

UNIVERSIDADE ESTADUAL DE PONTA GROSSA
PRÓ-REITORIA DE PESQUISA E PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO APLICADA
MESTRADO ACADÊMICO

HENIKE GUILHERME JORDAN VOSS
WILLIAM XAVIER MAUKOSKI

RELATÓRIO DE PROJETO
SISTEMA DE QUANTIFICAÇÃO DE DOENÇAS EM IMAGENS DE PLANTAS

PONTA GROSSA
2017

HENIKE GUILHERME JORDAN VOSS
WILLIAM XAVIER MAUKOSKI

RELATÓRIO DE PROJETO
SISTEMA DE QUANTIFICAÇÃO DE DOENÇAS EM IMAGENS DE PLANTAS

Relatório de projeto da disciplina de
Modelagem de Sistemas Agrícolas do curso
de Pós-Graduação em Computação
Aplicada da Universidade Estadual de
Ponta Grossa.
Prof^ª: Simone Nasser Matos

PONTA GROSSA
2017

SUMÁRIO

| | |
|---|-----------|
| 1. INTRODUÇÃO | 3 |
| 1.1. Importância | 3 |
| 1.2. Métodos de quantificação de doenças..... | 3 |
| 1.2.1. Métodos diretos | 3 |
| 1.3. Amostragem | 8 |
| 2. OBJETIVOS..... | 9 |
| 2.1. Objetivos Gerais | 9 |
| 2.2. Objetivos Específicos..... | 9 |
| 3. DESCRIÇÃO DO SISTEMA | 9 |
| 3.1. Manter..... | 9 |
| 3.2. Gerenciamento..... | 10 |
| 4. ABORDAGEM UTILIZADA PARA APLICAÇÃO DOS PADRÕES DE PROJETO | 10 |
| 4.1. Primeira fase..... | 11 |
| 4.2. Segunda fase | 12 |
| 5. APLICAÇÃO DA ABORDAGEM | 12 |
| 5.1. Primeira fase..... | 12 |
| 5.2. Segunda fase | 13 |
| 5.2.1. Persistência de Dados | 13 |
| 5.2.2. Criacionais | 14 |
| 5.2.3. Estruturais | 20 |
| 5.2.4. Comportamentais | 23 |
| 5.3. Vantagens e Desvantagens..... | 29 |
| 6. CONCLUSÃO | 30 |
| 7. REFERÊNCIAS | 30 |

1. INTRODUÇÃO

1.1. Importância

A quantificação de doenças de plantas, também denominada fitopatometria, visa avaliar os sintomas causados pelos agentes patogênicos nas plantas e seus sinais (estruturas do patógeno associadas aos tecidos doentes).

1.2. Métodos de quantificação de doenças

As doenças podem ser quantificadas por métodos diretos de avaliação dos sintomas e sinais, como a incidência, severidade, intensidade, e métodos indiretos, como a determinação da população do patógeno, sua distribuição espacial, seus efeitos na produção (danos e/ou perdas), a desfolha causada.

1.2.1. Métodos diretos

A quantificação das doenças é baseada na avaliação dos sintomas e sinais, através da proporção de tecido doente, sendo realizada pelos seguintes parâmetros:

Incidência: é o método quantitativo mais comum de medição de doença por ser fácil e rápido, sendo obtido pela contagem de plantas doentes ou órgãos doentes, através do número e/ou porcentagem (frequência) de folhas, folíolos, frutos, ramos infectados, sem levar em consideração a quantidade de doença em cada planta ou órgão individualmente. As avaliações de incidência podem ser feitas de diferentes formas, como nos exemplos a seguir:

- Contagem do número ou % de espigas de milho com carvão
- Número ou % de frutos de maçã com sarna
- Número ou % de plantas de algodoeiro com murcha de Fusarium
- % de fungos patogênicos em testes de patologia de sementes

Severidade: é um método quantitativo e qualitativo, que procura determinar a porcentagem da área de tecido doente (sintomas e/ou sinais visíveis), através da medição direta da área afetada, com medidores de área em computador ou não, chaves descritivas, diagramáticas, medição automática e sensores remotos.

Intensidade: é um termo mais amplo que pode ser expresso como incidência ou severidade. Significa o quanto intensa é a doença ou quão doente está a planta. A incidência é um parâmetro satisfatório para avaliar a intensidade de doenças, como murchas e viroses, pois a correlação é alta entre incidência e severidade, pelo fato da doença afetar a planta toda.

Para maioria das doenças foliares esta correlação é baixa (incidência de 100% de plantas com ferrugem, não reflete a intensidade real no campo, pois apesar de todas as plantas apresentarem pústulas de ferrugem, a quantidade de pústulas por folha pode ser baixa, causando pouco dano). Ao contrário da incidência a intensidade está estreitamente relacionada com a perda de produção.

Numa epidemia de doenças foliares deve-se levar em consideração que, quando a incidência é elevada (maioria das plantas com sintomas), a evolução da doença dá-se quase que exclusivamente pelo aumento do número e tamanho das lesões (severidade).

As avaliações de intensidade/severidade podem ser feitas de diferentes formas:

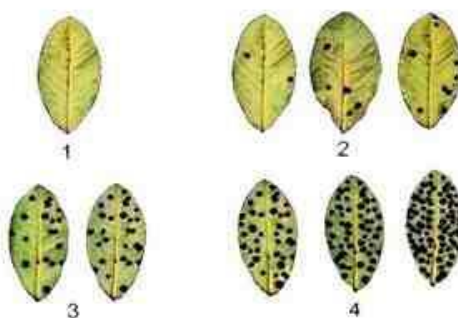
I - Medição direta dos sintomas da doença, através da contagem do número de lesões, medição de seu diâmetro, cálculo da área infectada por folíolo [$Si = \text{No médio de lesões} / \text{folíolo} \times (\text{diâmetro médio das lesões} / 2) \times 3,1416$] e índice de infecção ($I\% = Si \times 100 / S$ total), como nas manchas castanha e preta do amendoim.

II - Medição visual dos sintomas da doença - os patologistas usam a fotocélula humana (olho) para estimar as intensidades através da medição de áreas doentes e valores de infecção. Para este tipo de medição deve ser considerada a Lei de Weber-Fechner, segundo a qual, a acuidade visual é proporcional ao logaritmo da intensidade de estímulo. Dessa forma descreve 12 graus de intensidade ou severidade das doenças:

| | |
|--------------|---------------------|
| 1 - 0% | 7 - 50 – 75% |
| 2 - 0 – 3% | 8 - 75 – 87% |
| 3 - 3 – 6% | 9 - 87 – 94% |
| 4 - 6 – 12% | 10 - 94 – 97% |
| 5 - 12 – 25% | 11 - 97 – 100% |
| 6 - 25 – 50% | 12 - 100% de doença |

Usando esses princípios, as medições visuais da intensidade das doenças podem ser feitas através do uso de chaves descritivas, classes de intensidade, diagramas padrões (James, 1971) ou escalas diagramáticas.

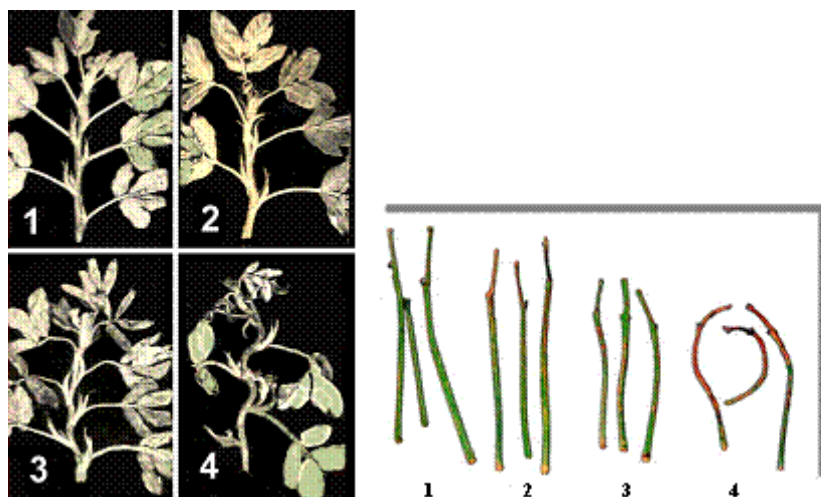
a) Chaves descritivas ou classes de severidade da doença: são escalas arbitrárias com certo número de graus ou notas para quantificar as doenças. A seguir são ilustrados diferentes exemplos de chaves descritivas, simples e mais detalhadas, utilizadas para avaliar algumas doenças foliares da cultura do amendoim.



Escala de notas para a mancha preta do amendoim. (Moraes, 2017).

Onde:

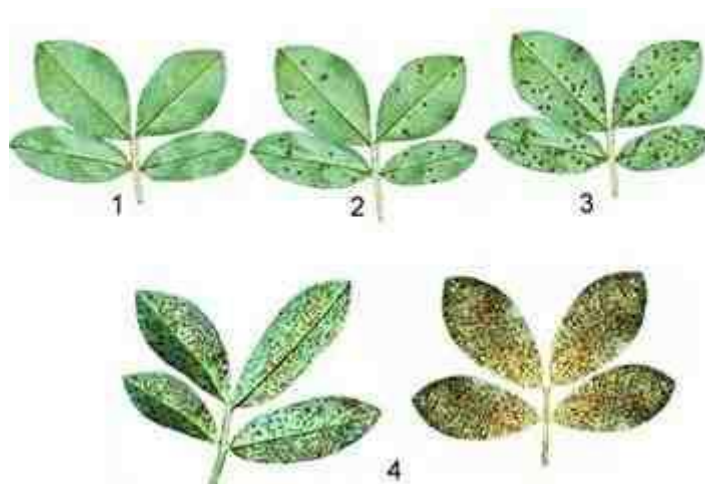
- 1 = Sem mancha.
- 2 = Com pouca doença, ou seja, folíolos com 0,5 a 3,0% de área infectada (1 a 10 manchas/ folíolo).
- 3 = Nível regular de doença, folíolos com 6 a 9% de área infectada (ou 11 a 25 manchas/folíolo).
- 4 = Nível alto de doença, folíolos com mais de 9% de área infectada (mais de 25 manchas/folíolo).



