

UNIVERSIDADE DE SÃO PAULO  
INSTITUTO DE CIÊNCIAS MATEMÁTICAS E COMPUTAÇÃO

SSC0124 – Análise e Projeto Orientados a Objetos  
Profª Drª Elisa Yumi Nakagawa

**ANÁLISE E PROJETO DO SISTEMA DA POLÍCIA CIVIL**

Rodolfo Megiato de Lima – 7987286  
Elias Italiano Rodrigues – 7987251

SÃO CARLOS  
2013



# Sumário

1	Introdução.....	2
2	Diagrama de Comunicação.....	3
2.1	InserirDadosPessoais.....	3
2.2	InserirDadosOcorrencia.....	4
2.3	InserirDadosEvidencia.....	5
2.4	Escolha dos padrões GRASP.....	6
3	Diagrama de Classes do Sistema.....	7
4	Código-fonte Gerado.....	8
5	Conclusão.....	12
5.1	Sugestão.....	12
6	Anexos.....	13
6.1	Diagrama de Casos de Uso.....	13
6.2	Descrição Textual Abstrata de Três Casos de Uso.....	14
6.3	Descrição Concreta de Um Caso de Uso.....	17
6.4	Modelo Conceitual.....	18
6.5	Diagramas de Sequência do Sistema.....	19
6.5.1	Gerenciar Ocorrência.....	19
6.5.1.1	Fluxo Normal.....	19
6.5.1.2	Fluxo Alternativo: Buscar.....	20
6.5.1.3	Fluxo Alternativo: Alterar.....	21
6.5.2	Gerenciar Evidência.....	22
6.5.2.1	Fluxo Normal.....	22
6.5.2.2	Fluxo Alternativo: Buscar.....	23
6.5.2.3	Fluxo Alternativo: Alterar.....	24
6.5.3	Expedir Mandado de Prisão/Busca e Apreensão.....	25
6.5.3.1	Fluxo Normal.....	25
6.6	Contratos de Operação.....	26
6.6.1	InserirDadosPessoais(dados).....	26
6.6.2	InserirDadosOcorrencia(dados).....	26
6.6.3	InformarDadosEvidencia(dados, oc).....	26
6.6.4	EmitirMandado(ocorrencia).....	26

