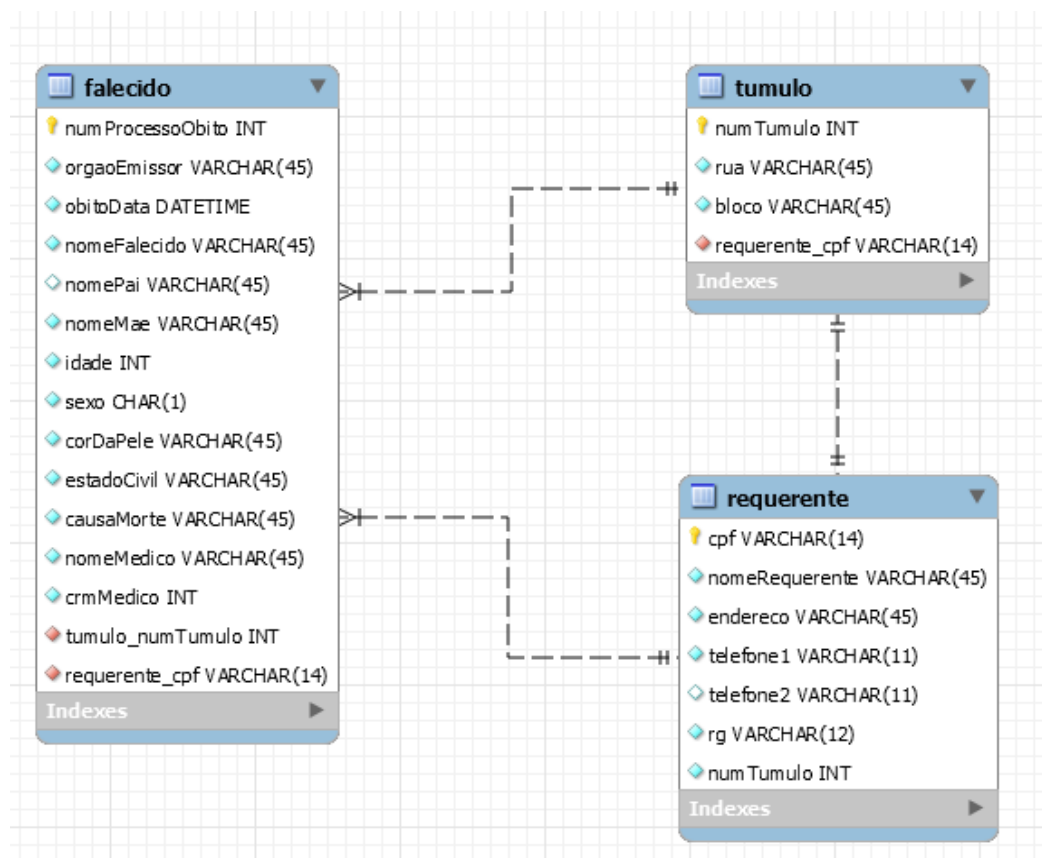


Figura 4: Diagrama Entidade Relacionamento

Nesse diagrama podemos ver as entidades: FALECIDO, REQUERENTE e TÚMULO. Cada entidade tem seus atributos e a relação entre as classes é feita de forma que cada túmulo pode ter N falecidos, entretanto, cada falecido só pode estar em um túmulo, que é igual a relação entre as classes FALECIDO e REQUERENTE, ou seja, cada requerente é responsável por N falecidos, mas um falecido só pode ser responsabilidade de um requerente. E a última cardinalidade, que é diferente das demais: a relação entre REQUERENTE e TÚMULO. Cada requerente só pode ser responsável por um túmulo e um túmulo pode ser responsabilidade apenas de um requerente.

## 4 REQUISITOS DO PROJETO E RECURSOS VISUAIS

Nessa seção serão apresentados os requisitos necessários para o projeto. Esses requisitos serão expostos de forma verbal (através dos casos de uso) e não-verbal (através dos fluxogramas).

### 4.1 CASOS DE USO

Os casos de uso são, especificamente, requisitos de comportamento para sistemas, que serve para descrever os processos do software.

#### 4.1.1 CASO DE USO: CADASTRAR FALECIDO

**Ator primário:** Funcionário do cemitério.

**Meta de contexto:** Armazenar no software a declaração de óbito do falecido, a carta de aforamento e uma documentação com foto do requerente, para que o falecido possa ser sepultado. E também inserir esses dados no banco de dados do sistema.

**Precondições:** Ter todos os dados necessários (declaração de óbito do falecido, carta de aforamento e documentação com foto do requerente) para poder cadastrar o falecido no software.