Escalas de notas para avaliação de verrugose em hastes e pecíolos do amendoim (Adaptada de Ribeiro, 1970, em hastes, e elaborada por Moraes, 2017).

Onde:

- 1 = ausência de sintomas de verrugose.
- 2 = baixa severidade - presença de poucas lesões em folhas e pecíolos.
- 3 = severidade regular - lesões de verrugose evidentes na parte apical das plantas, nos pecíolos, nas folhas e nas hastes, que se apresentam moderadamente retorcidos.
- 4 = severidade alta - lesões de verrugose evidentes na planta inteira, com as hastes e os pecíolos da parte apical apresentando-se completamente retorcidos e cobertos pelas lesões.



Notas para a avaliação da ferrugem do amendoim. (Moraes, 2017).

Onde:

- 1 = ausência de ferrugem;
- 2 = baixa severidade (1 a 10 pústulas de ferrugem/folíolo);
- 3 = severidade regular (10-40 pústulas de ferrugem/folíolo);
- 4 = severidade alta (mais de 40 pústulas de ferrugem/folíolo)

Para a determinação da nota média e o índice de doença (variando de 0 a 100%) das doenças da parte aérea (ferrugem, manchas preta e castanha, verrugose e ferrugem), segundo as escalas de notas apresentadas anteriormente, podem ser utilizadas as equações abaixo:

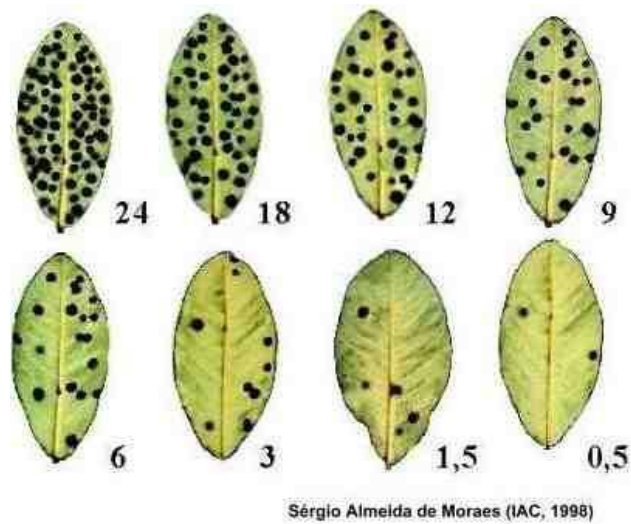
Nota média = $(n1 \times 1 + n2 \times 2 + n3 \times 3 + n4 \times 4) / N$.

Índice de doença (%) = $(n1 \times 0 + n2 \times 25 + n3 \times 50 + n4 \times 100) / N$, onde:

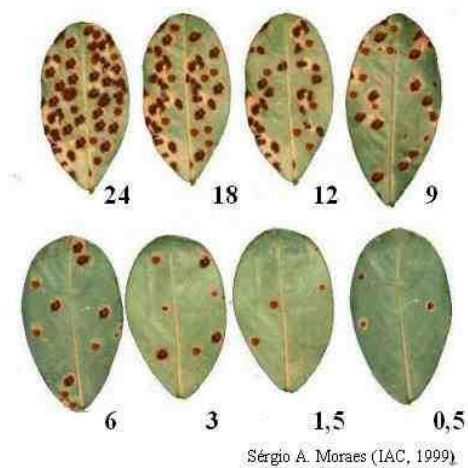
$n1, n2, n3$ e $n4$ = número de folíolos da amostra com as notas 1, 2, 3 e 4;

N = total de folíolos da amostra

b) Escalas diagramáticas: são representações ilustradas de plantas ou partes de plantas (padrões de comparação), mostrando a área necrosada ou coberta pelos sintomas e sinais do patógeno, em diferentes níveis de severidade. Exemplos de escalas diagramáticas utilizadas para avaliar a mancha preta (*Cercosporidium personatum*) e a mancha castanha (*Cercospora arachidicola*) do amendoim são apresentados a seguir.



Escala diagramática para mancha preta do amendoim (% de área infectada). (Moraes, 2017).



Escala diagramática para mancha castanha do amendoim (% de área infectada). (Moraes, 2017).

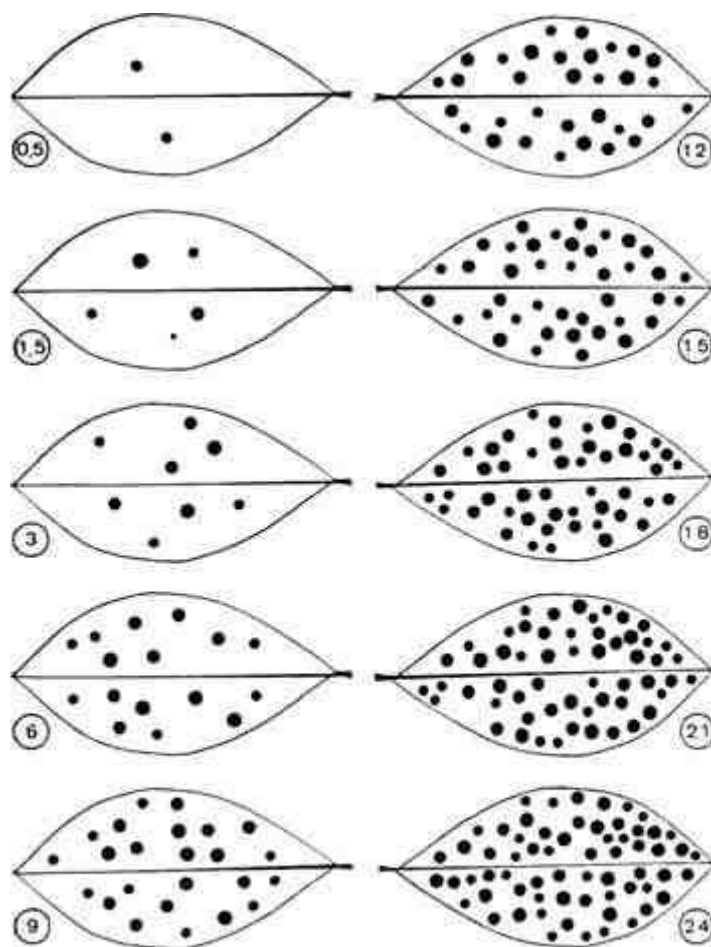


Figura 2. Escala diagramática II: porcentagens de área infectada (1 %)

14

O Agrônomo, Campinas, SP, 39(1), 1987

Escala diagramática para avaliar a intensidade das cercosporioses do amendoim (Moraes, 1987).

1.3. Amostragem

Após a escolha do método de avaliação, a amostragem é uma das etapas fundamentais para que a avaliação das doenças seja representativa da população original, devendo ser feita de maneira criteriosa. Em programas de manejo integrado uma estimativa errada da quantidade de doença, causada pela amostragem incorreta, pode acarretar decisões de controle não adequadas, causando perdas na produção.

O tipo e tamanho da amostra dependem da característica da doença, do objetivo do levantamento (avaliações em parcelas experimentais, manejo integrado das doenças, caracterização do nível de resistência, etc.) e do modelo de dispersão da doença. Devem-se estabelecer previamente alguns critérios, como:

- tipo de amostra (folhas, folíolos, ramos, planta inteira, frutos, sementes, etc.)
- tamanho da amostra (número de folhas, ramos, etc. ou pontos de amostragem)
- local ou pontos de amostragem (coleta representativa da planta, área experimental ou da cultura, marcação de plantas ou ramos).

- época de amostragem – estádios de crescimento da cultura, em função da característica de cada doença.

- número de amostragens durante o ciclo da planta – em função da finalidade, estágio fenológico da planta, curvas de progresso da doença, etc.

2. OBJETIVOS

2.1. Objetivos Gerais

- Gerenciar o processo de obtenção, armazenamento e avaliação de imagens de culturas com sintomas causados por agentes patogênicos.

2.2. Objetivos Específicos

- Gerenciamento de culturas, doenças, amostragem, usuários e imagens.
- Realização de correções necessárias nas imagens para a correta classificação (correção de ruídos).
- Quantificar por métodos diretos de avaliação dos sintomas e sinais, como a incidência, severidade, intensidade.

3. DESCRIÇÃO DO SISTEMA

O sistema é composto pelos módulos de manter, que inclui toda a parte de cadastros, e pelo módulo de gerenciamento, que inclui toda a parte de tratamento de imagens e geração de relatórios.

3.1. Manter

1. **Amostragem:** Tipo da Amostra, tamanho da amostra, local da amostragem, época da amostragem, objetivo do levantamento.
2. **Cultura:** Descrição, tipo de cultura (folha, fruto, grão, entre outros).
3. **Doença:** Descrição da doença, tipo, característica, níveis de doença (escala de notas).
4. **Chaves descritivas ou classes de severidade da doença:** são escalas arbitrárias com certo número de graus ou notas para quantificar as doenças.
5. **Câmera:** como existe um grande número de marcas e modelos, sendo que cada uma delas possuem características distintas, que servem como parâmetros para a realização das diferentes correções necessárias nas imagens, tais como distância focal, resolução, tipo de lente, dentre outros.
6. **Correções:** controle dos tipos de correção de imagem do sistema, sendo possível incluir uma nova correção. Segue os tipos de correções:

6.1. Correção de ruído: aplicação de filtros para remoção e/ou redução de falhas nas imagens.

3.2. Gerenciamento

1. **Armazenamento de imagens:** sistema de arquivos, importação das imagens que serão salvas em um repositório que o sistema irá definir.
2. **Recorte irregular da imagem:** utilizando a biblioteca Opencv será realizado o recorte da área de interesse da imagem.
3. **Correção das imagens:** operação para o qual se recebe uma imagem, o sistema realiza a correção e devolve uma ou mais imagens, corrigidas, além de realizar o backup das imagens inseridas.
4. **Relatórios:** de doenças, número de imagens, espaço utilizado, histórico de utilização, entre outros.