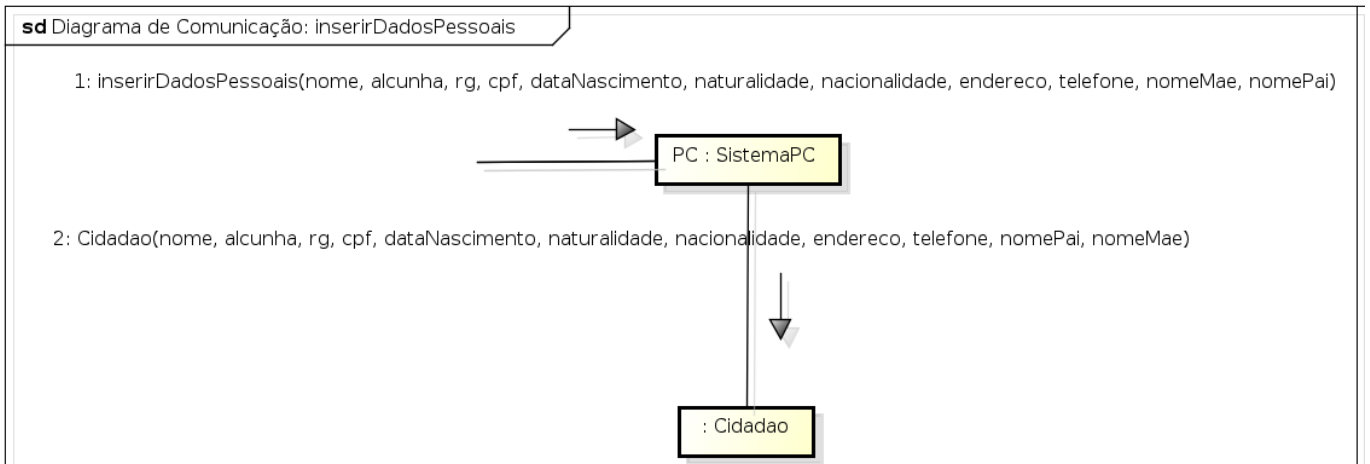


# 1 Introdução

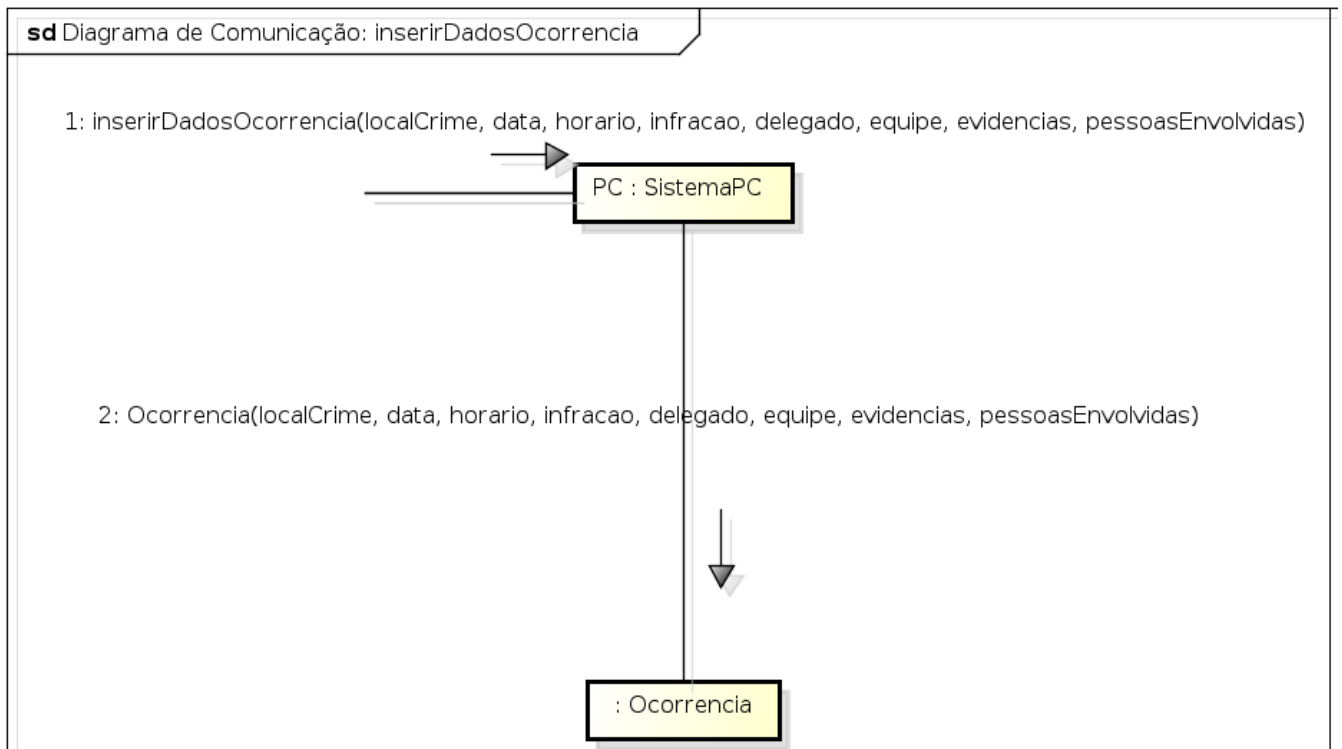
Este documento é referente a terceira parte do trabalho da disciplina de Análise e Projeto Orientados a Objetos em que são apresentados o Diagrama de Comunicação, o Diagrama de Classes do Sistema e o Código-fonte gerado a partir destes para o Sistema da Polícia Civil. Foi usada a ferramenta *Astah Professional* para confeccionar os diagramas da UML necessários nesta parte do projeto.