**Disparador:** O funcionário decide “Registrar novo falecido”.

**Cenário:**

**Funcionário:** colhe com o requerente a declaração de óbito do falecido, a carta de aforamento e documentação com foto do requerente.

**Funcionário:** introduz no software, através do escaneamento, as documentações colhidas.

**Funcionário:** seleciona “Registrar” ou “Cancelar”.

**Funcionário:** aguarda a mensagem de confirmação ou de cancelamento.

**Exceções:**

- O falecido não ter todas as informações necessárias para o cadastro: o funcionário verifica quais dados do falecido estão em falta para poder preencher.
- Falecido sem identificação.
- Falecido sem algum responsável.

**Prioridade:** Essencial, deve ser implementada.

**Quando disponível:** Primeiro incremento.

**Frequência de uso:** Toda vez que chegar ao cemitério um novo falecido para ser sepultado.

**Canal com o ator:** Via interface do software.

**Atores secundários:** Desenvolvedores do software.

**Canais com os atores secundários:** Linha telefônica ou através do IFRN – São Gonçalo do Amarante.

#### 4.1.2 CASO DE USO: CADASTRAR TÚMULO

**Ator primário:** Funcionário do cemitério.



**Meta de contexto:** Armazenar no software os documentos do requerente, consequentemente, responsável pelo túmulo. E também inserir esses dados no banco de dados do sistema.

**Precondições:** Ter autorização da prefeitura e documento com seu nome declarando que o túmulo pertence a você.

**Disparador:** O funcionário decide “Registrar novo túmulo”

**Cenário:**

**Funcionário:** colhe com o requerente a carta de aforamento e documentação com foto.

**Funcionário:** introduz no software, através do escaneamento, as documentações colhidas.

**Funcionário:** seleciona “Registrar” ou “Cancelar”.

**Funcionário:** aguarda a mensagem de confirmação ou de cancelamento.

**Exceções:**

- A carta de aforamento estar no nome de uma pessoa já falecida.

**Prioridade:** Essencial, deve ser implementada.

**Quando disponível:** Primeiro incremento.

**Frequência de uso:** Toda vez que for preciso registrar um novo túmulo.

**Canal com o ator:** Via interface do software.

**Atores secundários:** Desenvolvedores do software.

**Canais com os atores secundários:** Linha telefônica ou através do IFRN – São Gonçalo do Amarante.

#### 4.1.3 CASO DE USO: EDITAR FALECIDO

**Ator primário:** Funcionário do cemitério.

**Meta de contexto:** Buscar no software o falecido pelo CPF do seu requerente, afim de mudar algum dado informado incorretamente.

**Precondições:** O requerente ter seu documento em mãos para consultar seu CPF.

**Disparador:** O requerente vai até o cemitério para atualizar algum dado e o funcionário decide “Editar Falecido”.

**Cenário:**

**Funcionário:** colhe com o seu documento de identificação com foto.

**Funcionário:** introduz no software buscando pelas informações do falecido.

**Funcionário:** seleciona “Editar” ou “Cancelar”.

**Funcionário:** aguarda a mensagem de confirmação ou de cancelamento.

**Exceções:**

- Outro familiar querer atualizar alguma informação do falecido.
- O requerente não levar documentação.

**Prioridade:** Essencial, deve ser implementada.

**Quando disponível:** Primeiro incremento.

**Frequência de uso:** Toda vez que for necessário atualizar um falecido.

**Canal com o ator:** Via interface do software.

**Atores secundários:** Desenvolvedores do software.

**Canais com os atores secundários:** Linha telefônica ou através do IFRN – São Gonçalo do Amarante.

#### 4.1.4 CASO DE USO: EDITAR TÍTULO

**Ator primário:** Funcionário do cemitério.

**Meta de contexto:** Mudar alguma informação que diz respeito ao túmulo.

**Precondições:** O requerente ter documento com foto e carta de aforamento em mãos para a confirmação dos dados.

**Disparador:** O requerente vai até o cemitério para atualizar algum dado e o funcionário decide “Editar Túmulo”.

**Cenário:**

**Funcionário:** colhe com o seu documento de identificação com foto.

**Funcionário:** introduz no software buscando pelas informações do túmulo.

**Funcionário:** seleciona “Editar” ou “Cancelar”.

**Funcionário:** aguarda a mensagem de confirmação ou de cancelamento.

**Exceções:**

- Outro familiar querer atualizar alguma informação do falecido.
- O requerente não levar documentação.