4. ABORDAGEM UTILIZADA PARA APLICAÇÃO DOS PADRÕES DE PROJETO

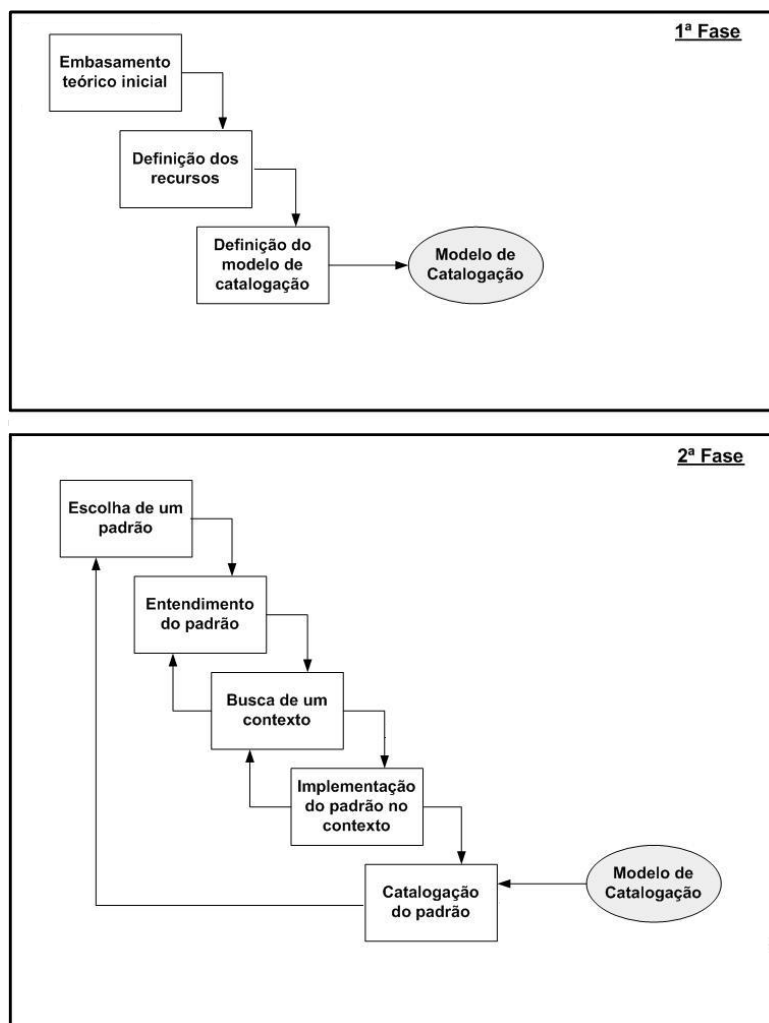
Abordagem utilizada para aplicação do padrão: Adaptação de Matos e Fernandes (2008).

Justificativa da abordagem utilizada: Como os integrantes da equipe não possuem experiência prévia com desenvolvimento de sistemas utilizando padrões de projetos, avaliou-se que o melhor seria a utilização da abordagem descrita acima, pois no que os alunos vão obtendo o embasamento teórico e o entendimento de cada padrão, estes vão implementando o padrão no contexto da aplicação.

Adaptações na abordagem: Na primeira fase foi retirada a etapa “Divisão dos Padrões por Aluno”, pois como abordagem proposta preocupa-se com a divisão de tarefas de uma equipe, nesta atividade são estabelecidas às responsabilidades de cada integrante. Os membros da equipe estudaram todos os padrões de maneira igualitária, catalogados segundo o esquema de relacionamento proposto por Gamma et al. (1994), conciliando uma didática mais dinâmica com o emprego bem-sucedido daquilo que será aprendido.

Já na segunda fase foi retirada a etapa “Reunião dos alunos”, pois compõe uma atividade que foi realizada através de conversas semanais entre os integrantes da equipe, não seguindo, portanto a ordem cronológica descrita por Matos e Fernandes (2008). , Pelo mesmo motivo foi retirada também a etapa “Reunião dos alunos com o responsável” que foi realizada algumas vezes com a professora responsável pela disciplina.

O restante é similar ao descrito por Matos e Fernandes (2008).



Adaptado de: Matos e Fernandes (2008).

4.1. Primeira fase

| Atividades | Descrição |
|-----------------------------|--|
| Embasamento Teórico Inicial | Padrão de projeto é um tema bastante encontrado em publicações da literatura especializada. Entretanto, pouco se fala na parte de aprendizado e assimilação dos padrões antes de optar pela sua adoção. Assim sendo, nesta atividade se realiza a busca de material teórico sobre o assunto, tais como, livros, guias de desenvolvimento e tutoriais. |
| Definição dos Recursos | Na criação de uma aplicação ou sistema, a fim de abordar um dado contexto, deve-se escolher os recursos, tais como, linguagem de programação, ferramenta e linguagem de modelagem, ambiente de desenvolvimento e persistência dos dados. Portanto, nesta atividade se identifica os recursos necessários para o desenvolvimento de uma dada aplicação. |

| | |
|------------------------------------|--|
| Definição do Modelo de Catalogação | Não existe um consenso sobre como descrever um padrão: diferentes autores preferem diferentes estilos. Por esse motivo, nesta atividade busca-se selecionar da literatura especializada um modelo ideal para catalogação. A saída dessa atividade é o “Modelo de Catalogação” a ser utilizado pela equipe. |
|------------------------------------|--|

4.2. Segunda fase

| Atividades | Descrição |
|-------------------------------------|---|
| Escolha de um padrão | Cada participante deverá, nesta atividade, escolher um padrão para compreensão e implementação. Os padrões que serão implementados por cada integrante da equipe já foram definidos na primeira fase por meio da atividade “Divisão dos Padrões por Aluno”. |
| Entendimento dos padrões de projeto | Cada participante deve procurar material específico, principalmente exemplos práticos, sobre o padrão escolhido anteriormente, com o objetivo de compreender seu funcionamento para implementá-lo numa atividade posterior. |
| Busca de um contexto | Delimitar um contexto prático simples, no qual o padrão escolhido pode ser implementado. |
| Implementação do padrão no contexto | Implementação do padrão usando os recursos definidos na primeira fase por meio da atividade “Definição dos Recursos”. |
| Catalogação do padrão | A partir da atividade “Definição do Modelo de Catalogação”, primeira fase, o padrão será documentado. Por isso, possui como entrada o “Modelo de Catalogação”. |

5. APLICAÇÃO DA ABORDAGEM

Para a aplicação da abordagem seguiu-se o roteiro descrito pela seção 4. A ordem das iterações respeitou a sequência de aprendizagem dos integrantes da equipe. Assim, como descrito por Gamma et al. (1994), a ordem de aprendizagem dos padrões seguiu pelas categorias em que estes foram divididos: Criacionais, Estruturais e Comportamentais. Apenas o padrão *Data Access Object* (DAO) não faz parte da catalogação proposta por Gamma et al. (1994).

5.1. Primeira fase

| Atividades | Descrição |
|------------------------------------|---|
| Embasamento Teórico Inicial | Através das aulas e da literatura passada pela professora, os alunos obtiveram o embasamento teórico para cada Padrão de Projeto. |
| Definição dos Recursos | Linguagem de programação: Java (JDK 8.0), em conjunto com a biblioteca OpenCV. Linguagem de modelagem: UML (<i>Unified Modeling Language</i>). Banco de Dados: PostgreSQL. 9.6. Ambiente desenvolvimento: Netbeans 8.1. Ambiente de modelagem: Astah Professional 7.2. Tipo de aplicação: Desktop. |
| Definição do Modelo de Catalogação | Nesta atividade busca-se selecionar da literatura especializada um modelo ideal para catalogação. Assim, selecionou-se o modelo de catalogação proposto por Gamma et al. (1994). |

5.2. Segunda fase

A segunda etapa será descrita com mais detalhes. Para cada padrão catalogado, foi realizada cada uma das atividades descritas por Matos e Fernandes (2008).