## 2 Diagrama de Comunicação

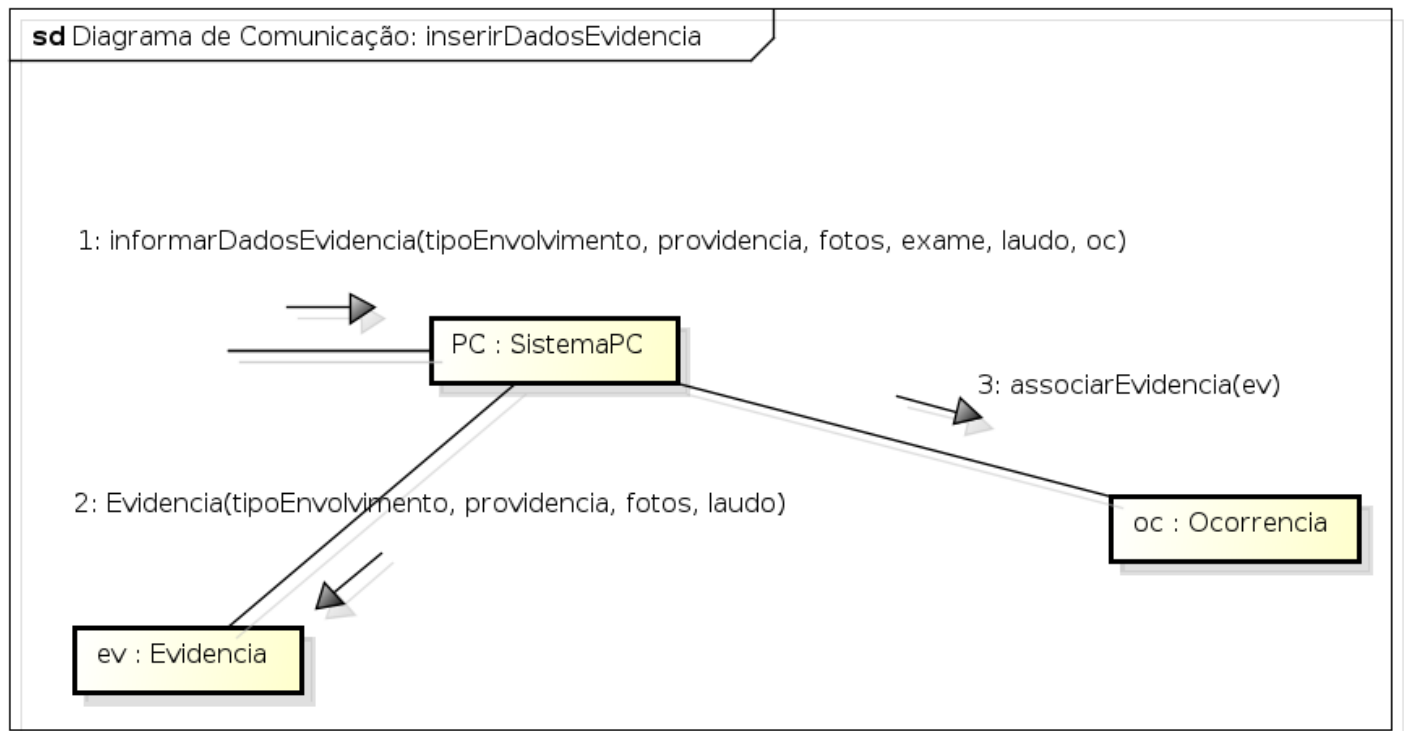
### 2.1 *InserirDadosPessoais*



## 2.2 InserirDadosOcorrencia



## 2.3 InserirDadosEvidencia



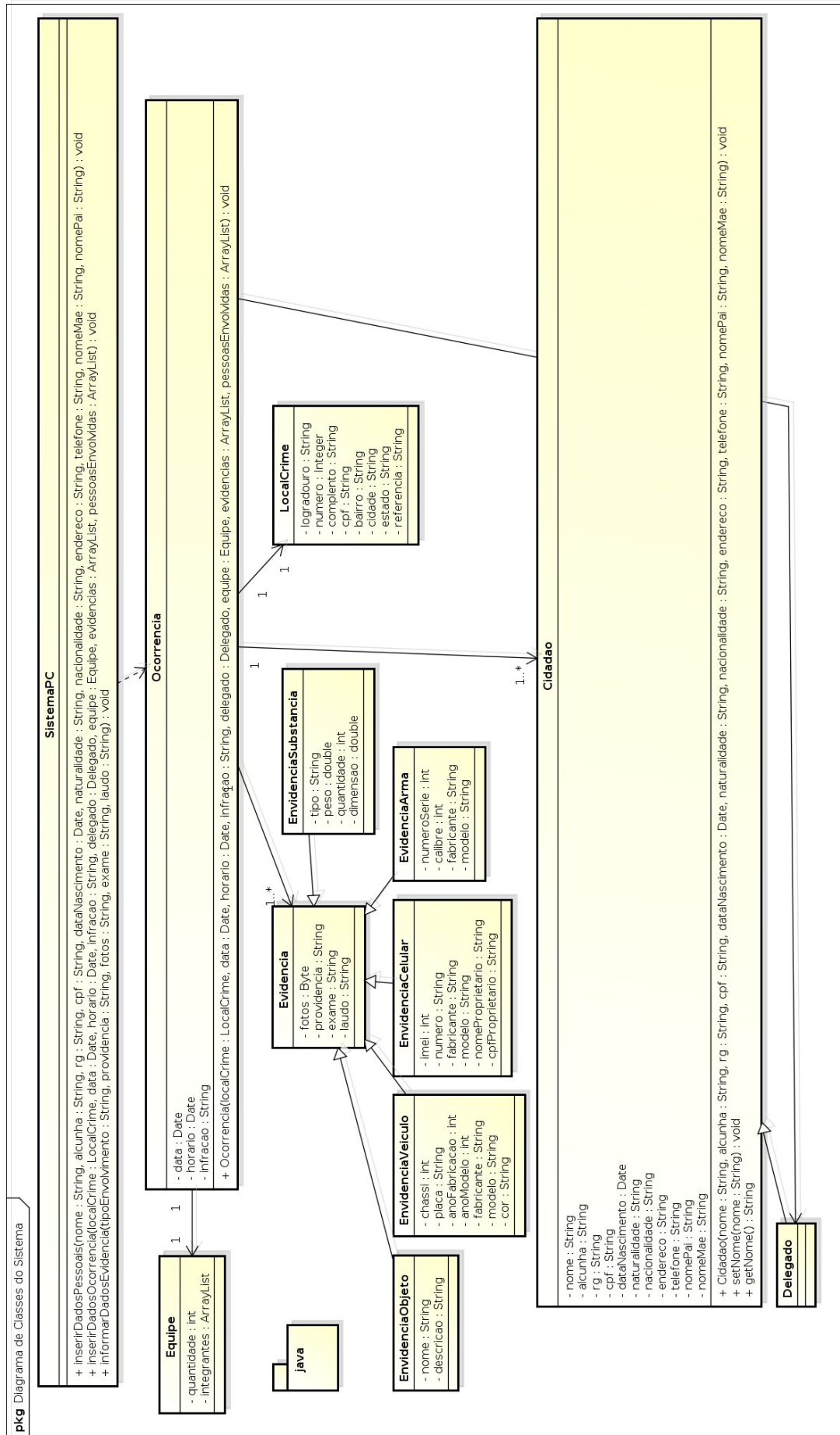


## **2.4 Escolha dos padrões GRASP**

Foram utilizados os padrões GRASP (General Responsibility Assignment Software Patterns) Controlador e Criador para a realização dos Diagramas de Comunicação, onde temos o SistemaPC como controlador e responsável por criar os objetos Cidadao e Ocorrencia uma vez que a ele é passado as informações necessárias para a criação de tais objetos. No Diagrama de Classes do Sistema pode-se observar os padrões de Acoplamento Fraco e Alta Coesão uma vez que há uma baixa dependência entre as classes de objetos e operações que se limitam ao escopo de cada classe.

Não foi possível estudar com detalhes os padrões GoF, portanto omitimos sua aplicação nesta última parte do trabalho.