**Prioridade:** Essencial, deve ser implementada.

**Quando disponível:** Primeiro incremento.

**Frequência de uso:** Toda vez que for necessário atualizar um túmulo.

**Canal com o ator:** Via interface do software.

**Atores secundários:** Desenvolvedores do software.

**Canais com os atores secundários:** Linha telefônica ou através do IFRN – São Gonçalo do Amarante.

#### 4.1.5 CASO DE USO: CADASTRAR REQUERENTE

**Ator primário:** Funcionário do cemitério.

**Meta de contexto:** Armazenar no software a carta de aforamento e uma documentação com foto, para que o requerente possa sepultar falecidos no cemitério. E também inserir esses dados no banco de dados do sistema.

**Precondições:** Ter todos os dados necessários (carta de aforamento e documentação com foto) para poder cadastrar o requerente no software.

**Disparador:** O funcionário decide “Registrar novo requerente”.

**Cenário:**

**Funcionário:** colhe com o requerente a declaração a carta de aforamento e sua documentação com foto.

**Funcionário:** introduz no software, através do escaneamento, as documentações colhidas.

**Funcionário:** seleciona “Registrar” ou “Cancelar”.

**Funcionário:** aguarda a mensagem de confirmação ou de cancelamento.

**Exceções:**

- O requerente não ter todas as informações necessárias para o cadastro: o funcionário verifica quais dados do requerente estão em falta para poder preencher.
- Carta de aforamento estar no nome de outro falecido.

**Prioridade:** Essencial, deve ser implementada.

**Quando disponível:** Primeiro incremento.

**Frequência de uso:** Toda vez que chegar ao cemitério um novo requerente.

**Canal com o ator:** Via interface do software.

**Atores secundários:** Desenvolvedores do software.

**Canais com os atores secundários:** Linha telefônica ou através do IFRN – São Gonçalo do Amarante.

#### 4.1.6 CASO DE USO: EDITAR REQUERENTE

**Ator primário:** Funcionário do cemitério.

**Meta de contexto:** Mudar alguma informação que diz respeito ao requerente.

**Precondições:** O requerente ter documento com foto e carta de aforamento em mãos para a confirmação dos dados.

**Disparador:** O requerente vai até o cemitério para atualizar algum dado e o funcionário decide “Editar Requerente”.

**Cenário:**

**Funcionário:** colhe com o seu documento de identificação com foto.

**Funcionário:** introduz no software buscando pelas informações do requerente.

**Funcionário:** seleciona “Editar” ou “Cancelar”.

**Funcionário:** aguarda a mensagem de confirmação ou de cancelamento.

**Exceções:**

- Outro familiar querer atualizar alguma informação do requerente.
- O requerente não levar documentação.

**Prioridade:** Essencial, deve ser implementada.

**Quando disponível:** Primeiro incremento.

**Frequência de uso:** Toda vez que for necessário atualizar um requerente.

**Canal com o ator:** Via interface do software.

**Atores secundários:** Desenvolvedores do software.

**Canais com os atores secundários:** Linha telefônica ou através do IFRN – São Gonçalo do Amarante.

#### 4.1.7 CASO DE USO: LISTAR FALECIDO

**Ator primário:** Funcionário do cemitério.

**Meta de contexto:** Listar todos os falecidos armazenados junto com o seu número de óbito.

**Precondições:** Ter acesso ao software.

**Disparador:** O funcionário decide “Listar falecidos”.

**Cenário:**

**Funcionário:** decide listar os falecidos cadastrados no software.

**Funcionário:** seleciona “Listar” ou “Cancelar”.

**Funcionário:** aguarda a mensagem de confirmação ou de cancelamento.

**Exceções:**

- Falta de qualquer dado no cadastro.
- Determinada informação ser inválida.

**Prioridade:** Essencial, deve ser implementada.

**Quando disponível:** Primeiro incremento.

**Frequência de uso:** Toda vez que for necessário listar os falecidos que estão sepultados no cemitério.

**Canal com o ator:** Via interface do software.

**Atores secundários:** Desenvolvedores do software.

**Canais com os atores secundários:** Linha telefônica ou através do IFRN – São Gonçalo do Amarante.

#### 4.1.8 CASO DE USO: LISTAR TÚMULO

**Ator primário:** Funcionário do cemitério

**Meta de contexto:** Listar todos os túmulos do cemitério junto com o seu número de identificação.

**Precondições:** Ter acesso ao software.

**Disparador:** O funcionário decide “Listar túmulos”

**Cenário:**

**Funcionário:** decide listar os túmulos cadastrados no software.

**Funcionário:** seleciona “Listar” ou “Cancelar”.

**Funcionário:** aguarda a mensagem de confirmação ou de cancelamento.

**Exceções:**

- Falta de qualquer dado no cadastro.
- Determinada informação ser inválida.