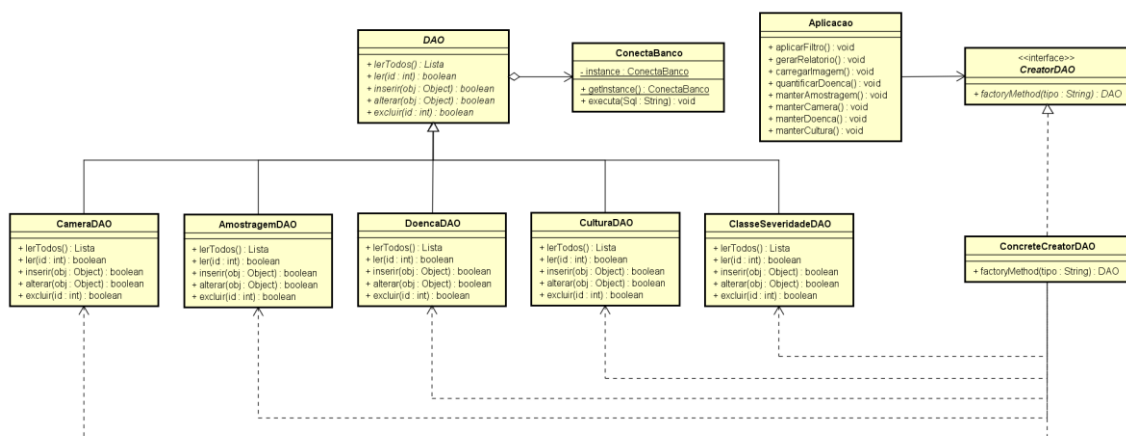
5.2.1. Persistência de Dados

1ª Iteração

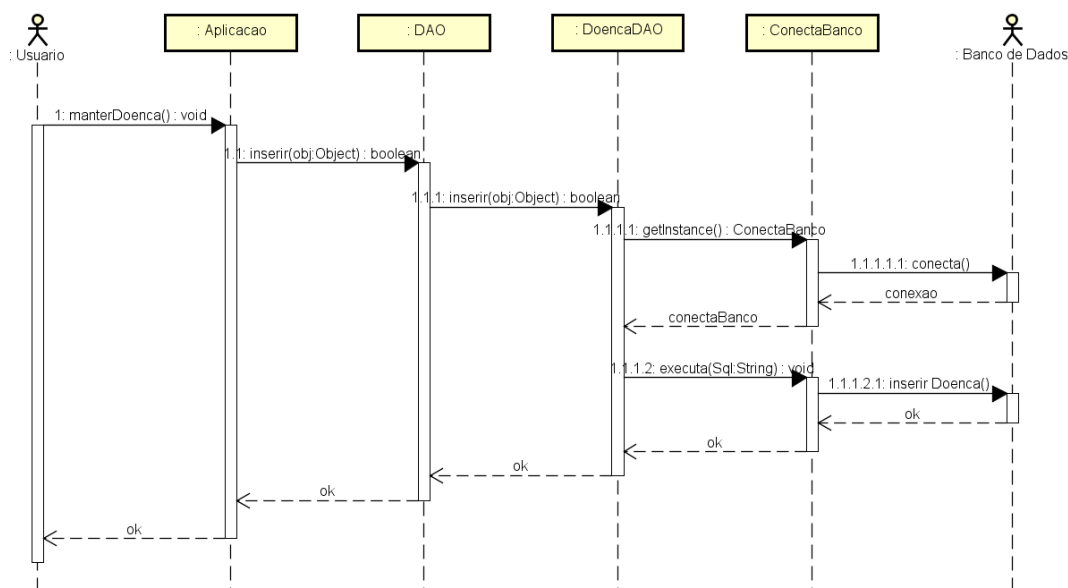
| Atividades | Descrição |
|-------------------------------------|--|
| Padrão escolhido | <i>Data Access Object</i> (DAO) |
| Entendimento do padrão | Através de material disponibilizado por Macoratti (2010). |
| Busca de um contexto de aplicação | Através dos exemplos mostrados por Macoratti (2010). |
| Implementação do padrão no contexto | Como o sistema necessita persistir os dados em um Banco de Dados relacional, verificou-se a possibilidade de incluir o padrão DAO para tratar de fazer o ORM (Mapeamento Objeto Relacional). |
| Catalogação do padrão | Padrão é documentado no diagrama a seguir. |

A classe abstrata DAO provê as operações para leitura, inserção, alteração e exclusão dos dados, além de se comunicar com a classe de conexão com o Banco de Dados. Assim, as classes que herdam da classe DAO irão realizar toda parte de comunicação com o Banco de Dados, fazendo com que a lógica de regra de negócio fique

desacoplada da lógica de persistência dos dados. Portanto, as classes DAO recebem e devolvem um objeto de uma determinada entidade, ficando a caráter de realizar o mapeamento objeto relacional.



O diagrama de sequência abaixo ilustra a troca de mensagens entre as classes envolvidas para inserção de uma entidade (doença) no Banco de Dados. Para as outras entidades e operações suportadas pelo sistema o processo é similar, não necessitando ser ilustrado por outros diagramas de sequência.



5.2.2. Criacionais

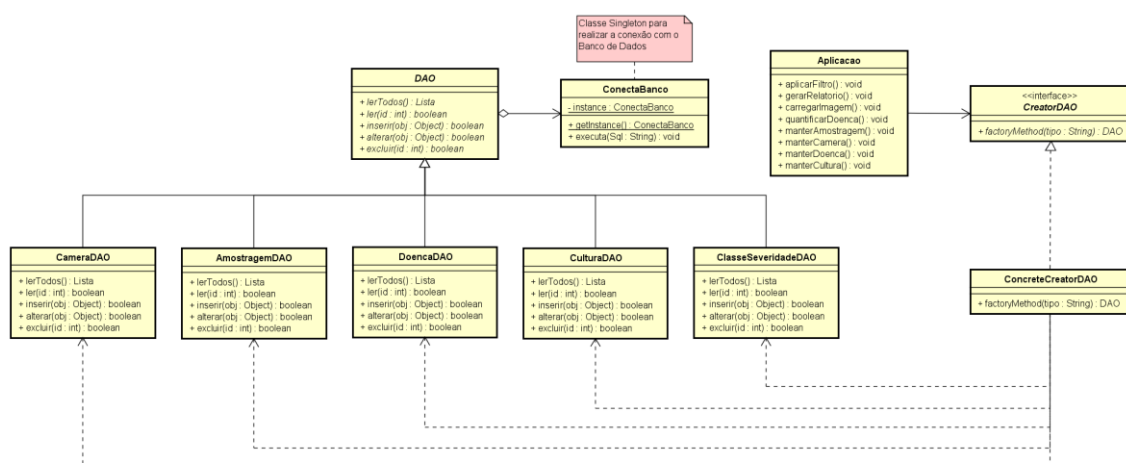
2ª Iteração

| Atividades | Descrição |
|-----------------------------------|--|
| Padrão escolhido | Singleton |
| Entendimento do padrão | Através de material disponibilizado por Matos (2017a). |
| Busca de um contexto de aplicação | Através de material disponibilizado por |

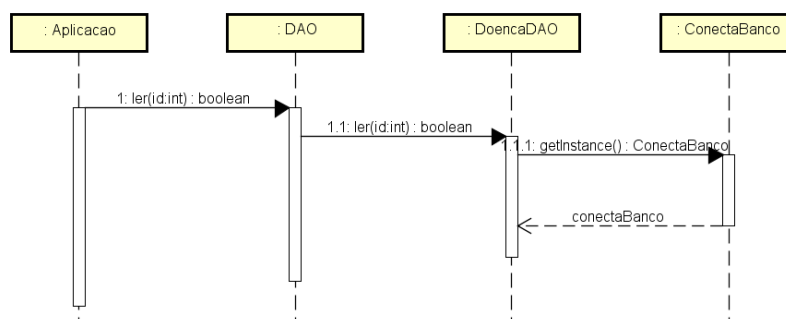
| | |
|-------------------------------------|---|
| | Matos (2017a). |
| Implementação do padrão no contexto | Na classe de conexão com o Banco de Dados e em todas as telas do sistema. |
| Catologação do padrão | Padrão é documentado no diagrama a seguir. |

Todas as telas da aplicação utilizam o padrão Singleton. Assim, é apenas permitido ser aberto uma tela de cada tipo por vez. Além disso, a classe de conexão com o Banco de Dados também aplica este padrão, ou seja, isso garante que haja apenas uma conexão por vez com o Banco, evitando que sejam abertas múltiplas conexões desnecessárias que podem prejudicar o desempenho do SGBD.

Pode-se verificar que a classe ConectaBanco possui um atributo estático que retorna um objeto da própria classe, além de uma operação também estática que irá verificar se este objeto já foi instanciado alguma vez na execução do programa. O modificador estático permite que essa operação seja acessada por outras classes, sem a necessidade destas terem que instanciar o objeto da classe ConectaBanco. Por fim, essa classe possui operações necessárias para realizar as consultas SQL com o Banco de Dados.



O diagrama de sequência abaixo ilustra as trocas de mensagens envolvidas entre duas classes participantes a fim de obter uma instância da conexão.

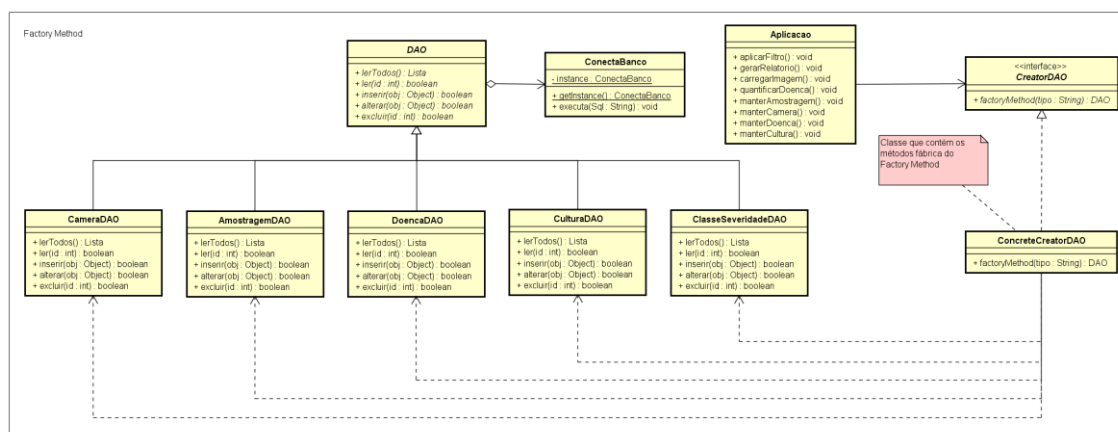


3ª Iteração

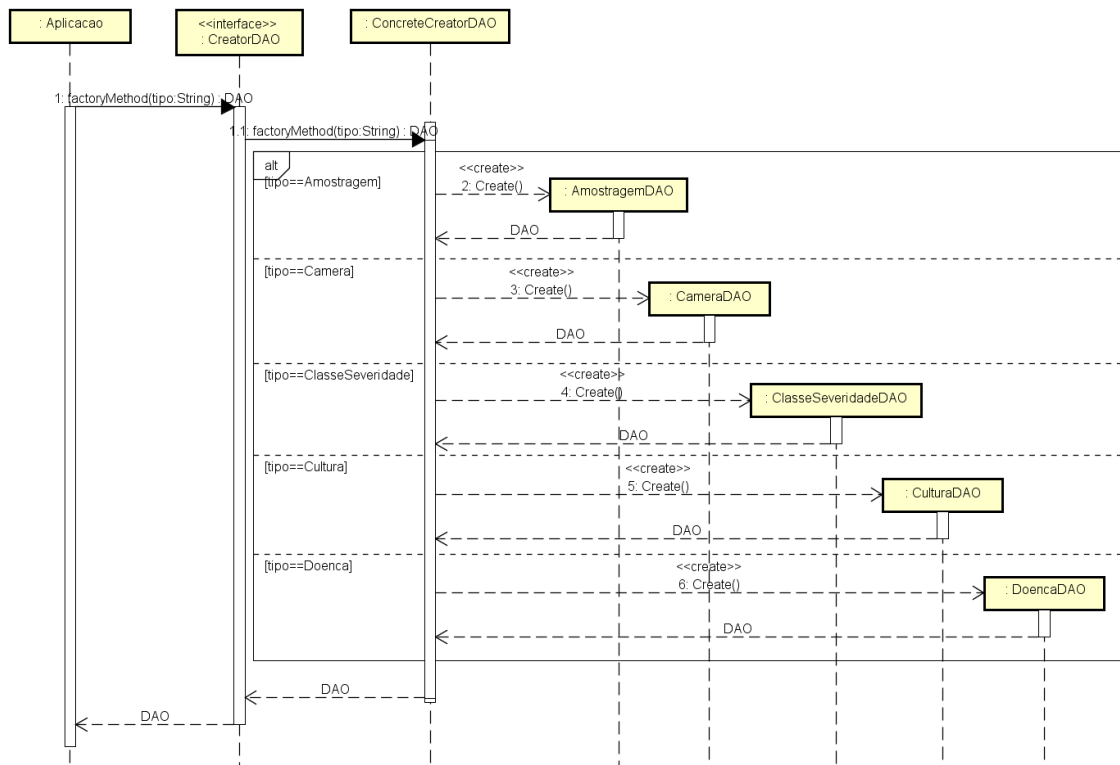
| Atividades | Descrição |
|------------|-----------|
|------------|-----------|