### 3 Diagrama de Classes do Sistema



## 4 Código-fonte Gerado

Código-fonte em linguagem de programação Java gerado pela ferramenta UML *Astah Professional* sobre o diagrama de classes do sistema.

```
public class Cidadao {
    private String nome;
    private String alcunha;
    private String rg;
    private String cpf;
    private Date dataNascimento;
    private String naturalidade;
    private String nacionalidade;
    private String endereco;
    private String telefone;
    private String nomePai;
    private String nomeMae;

    public void Cidadao(String nome, String alcunha, String rg, String cpf, Date
dataNascimento, String naturalidade, String nacionalidade, String endereco,
String telefone, String nomePai, String nomeMae) { }
    public void setNome(String nome) { }
    public void setAlcunha(String alcunha) { }
    public void setRg(String rg) { }
    public void setCpf(String cpf) { }
    public void setDataNascimento(Date dataNascimento) { }
    public void setNaturalidade(String naturalidade) { }
    public void setNacionalidade(String nacionalidade) { }
    public void setEndereco(String endereco) { }
    public void setTelefone(String telefone) { }
    public void setNomePai(String nomePai) { }
    public void setNomeMae(String nomeMae) { }
    public String getNome() { return null; }
    public String getAlcunha() { return null; }
    public String getRg() { return null; }
    public String getCpf() { return null; }
    public Date getDataNascimento() { return null; }
    public String getNaturalidade() { return null; }
    public String getNacionalidade() { return null; }
    public String getEndereco() { return null; }
    public String getTelefone() { return null; }
    public String getNomePai() { return null; }
    public String getNomeMae() { return null; }
}

public class Delegado extends Cidadao { }

public class EnvidenciaCelular extends Evidencia {
    private int imei;
    private String numero;
    private String fabricante;
    private String modelo;
    private String nomeProprietario;
    private String cpfProprietario;

    public void setImei(int imei) { }
    public void setNumero(String numero) { }
```

```

    public void setFabricante(String fabricante) { }
    public void setModelo(String modelo) { }
    public void setNomeProprietario(String nomeProprietario) { }
    public int getImei() { return 0; }
    public String getNumero() { return null; }
    public String getFabricante() { return null; }
    public String getModelo() { return null; }
    public String getNomeProprietario() { return null; }
}

public class EvidenciaObjeto extends Evidencia {
    private String nome;
    private String descricao;

    public void setNome(String nome) { }
    public void setDescricao(String descricao) { }
    public String getNome() { return null; }
    public String getDescricao() { return null; }
}

public class EvidenciaSubstancia extends Evidencia {
    private String tipo;
    private double peso;
    private int quantidade;
    private double dimensao;

    public void setTipo(String tipo) { }
    public void setPeso(double peso) { }
    public void setQuantidade(int quantidade) { }
    public void setDimensao(double dimensao) { }
    public String getTipo() { return null; }
    public double getPeso() { return 0; }
    public int getQuantidade() { return 0; }
    public double getDimensao() { return 0; }
}

public class EvidenciaVeiculo extends Evidencia {
    private int chassi;
    private String placa;
    private int anoFabricacao;
    private int anoModelo;
    private String fabricante;
    private String modelo;
    private String cor;

    public void setChassi(int chassi) { }
    public void setPlaca(String placa) { }
    public void setAnoFabricacao(int anoFabricacao) { }
    public void setAnoModelo(int anoModelo) { }
    public void setFabricante(String fabricante) { }
    public void setModelo(String modelo) { }
    public void setCor(String cor) { }
    public int getChassi() { return 0; }
    public String getPlaca() { return null; }
    public int getAnoFabricacao() { return 0; }
    public int getAnoModelo() { return 0; }
    public String getFabricante() { return null; }
    public String getModelo() { return null; }
    public String getCor() { return null; }
}