**Prioridade:** Essencial, deve ser implementada.

**Quando disponível:** Primeiro incremento.

**Frequência de uso:** Toda vez que for necessário listar os túmulos que possuem no cemitério.

**Canal com o ator:** Via interface do software.

**Atores secundários:** Desenvolvedores do software.

**Canais com os atores secundários:** Linha telefônica ou através do IFRN – São Gonçalo do Amarante.

#### 4.1.9 CASO DE USO: LISTAR REQUERENTE

**Ator primário:** Funcionário do cemitério.

**Meta de contexto:** Listar todos os requerentes armazenados junto com o seu CPF.

**Precondições:** Ter acesso ao software.

**Disparador:** O funcionário decide “Listar requerentes”.

**Cenário:**

**Funcionário:** decide listar os requerentes cadastrados no software.

**Funcionário:** seleciona “Listar” ou “Cancelar”.

**Funcionário:** aguarda a mensagem de confirmação ou de cancelamento.

**Exceções:**

- Falta de qualquer dado no cadastro.
- Determinada informação ser inválida.

**Prioridade:** Essencial, deve ser implementada.

**Quando disponível:** Primeiro incremento.

**Frequência de uso:** Toda vez que for necessário listar os requerentes cadastrados no sistema.

**Canal com o ator:** Via interface do software.

**Atores secundários:** Desenvolvedores do software.

**Canais com os atores secundários:** Linha telefônica ou através do IFRN – São Gonçalo do Amarante.

#### 4.1.10 CASO DE USO: BUSCAR FALECIDO/REQUERENTE

**Ator primário:** Funcionário do cemitério.

**Meta de contexto:** Buscar o falecido juntamente com o requerente responsável por ele através do CPF do requerente.

**Precondições:** Ter acesso ao software.

**Disparador:** O funcionário decide “Buscar falecido/requerente”.



**Cenário:**

**Funcionário:** decide buscar o falecido e o requerente.

**Funcionário:** seleciona “Buscar” ou “Cancelar”.

**Funcionário:** aguarda a mensagem de confirmação ou de cancelamento.

**Exceções:**

- Falta de qualquer dado no cadastro.
- Determinada informação ser inválida.
- O requerente não levar documentação.
- Falecido sem identificação.
- Falecido sem algum responsável.

**Prioridade:** Essencial, deve ser implementada.

**Quando disponível:** Primeiro incremento.

**Frequência de uso:** Toda vez que for necessário fazer uma busca por algum falecido ou requerente cadastrados no sistema.

**Canal com o ator:** Via interface do software.

**Atores secundários:** Desenvolvedores do software.

**Canais com os atores secundários:** Linha telefônica ou através do IFRN – São Gonçalo do Amarante.

#### 4.1.11 CASO DE USO: BUSCAR FALECIDO PELA DATA DE ÓBITO

**Ator primário:** Funcionário do cemitério.

**Meta de contexto:** Buscar o falecido através da data de óbito.

**Precondições:** Ter acesso ao software.

**Disparador:** O funcionário decide “Buscar falecido pela data de óbito”.

**Cenário:**

**Funcionário:** decide buscar o falecido.

**Funcionário:** seleciona “Buscar” ou “Cancelar”.

**Funcionário:** aguarda a mensagem de confirmação ou de cancelamento.

**Exceções:**

- Falta de qualquer dado no cadastro.
- Determinada informação ser inválida.
- Falecido sem identificação.
- Falecido sem algum responsável.

**Prioridade:** Essencial, deve ser implementada.

**Quando disponível:** Primeiro incremento.

**Frequência de uso:** Toda vez que for necessário fazer uma busca por algum falecido cadastrado no sistema.

**Canal com o ator:** Via interface do software.

**Atores secundários:** Desenvolvedores do software.

**Canais com os atores secundários:** Linha telefônica ou através do IFRN – São Gonçalo do Amarante.

## 4.2 RECURSOS VISUAIS - FLUXOGRAMAS

O uso de fluxogramas nesse trabalho será para facilitar a organização do raciocínio, além de ser mais representativo que um processo escrito, os fluxogramas facilitam a elaboração das tarefas e contém de ser mais representativo que um processo escrito, os fluxogramas facilitam a elaboração das tarefas e controlam processos.