| | |
|-------------------------------------|---|
| Padrão escolhido | Factory Method |
| Entendimento do padrão | Através de material disponibilizado por Matos (2017c). |
| Busca de um contexto de aplicação | Através de material disponibilizado por Matos (2017c). |
| Implementação do padrão no contexto | Implementou-se o método fábrica para facilitar a manutenção de classes DAO e de classes de Quantificação. |
| Catologação do padrão | Padrão é documentado no diagrama a seguir. |

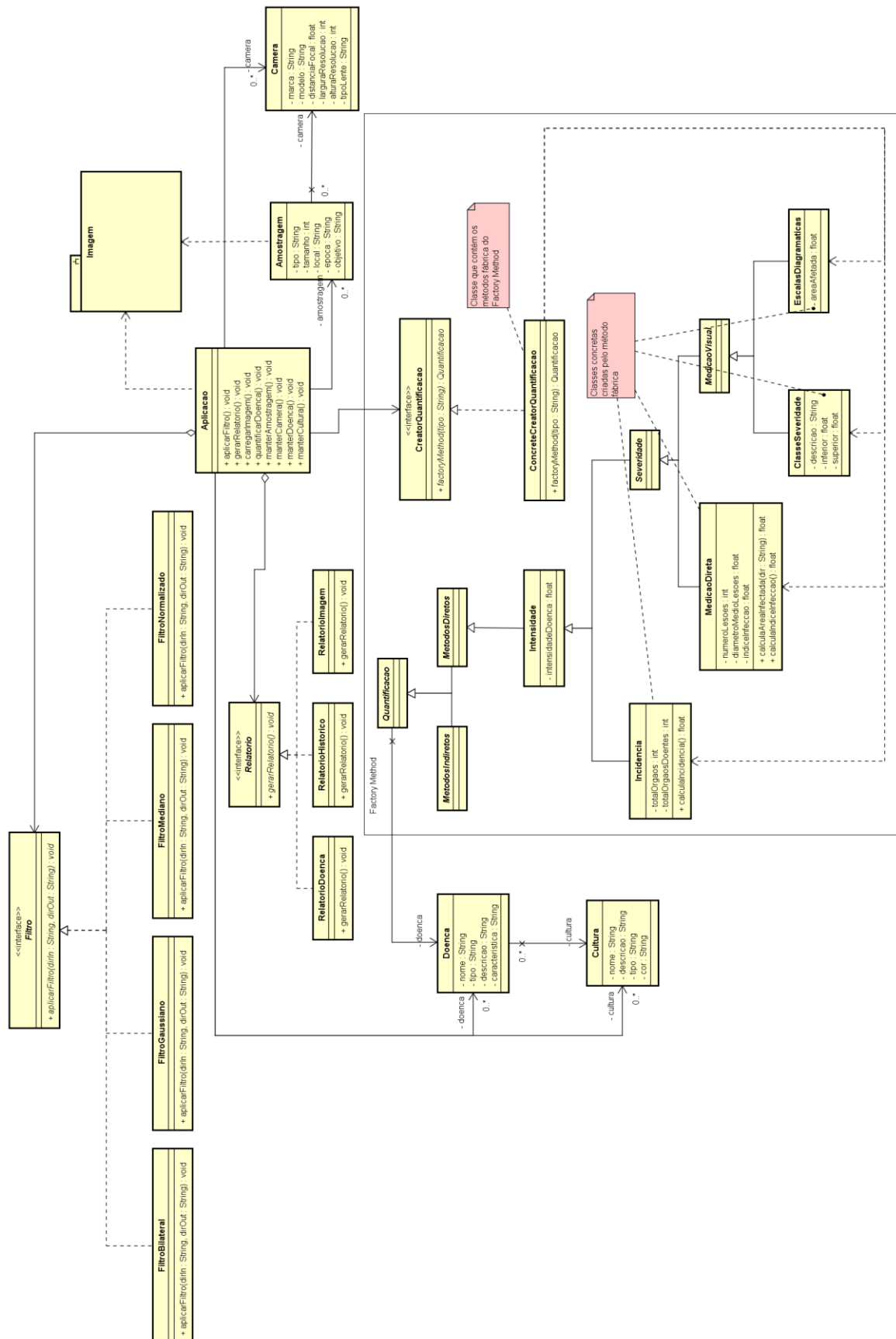
O padrão Factory Method foi aplicado juntamente com o padrão DAO, facilitando a manutenção no caso de haver a necessidade de expansão da aplicação com a inserção de novas classes para persistência. No diagrama de classes, pode-se verificar a classe CreatorDAO que retorna um objeto do tipo DAO, a classe ConcreteCreatorDAO que sobrepõe o factory method para retornar qualquer umas das instâncias concretas de DAO (DoencaDAO, CulturaDAO, entre outras). A classe DAO no padrão Factory Method é equivalente ao Product, e todas as classes que herdaram de DAO são as classes ConcreteProduct do padrão.



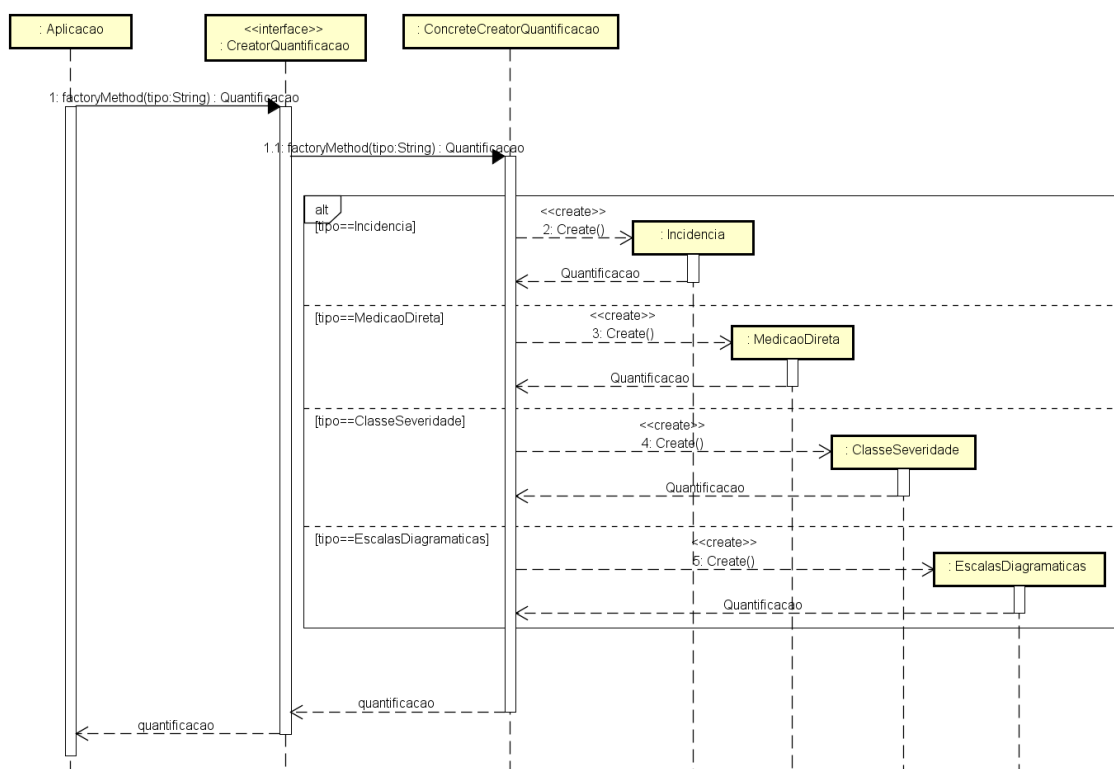
O diagrama de sequência mostra a sequência de operações necessárias para a criação das classes concretas de DAO.



O padrão Factory Method foi também aplicado para facilitar o reuso das classes que realizam a Quantificacao de doenças da aplicação. No diagrama de classes, existe a classe `CreatorQuantificacao` que retorna um objeto do tipo `Quantificacao` (Product), que por sua vez é uma interface de objetos que o factory method cria. A classe `ConcreteCreatorQuantificacao` herda de `Creator` e sobrepõe o factory method, permitindo retornar qualquer classe concreta (`ConcreteProduct`) que herde da classe `Product`.



O diagrama de sequência mostra a sequência de operações necessárias para a criação das classes concretas de **Quantificacao**.