```

```

public class Equipe {
    private int quantidade;
    private ArrayList integrantes;

    public void setQuantidade(int quantidade) { }
    public void setIntegrantes(ArrayList integrantes) { }
    public int getQuantidade() { return 0; }
    public ArrayList getIntegrantes() { return null; }
}

public class EvidenciaArma extends Evidencia {
    private int numeroSerie;
    private int calibre;
    private String fabricante;
    private String modelo;

    public void setNumeroSerie(int numeroSerie) { }
    public void setCalibre(int calibre) { }
    public void setFabricante(String fabricante) { }
    public void setModelo(String modelo) { }
    public int getNumeroSerie() { return 0; }
    public int getCalibre() { return 0; }
    public String getFabricante() { return null; }
    public String getModelo() { return null; }
}

public class Evidencia {
    private Byte fotos;
    private String providencia;
    private String exame;
    private String laudo;

    public void setFotos(Byte fotos) { }
    public void setProvidencia(String providencia) { }
    public void setExame(String exame) { }
    public void setLaudo(String laudo) { }
    public Byte getFotos() { return null; }
    public String getProvidencia() { return null; }
    public String getExame() { return null; }
    public String getLaudo() { return null; }
}

public class LocalCrime {
    private String logradouro;
    private Integer numero;
    private String complemento;
    private String cpf;
    private String bairro;
    private String cidade;
    private String estado;
    private String referencia;

    public void setLogradouro(String logradouro) { }
    public void setNumero(Integer numero) { }
    public void setComplemento(String complemento) { }
    public void setCpf(String cpf) { }
    public void setBairro(String bairro) { }
    public void setCidade(String cidade) { }
    public void setEstado(String estado) { }
    public void setReferencia(String referencia) { }
}

```

```

    public String getLogradouro() { return null; }
    public Integer getNumero() { return null; }
    public String getComplento() { return null; }
    public String getCpf() { return null; }
    public String getBairro() { return null; }
    public String getCidade() { return null; }
    public String getEstado() { return null; }
    public String getReferencia() { return null; }
}

public class Ocorrencia {
    private Date data;
    private Date horario;
    private String infracao;
    private Evidencia[] evidencia;
    private Delegado delegado;
    private Cidadao[] cidadao;
    private Equipe equipe;
    private LocalCrime localCrime;

    public void Ocorrencia(LocalCrime localCrime, Date data, Date horario,
String infracao, Delegado delegado, Equipe equipe, ArrayList evidencias,
ArrayList pessoasEnvolvidas) { }
    public void setData(Date data) { }
    public void setHorario(Date horario) { }
    public void setInfracao(String infracao) { }
    public Date getData() { return null; }
    public Date getHorario() { return null; }
    public String getInfracao() { return null; }
    public void associarEvidencia(Evidencia ev) { }
}

public class SistemaPC {
    public void inserirDadosPessoais(String nome, String alcunha, String rg,
String cpf, Date dataNascimento, String naturalidade, String nacionalidade,
String endereco, String telefone, String nomeMae, String nomePai) { }
    public void inserirDadosOcorrencia(LocalCrime localCrime, Date data, Date
horario, String infracao, Delegado delegado, Equipe equipe, ArrayList
evidencias, ArrayList pessoasEnvolvidas) { }
    public void informarDadosEvidencia(String tipoEnvolvimento, String
providencia, String fotos, String exame, String laudo, Ocorrencia oc) { }
}

```

## 5 Conclusão

Nessa terceira e última parte do trabalho, foram criados os diagramas da fase de projeto orientados a objeto do Sistema da Polícia Civil, complementando os diagramas realizados na primeira e segunda parte, concluindo então com a geração do "esqueleto" do código-fonte.

Foram desenvolvidos: o Diagrama de Comunicação, Diagrama de Classes do Sistema e a geração de Código-fonte a partir dos diagramas para o Sistema da Polícia Civil, que assim, terminam a parte de projeto do sistema complementando a fase de análise que foi desenvolvida anteriormente.