#### 4.2.1 CADASTRAR – FLUXOGRAMA

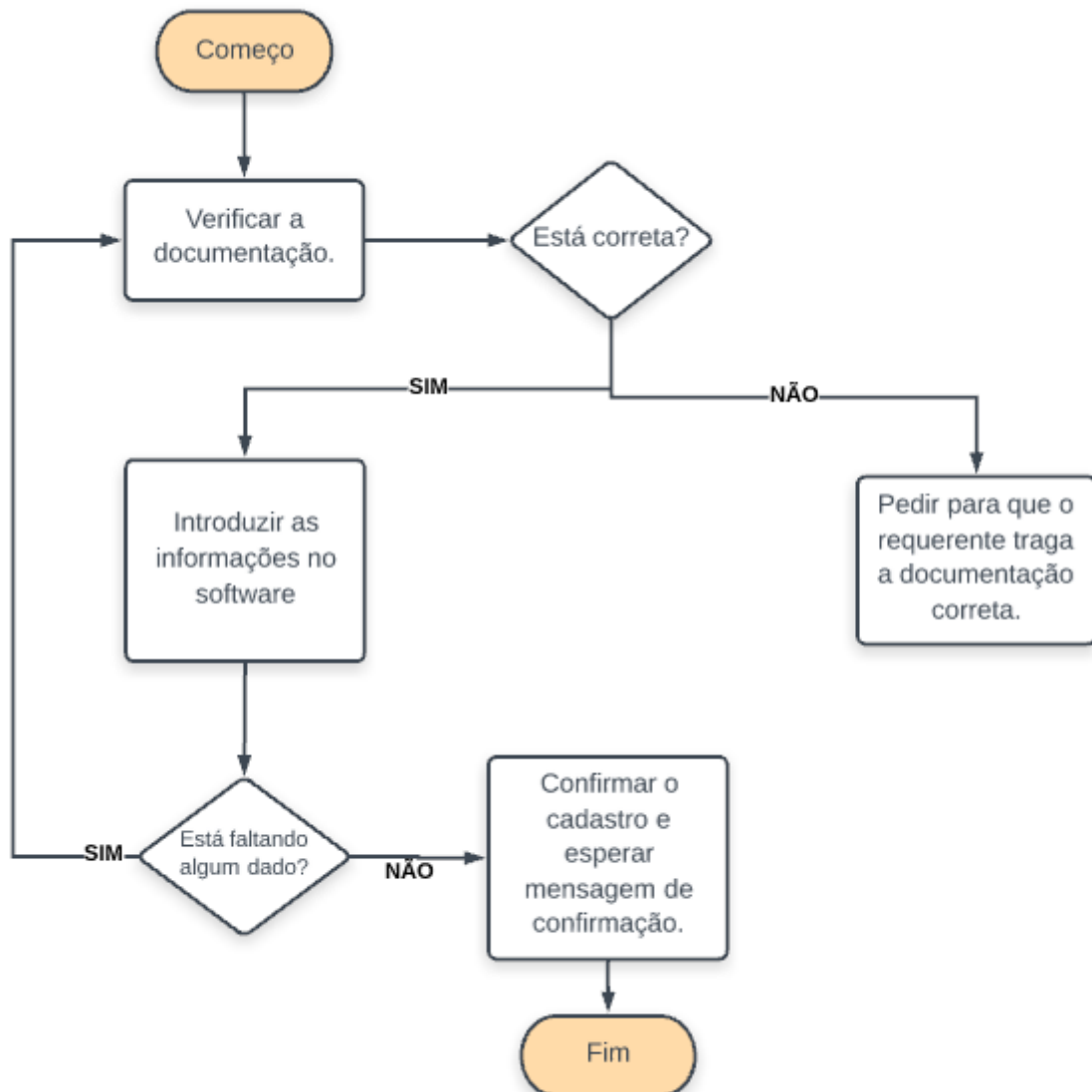


Figura 5: Fluxograma de Cadastro

Esse primeiro cenário inicia-se quando for realizar algum cadastro, seja ele de falecido, requerente ou túmulo. Para isso, o requerente precisa levar toda a documentação. Se a operação for válida, o funcionário do cemitério dará continuidade ao cadastro. Se não, ele cancelará e o sistema voltará ao início.