4ª Iteração

| Atividades | Descrição |
|-------------------------------------|--|
| Padrão escolhido | Abstract Factory |
| Entendimento do padrão | Através de material disponibilizado por Matos (2017c). |
| Busca de um contexto de aplicação | Através de material disponibilizado por Matos (2017c). |
| Implementação do padrão no contexto | Como o sistema não possui nenhuma família de produtos relacionados, não é possível a aplicação deste padrão. |
| Catologação do padrão | - |

5ª Iteração

| Atividades | Descrição |
|-------------------------------------|---|
| Padrão escolhido | Prototype |
| Entendimento do padrão | Através de material disponibilizado por Matos (2017c). |
| Busca de um contexto de aplicação | Através de material disponibilizado por Matos (2017c). |
| Implementação do padrão no contexto | Como não existem carregamentos dinâmicos de classes e as todas as classes |

| | |
|--------------------|--|
| | do sistema possuem grandes combinações de estado, não é possível haver a aplicação deste padrão. |
| Catálogo do padrão | - |

6ª Iteração

| Atividades | Descrição |
|-------------------------------------|---|
| Padrão escolhido | Builder |
| Entendimento do padrão | Através de material disponibilizado por Brizen (2011a). |
| Busca de um contexto de aplicação | Através de material disponibilizado por Brizen (2011a). |
| Implementação do padrão no contexto | Não é possível separar a construção dos objetos do sistema em pequenos passos, assim o Builder não é aplicável. |
| Catálogo do padrão | - |

5.2.3. Estruturais

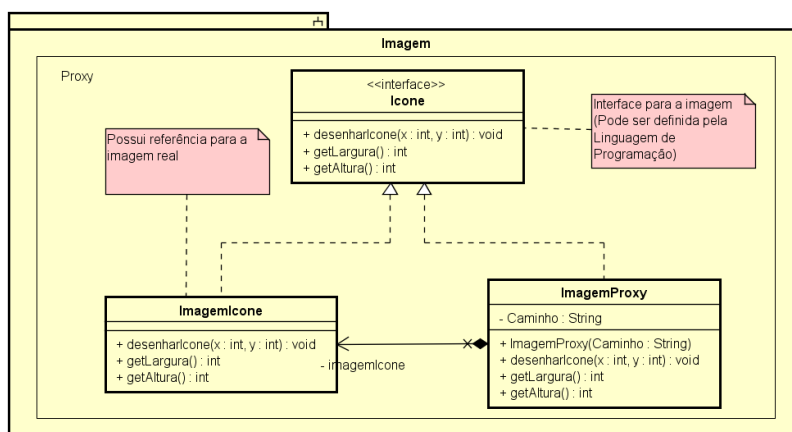
7ª Iteração

| Atividades | Descrição |
|-------------------------------------|---|
| Padrão escolhido | Adapter |
| Entendimento do padrão | Através de material disponibilizado por Matos (2017b). |
| Busca de um contexto de aplicação | Através de material disponibilizado por Matos (2017b). |
| Implementação do padrão no contexto | Não há necessidade de adaptação de interface para qualquer classe existente, assim o Adapter não é aplicável. |
| Catálogo do padrão | - |

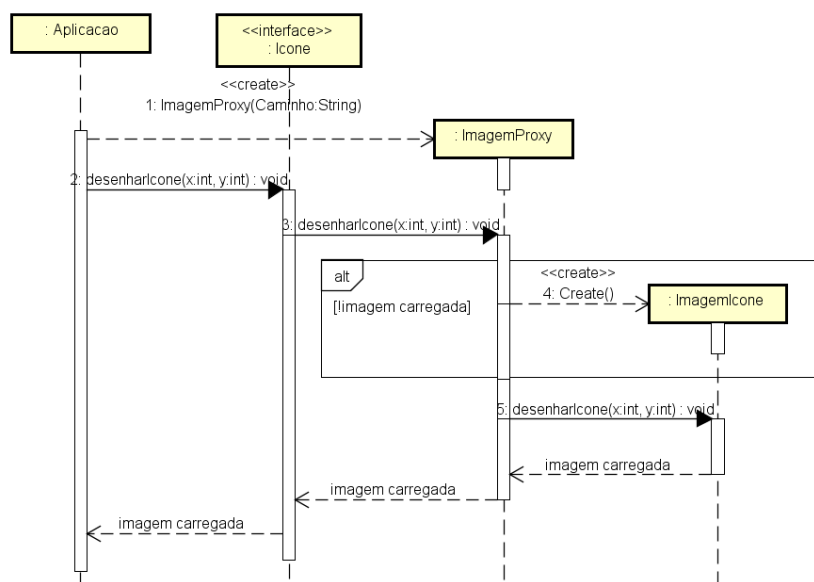
8ª Iteração

| Atividades | Descrição |
|-------------------------------------|--|
| Padrão escolhido | Proxy |
| Entendimento do padrão | Através de material disponibilizado por Beluzzo (2017f). |
| Busca de um contexto de aplicação | Através de material disponibilizado por Beluzzo (2017f). |
| Implementação do padrão no contexto | Como a sistema trata de imagens, verificou-se a possibilidade de implementar o padrão Proxy (Virtual Proxy) para evitar o recarregamento desnecessário de imagens previamente carregadas no sistema. |
| Catálogo do padrão | Padrão é documentado no diagrama a seguir. |

O padrão de projeto Proxy, mais precisamente o Virtual Proxy foi aplicado no subsistema Imagem. Assim, as imagens previamente carregadas no sistema não precisarão ser recarregadas novamente. A estrutura é ilustrada no diagrama de classes a seguir, onde a interface *Icone* é o Subject do padrão, que possui as operações que serão implementadas pelas classes *ImagemProxy* (Proxy) e *ImagemIcone* (Real). A classe *ImagemProxy* é um substituto que controla o acesso a *ImagemIcone*, sendo assim ela que gerencia a criação de novos objetos da classe *ImagemIcone*.



O diagrama de sequência mostra a troca de mensagens através de chamadas de operações entre as classes envolvidas, no contexto de carregar uma imagem na tela da aplicação.



9ª Iteração

| Atividades | Descrição |
|------------------|-----------|
| Padrão escolhido | Composite |

| | |
|-------------------------------------|---|
| Entendimento do padrão | Através de material disponibilizado por Beluzzo (2017a). |
| Busca de um contexto de aplicação | Através de material disponibilizado por Beluzzo (2017a). |
| Implementação do padrão no contexto | Não existem objetos com estruturas de árvore e nem a necessidade de tratá-los de maneira uniforme. Portanto, não é aconselhável a aplicação do Composite. |
| Catologação do padrão | - |

10ª Iteração

| Atividades | Descrição |
|-------------------------------------|--|
| Padrão escolhido | Flyweight |
| Entendimento do padrão | Através de material disponibilizado por Beluzzo (2017c). |
| Busca de um contexto de aplicação | Através de material disponibilizado por Beluzzo (2017c). |
| Implementação do padrão no contexto | Os custos de armazenamento para a aplicação não é tão elevado, pois não há uma quantidade tão alta de objetos. A grande maioria dos estados desses objetos não pode ser tomada de maneira extrínseca. Desta forma, este padrão não é aplicável nessa situação. |
| Catologação do padrão | - |

11ª Iteração

| Atividades | Descrição |
|-------------------------------------|---|
| Padrão escolhido | Decorator |
| Entendimento do padrão | Através de material disponibilizado por Matos (2017d). |
| Busca de um contexto de aplicação | Através de material disponibilizado por Matos (2017d). |
| Implementação do padrão no contexto | Em nenhuma classe a extensão por subclasseamento foi impraticável. Também, a aplicação não exige adicionar responsabilidades a objetos individuais dinamicamente e de forma transparente. Portanto, não há a necessidade de aplicação do Decorator. |
| Catologação do padrão | - |

12ª Iteração

| Atividades | Descrição |
|------------------|-----------|
| Padrão escolhido | Facade |

| | |
|-------------------------------------|--|
| Entendimento do padrão | Através de material disponibilizado por Brizeno (2011b). |
| Busca de um contexto de aplicação | Através de material disponibilizado por Brizeno (2011b). |
| Implementação do padrão no contexto | Como a aplicação não possui nenhum subsistema complexo que necessite de uma fachada para facilitar a sua utilização por classes externas, julgou-se que seria inadequado aplicar o Facade. |
| Catálogo do padrão | - |

5.2.4. Comportamentais

13ª Iteração

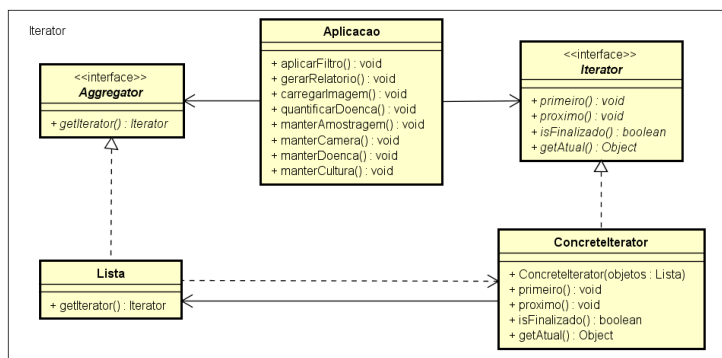
| Atividades | Descrição |
|-------------------------------------|---|
| Padrão escolhido | State |
| Entendimento do padrão | Através de material disponibilizado por Beluzzo (2017d). |
| Busca de um contexto de aplicação | Através de material disponibilizado por Beluzzo (2017d). |
| Implementação do padrão no contexto | Não existem objetos que tenham comportamentos dependentes de seus estados, em tempo de execução. Neste contexto não é adequado aplicar este padrão. |
| Catálogo do padrão | - |

14ª Iteração

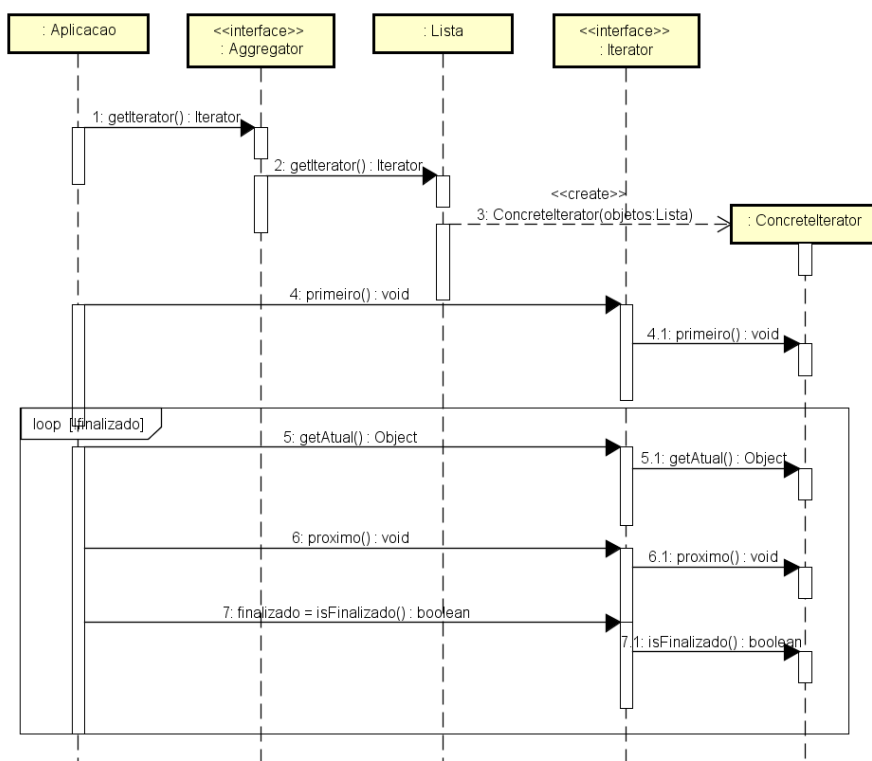
| Atividades | Descrição |
|-------------------------------------|--|
| Padrão escolhido | Iterator |
| Entendimento do padrão | Através de material disponibilizado por Beluzzo (2017g). |
| Busca de um contexto de aplicação | Através de material disponibilizado por Beluzzo (2017g). |
| Implementação do padrão no contexto | Aplicou-se o padrão Iterator a fim de percorrer a listagem das entidades gravadas no Banco de Dados (Amostragem, Cultura, Doença, Classe de Severidade e Câmera) |
| Catálogo do padrão | Padrão é documentado no diagrama a seguir. |

A estrutura do padrão Iterator é ilustrado no diagrama de classes a seguir. A fim de ser mais reusável e aplicável a todas as entidades do sistema, procurou-se construí-la da forma mais genérica possível, com o iterador retornando um objeto da classe Object. A estrutura das classes é composta pela interface Iterator que declara as operações

realizadas pelo ConcreteIterator, a fim de percorrer a coleção de objetos. A classe Agregator declara uma operação implementada pela classe Lista (ConcreteAgregator), que retorna um objeto concreto do tipo Iterator, que será chamado no laço da Aplicação.



A sequência de chamadas realizadas entre as classes do padrão Iterator é mostrada no diagrama de sequência.



15ª Iteração

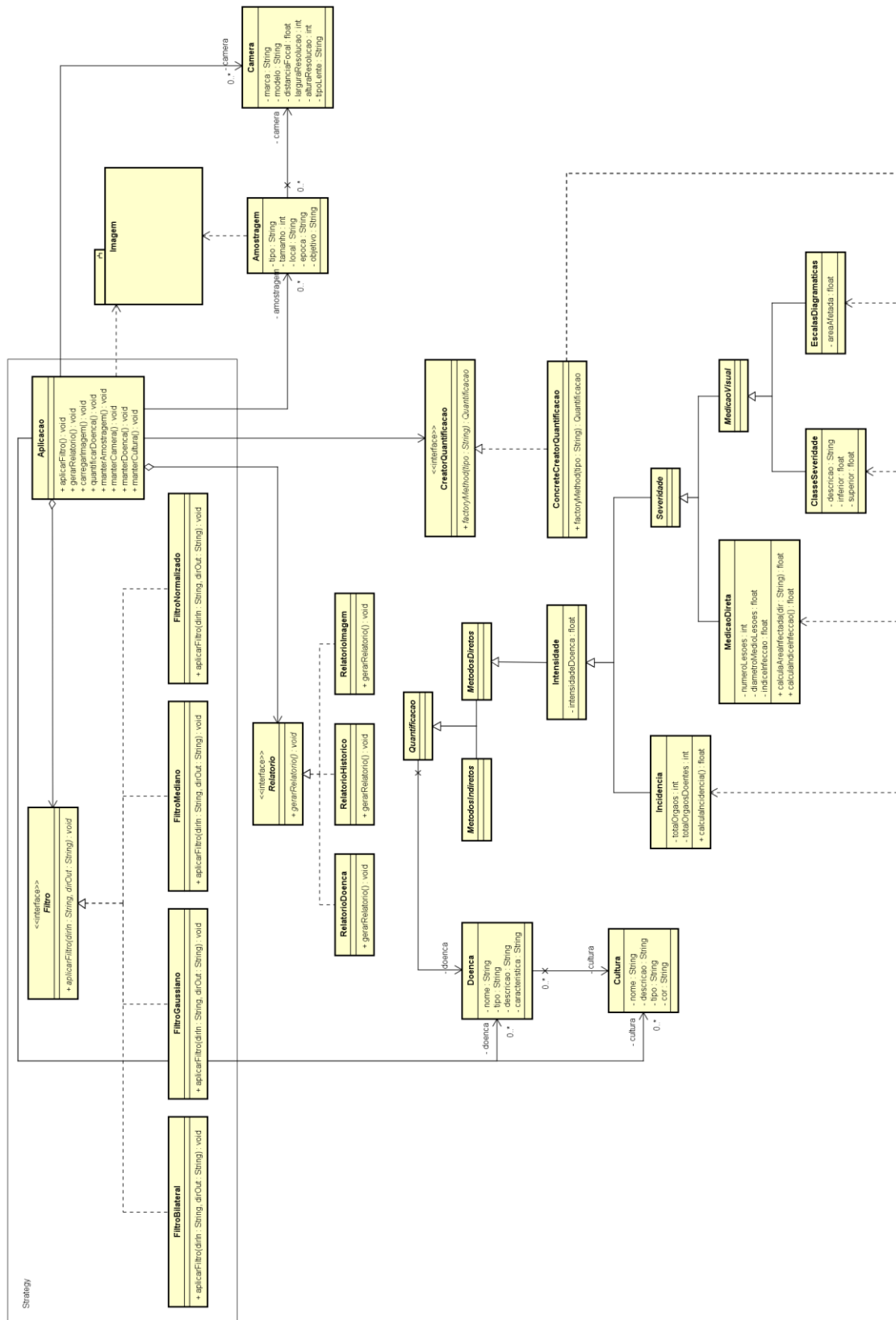
| Atividades | Descrição |
|-----------------------------------|--|
| Padrão escolhido | Observer |
| Entendimento do padrão | Através de material disponibilizado por Matos (2017e). |
| Busca de um contexto de aplicação | Através de material disponibilizado por Matos (2017e). |

| | |
|-------------------------------------|---|
| Implementação do padrão no contexto | Não existem contextos no sistema em que a alteração de um objeto necessite que haja a mudança de outros objetos observadores. Portanto, o padrão Observer não pode ser implantado na modelagem. |
| Catologação do padrão | - |

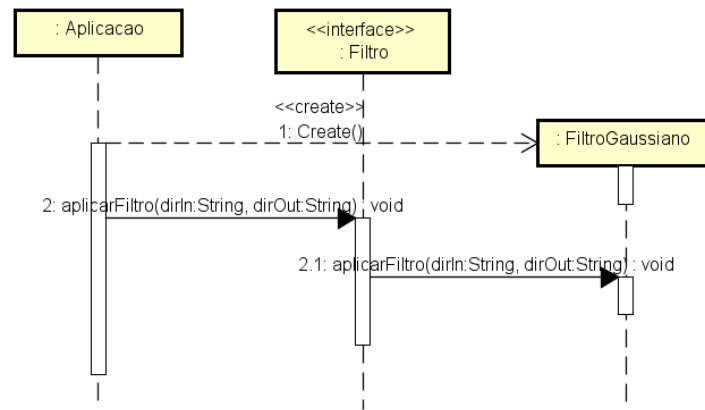
16ª Iteração

| Atividades | Descrição |
|-------------------------------------|---|
| Padrão escolhido | Strategy |
| Entendimento do padrão | Através de material disponibilizado por Matos (2017e). |
| Busca de um contexto de aplicação | Através de material disponibilizado por Matos (2017e). |
| Implementação do padrão no contexto | Como as classes para geração de relatórios e aplicação de filtros possuem varias formas de implementação, diferenciando-se pela forma de execução dos algoritmos, o padrão Strategy encaixa-se perfeitamente nesse contexto de aplicação. |
| Catologação do padrão | Padrão é documentado no diagrama a seguir. |