#### 4.2.2 EDITAR – FLUXOGRAMA

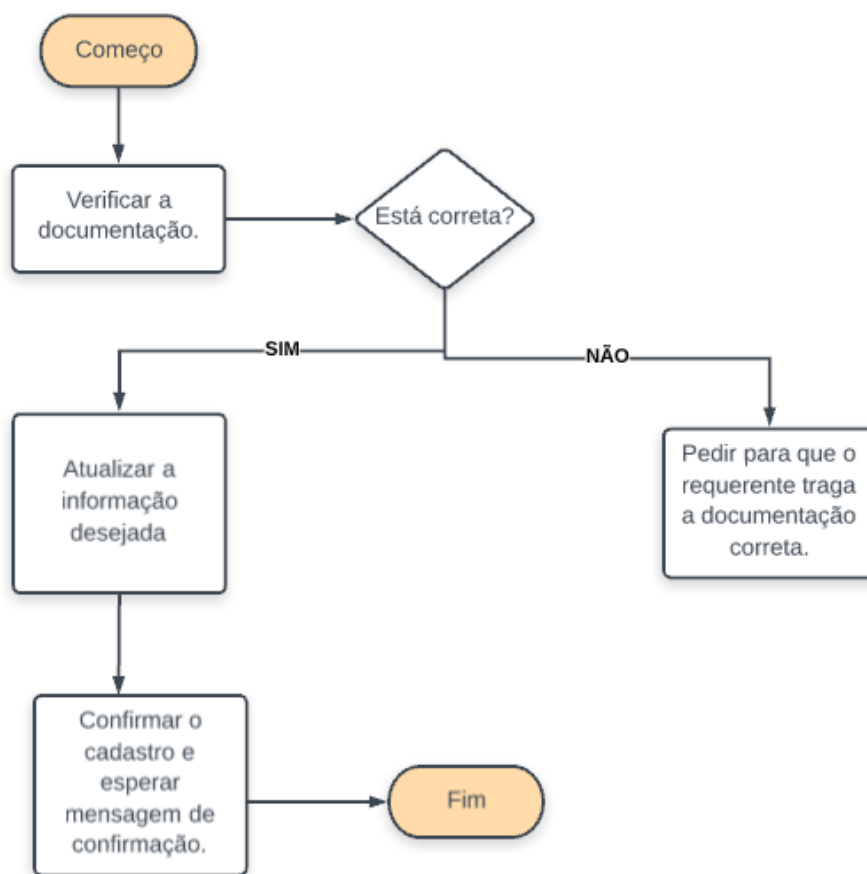


Figura 6: Fluxograma de Edição

O funcionário realiza esse procedimento quando o requerente chega ao cemitério para editar alguma informação sobre um falecido, requerente ou túmulo cadastrado no sistema. Para isso, o requerente precisa levar um documento seu com foto. Se a operação for válida, o funcionário do cemitério dará continuidade à alteração do dado. Se não, ele cancelará e o sistema voltará ao início.

#### 4.2.7 LISTAR – FLUXOGRAMA

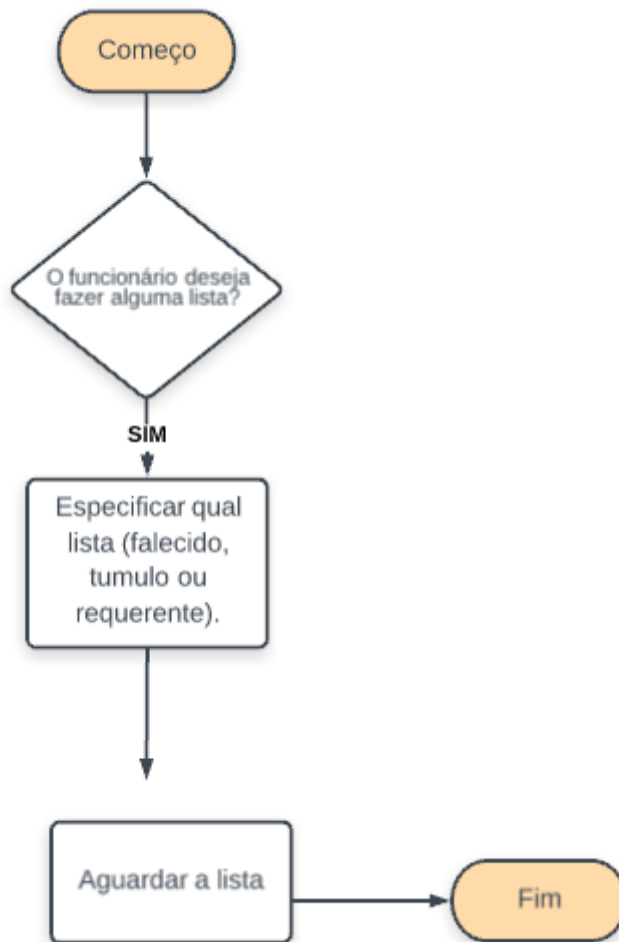


Figura 7: Fluxograma de Listagem

Isso acontece quando o funcionário do cemitério decide fazer uma lista de todos os cadastros que possuem no cemitério.