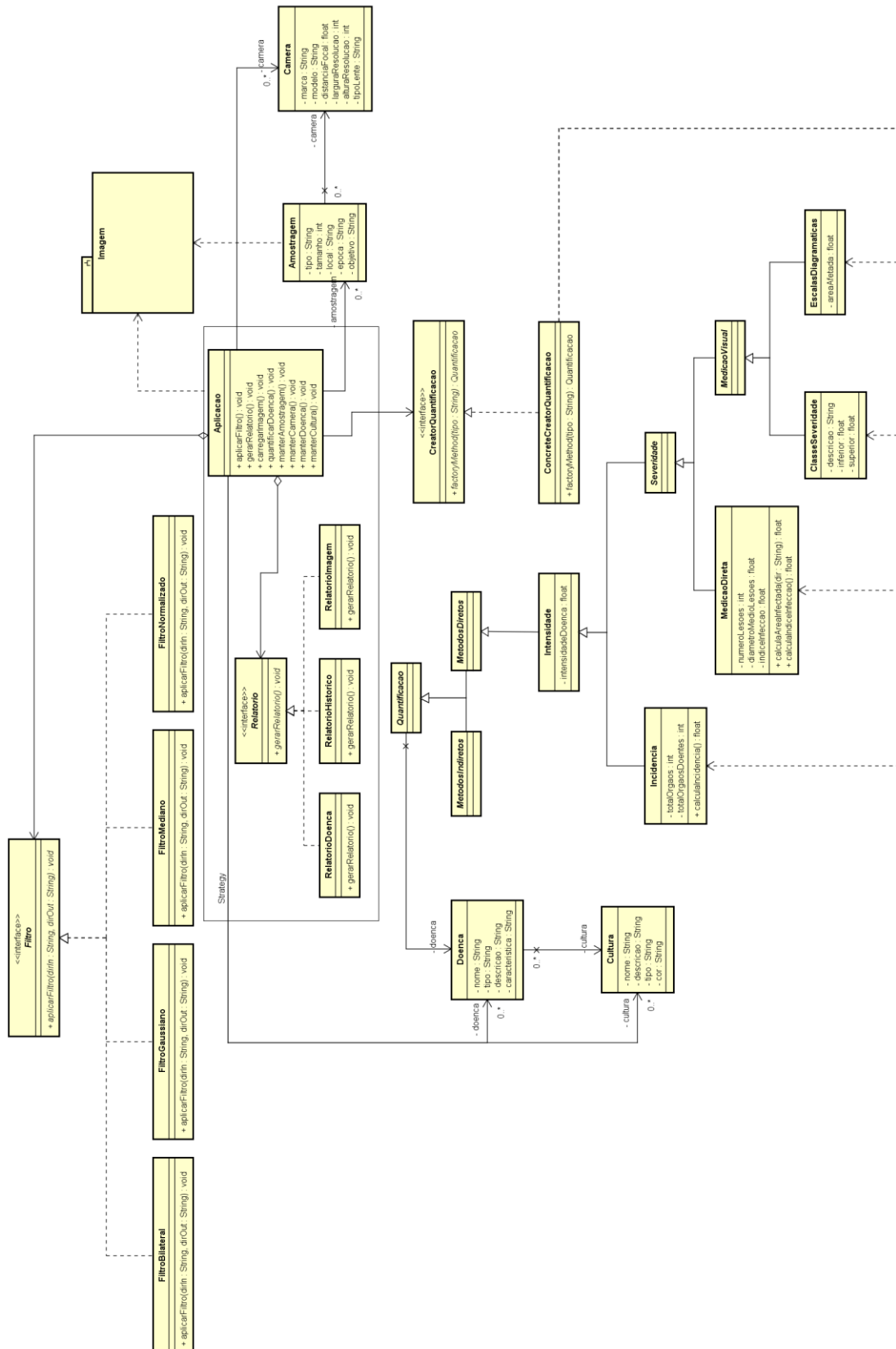
O padrão Strategy foi implementado nas classes de aplicação de filtros em imagens. No diagrama de classes são destacadas as classes que fazem parte da estrutura deste padrão. A classe Filtro (Strategy) declara uma interface comum para todos os algoritmos de filtros suportados na aplicação. No diagrama é visto a facilidade que este tipo de estrutura suporta para a adição de novos tipos de filtros, simplesmente criando classes que implementam a interface Filtro. O Context (classe Aplicação) usa esta interface para chamar os algoritmos definidos pelas classes concretas de Filtro (FiltroBilateral, FiltroGaussiano, FiltroMediano, FiltroNormalizado).



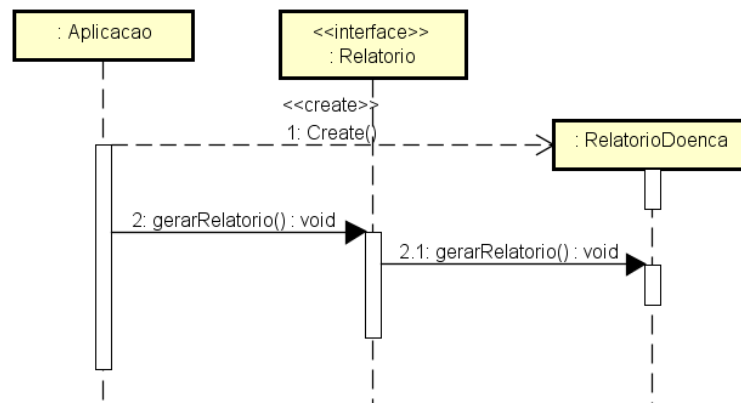
O diagrama de sequência ilustra as trocas de mensagens efetuadas pelos participantes do padrão Strategy.



O padrão Strategy também foi implementado nas classes de geração de relatórios. No diagrama de classes a seguir são destacadas as classes que fazem parte da estrutura deste padrão. A classe Relatório (Strategy) declara uma interface comum para todos os algoritmos de criação de relatórios suportados na aplicação. No diagrama é visto a facilidade que este tipo de estrutura suporta para a adição de novos tipos de relatórios, simplesmente criando classes que implementam a interface Relatório. O Context (classe Aplicação) usa esta interface para chamar os algoritmos definidos pelas classes concretas de Relatório (RelatorioDoenca, RelatorioHistorico, RelatorioImagem).



O diagrama de sequência ilustra as trocas de mensagens efetuadas pelos participantes do padrão Strategy.



17ª Iteração

| Atividades | Descrição |
|-------------------------------------|---|
| Padrão escolhido | Command |
| Entendimento do padrão | Através de material disponibilizado por Beluzzo (2017b). |
| Busca de um contexto de aplicação | Através de material disponibilizado por Beluzzo (2017b). |
| Implementação do padrão no contexto | O sistema não possui clientes com diferentes solicitações, portanto, não necessita da realização do registro (log) ou da criação de filas de prioridade. Assim, este padrão de projeto não pode ser aplicado. |
| Catologação do padrão | - |

18ª Iteração

| Atividades | Descrição |
|-------------------------------------|--|
| Padrão escolhido | Memento |
| Entendimento do padrão | Através de material disponibilizado por Beluzzo (2017e). |
| Busca de um contexto de aplicação | Através de material disponibilizado por Beluzzo (2017e). |
| Implementação do padrão no contexto | A aplicação não permite a restauração do estado de nenhum objeto. Portanto, esse padrão não pode ser aplicado. |
| Catologação do padrão | - |

5.3. Vantagens e Desvantagens

(Opinião pessoal)

6. CONCLUSÃO

O que aprendi com o trabalho.

7. REFERÊNCIAS

BELUZZO, L. B. **PADRÕES DE PROJETO - COMPOSITE**. 2017a. 10 slides. Material apresentado para a disciplina de Modelagem de Sistemas Agrícolas da UEPG.

BELUZZO, L. B. **PADRÕES DE PROJETO - COMMAND**. 2017b. 10 slides. Material apresentado para a disciplina de Modelagem de Sistemas Agrícolas da UEPG.

BELUZZO, L. B. **PADRÕES DE PROJETO - FLYWEIGHT**. 2017c. 11 slides. Material apresentado para a disciplina de Modelagem de Sistemas Agrícolas da UEPG.

BELUZZO, L. B. **PADRÕES DE PROJETO - ITERATOR**. 2017d. 8 slides. Material apresentado para a disciplina de Modelagem de Sistemas Agrícolas da UEPG.

BELUZZO, L. B. **PADRÕES DE PROJETO - MEMENTO**. 2017e. 8 slides. Material apresentado para a disciplina de Modelagem de Sistemas Agrícolas da UEPG.

BELUZZO, L. B. **PADRÕES DE PROJETO - PROXY**. 2017f. 18 slides. Material apresentado para a disciplina de Modelagem de Sistemas Agrícolas da UEPG.

BELUZZO, L. B. **PADRÕES DE PROJETO - STATE**. 2017g. 8 slides. Material apresentado para a disciplina de Modelagem de Sistemas Agrícolas da UEPG.

BRIZENO, M. **Builder**. 2011a. Artigo em Hypertexto. Disponível em: <<https://brizeno.wordpress.com/category/padroes-de-projeto/builder>>. Acesso em: 22/11/2017.

BRIZENO, M. **Facade**. 2011b. Artigo em Hypertexto. Disponível em: <<https://brizeno.wordpress.com/category/padroes-de-projeto/facade>>. Acesso em: 22/11/2017.

GAMMA, Erich & HELM, Richard & JOHNSON, Ralph & VLISSIDES, John. **Design Patterns: Elements of Reusable Object-Oriented Software**. Reading, MA: Addison-Wesley, 1994.

JAMES, W.C. **A manual of assessment keys of plant diseases**. Canada Department of Agriculture Publication, No 1458, 1971, 74p.

MACORATTI, J. C. **Apresentando o padrão DAO - Data Access Object**. 2010. Artigo em Hypertexto. Disponível em: <http://www.macoratti.net/11/10/pp_dao1.htm>. Acesso em: 22/11/2017.

MATOS, S. N.; FERNANDES, C. T. **Abordagem Metodológica para Compreensão e Aplicação de Padrões de Projeto**. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO, 16., 2008, Belém do Pará. Anais do XXVIII Congresso da SBC. Belém do Pará: SBC, 2008.

MATOS, S. N. **INTRODUÇÃO À PADRÕES DE PROJETO**. 2017a. 35 slides. Material apresentado para a disciplina de Modelagem de Sistemas Agrícolas da UEPG.

MATOS, S. N. **PADRÕES DE PROJETO - ADAPTER**. 2017b. 12 slides. Material apresentado para a disciplina de Modelagem de Sistemas Agrícolas da UEPG.

MATOS, S. N. **PADRÕES DE PROJETO - CRIACIONAIS**. 2017c. 26 slides. Material apresentado para a disciplina de Modelagem de Sistemas Agrícolas da UEPG.

MATOS, S. N. **PADRÕES DE PROJETO – DECORATOR E FACADE**. 2017d. 12 slides. Material apresentado para a disciplina de Modelagem de Sistemas Agrícolas da UEPG.

MATOS, S. N. **PADRÕES DE PROJETO – OBSERVER E STRATEGY**. 2017e. 20 slides. Material apresentado para a disciplina de Modelagem de Sistemas Agrícolas da UEPG.

MORAES, S. A. **Escalas diagramáticas para avaliar a intensidade das cercosporioses do amendoim**. O Agrônomo, v.39, n.1, p.9-18, 1987.

MORAES, S. A. Quantificação de doenças de plantas. 2007. Artigo em Hypertexto. Disponível em: <http://www.infobibos.com/Artigos/2007_1/doencas/index.htm>. Acesso em: 22/11/2017.

RIBEIRO, I. J. A. **Relatório das atividades desenvolvidas durante o estágio probatório**. Instituto Agrônomo de Campinas, 78p., 1970 (Relatório datilografado).