#### 4.2.10 BUSCAR FALECIDO/REQUERENTE – FLUXOGRAMA

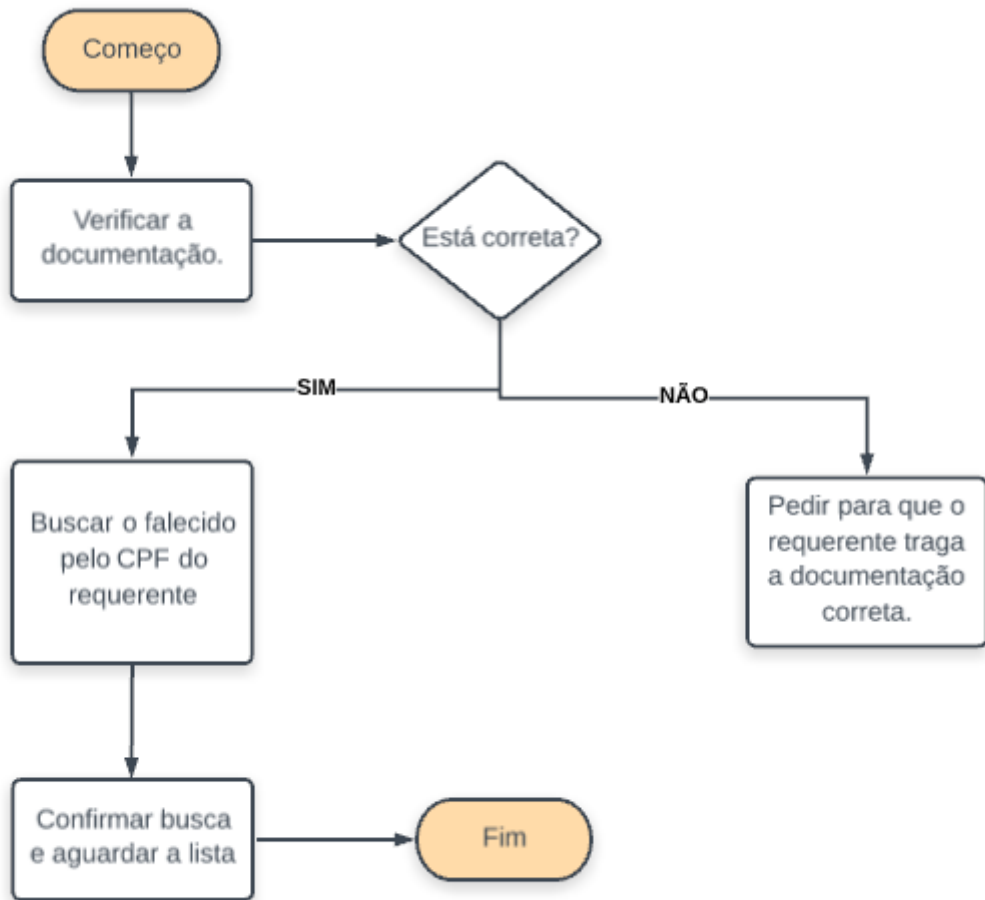


Figura 8: Fluxograma de Casos Uso: Buscar Falecido/Requerente

O funcionário realiza essa ação quando o requerente, através do seu CPF, decide realizar uma busca de um falecido. Para isso, o requerente precisa levar um documento seu com foto. Se a operação for válida, o funcionário do cemitério dará continuidade à busca do falecido. Se não, ele cancelará e o sistema voltará ao início.

#### 4.2.11 BUSCAR FALECIDO PELA DATA DE ÓBITO– FLUXOGRAMA

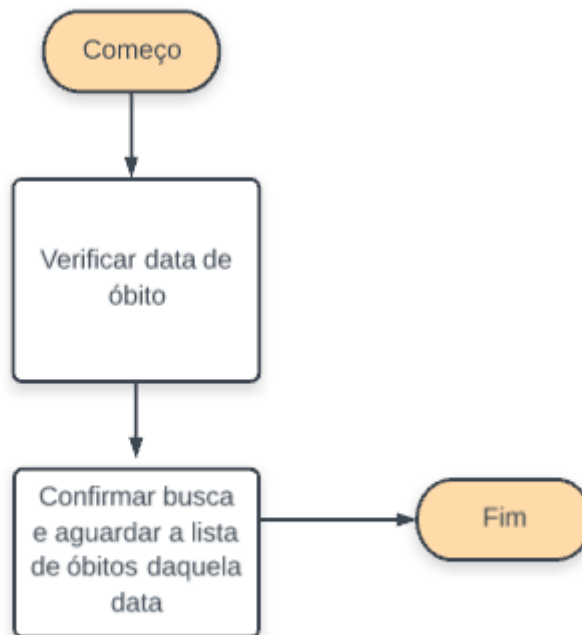


Figura 9: Fluxograma de Casos Uso: Buscar Falecido Pela Data de Óbito

O funcionário realiza esse procedimento quando o requerente decide realizar uma busca de um falecido através da data de óbito. Para isso, o requerente precisa saber a data. Se a operação for válida, o funcionário do cemitério dará continuidade à busca do falecido. Se não, ele cancelará e o sistema voltará ao início.

## 5 CONSIDERAÇÕES FINAIS

Os modelos conceituais utilizados (Diagrama de Classes e Diagrama de Casos de Uso) no desenvolvimento dessa modelagem foram ministrados em sala nas aulas de Projeto de Desenvolvimento de Software, tais quais o diagrama de caso de uso e o diagrama de classes. São ferramentas poderosas no desenvolvimento software, permitindo a construção de sistema seguros.

Este trabalho permitiu o aprimoramento dos meus conhecimentos na área, a qual é um ramo que tem muito a se expandir. Até porque como disse Charles Antony Richard

Hoare, há duas formas de construir um projeto de software: uma maneira de fazer isso deve ser tão simples que, obviamente, não deixe deficiências, e a outra forma é a de torná-lo tão complicado que não percebem as evidentes deficiências. O primeiro método é muito mais difícil. Acredito que isso caiba perfeitamente na importância da modelagem de projeto.

Devido à falta de tempo, a modelagem não seguiu a arquitetura esperada, então o ponto mais importante para trabalhos futuros seria seguir a arquitetura de software MVC. A maior dificuldade foi a falta de comunicação e retorno no que diz respeito ao funcionário do cemitério, por essa razão, alguns detalhes e requisitos estão a definir, o que ficaria também para trabalhos futuros.

Espera-se que a partir do desenvolvimento do presente estudo permita uma análise de como um software feito sob encomenda pode melhorar a organização e administração do Cemitério Público de Ponta Negra.

## 6 REFERÊNCIAS BIBLIOGRÁFICAS

**Booch, Grady, James Rumbaugh, and Ivar Jacobson.** *UML: guia do usuário*. Elsevier Brasil, 2006.

**Da Conceição, Juliano, and Sidnei Renato Silveira.** "Aplicação de Metodologias Ágeis para Desenvolvimento de Software: um Estudo de Caso na Empresa Alliance Software."

**Dos Santos Soares, Michel.** "Metodologias ágeis extreme programming e scrum para o desenvolvimento de software." *Revista Eletrônica de Sistemas de Informação ISSN 1677-3071 doi: 10.21529/RESI 3.1 (2004)*.

**Ferreira, Elizabette Caldas.** "Proposta de metodologia de gestão de riscos para projetos ágeis de software no Instituto Nacional de Estudos e Pesquisas Anísio Teixeira (INEP)." (2017).

**Guedes, Gilleanes TA.** *UML: uma abordagem prática*. Novatec Editora, 2008.



**Liebens, Johan.** "Map and database construction for an historic cemetery: Methods and applications." *Historical Archaeology* 37.4 (2003): 56-68.

**Nakagawa, Elisa Yumi, P. A. E. Estagiária, and Lina María Garcés Rodríguez.** "Arquitetura de Software.", 2015.

**RODRIGUES, Joel.** Modelo Entidade Relacionamento (MER) e Diagrama Entidade-Relacionamento (DER). Disponível em: < <https://www.devmedia.com.br/modelo-entidade-relacionamento-mer-e-diagrama-entidade-relacionamento-der/14332/> >  
Acesso em 08 de dezembro de 2017.

**Sizo, Amanda Monteiro, Adriano Del Pino Lino, and Eloi Luiz Favero.** "Uma proposta de arquitetura de software para construção e integração de ambientes virtuais de aprendizagem." *RISTI-Revista Ibérica de Sistemas e Tecnologias de Informação* 6 (2010): 17-30.

**Sommerville, Ian.** "Engenharia de Software, 8ª edição, Tradução: Selma Shin Shimizu Mel-nikoff, Reginaldo Arakaki, Edilson de Andrade Barbosa." *São Paulo: Pearson Addison-Wesley* 22 (2007): 103.