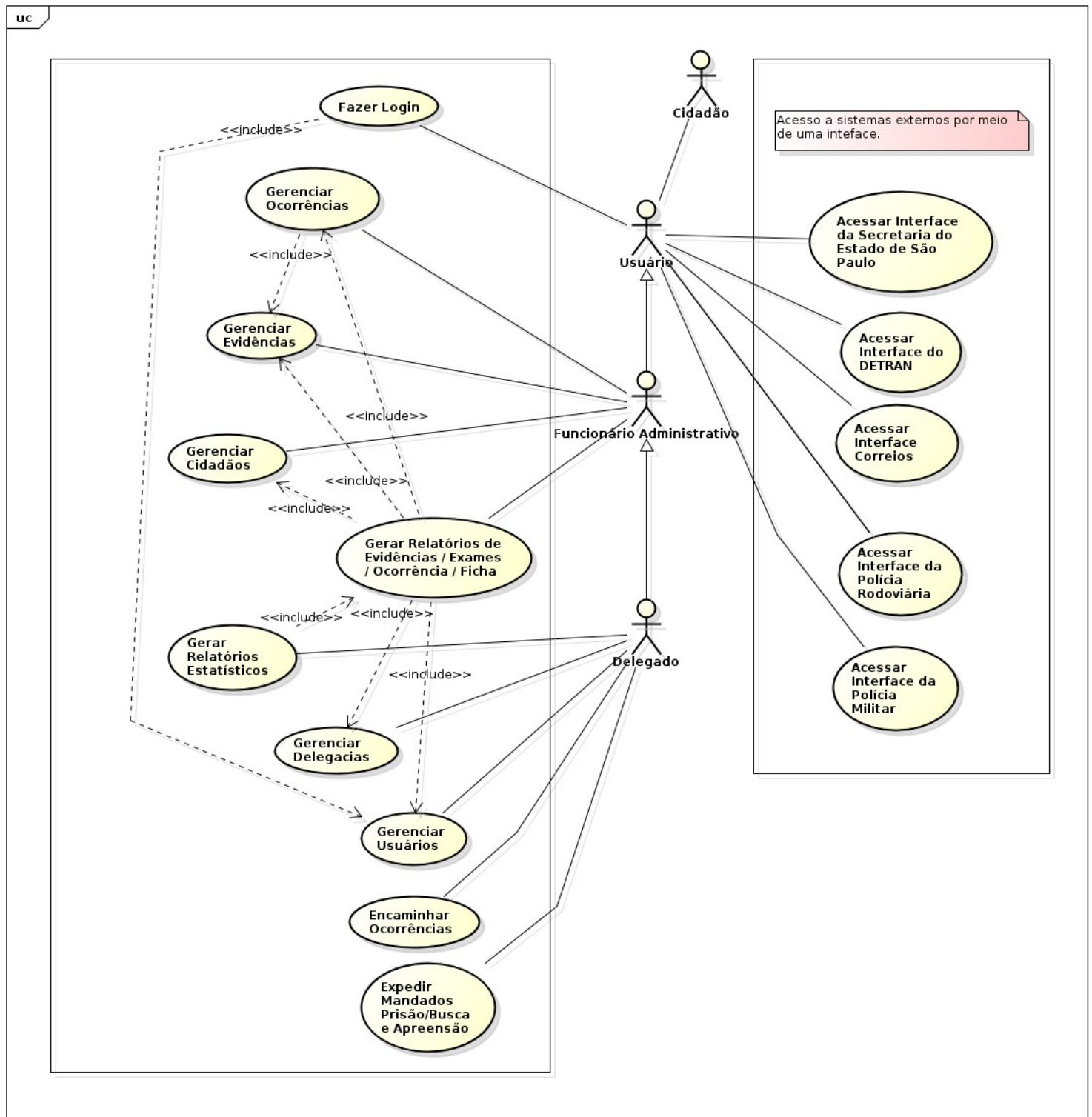
### 5.1 Sugestão

Correção dos exercícios de sala de aula, para que tenhamos um retorno sobre como resolver corretamente os diagramas da UML; fazer com que o cronograma das *aulas* da disciplina consiga cobrir por inteiro o seu conteúdo, principalmente se cobrado nos projetos.

## 6 Anexos

Seguem anexos os artefatos produzidos no segundo trabalho com alterações feitas de acordo com a correção indicada.

### 6.1 Diagrama de Casos de Uso





## 6.2 Descrição Textual Abstrata de Três Casos de Uso

**Caso de Uso:** Gerenciar Ocorrências.

**Ator Principal:** Funcionário Administrativo.

### Interessados e Interesses

- Cidadão: deseja registrar uma ocorrência na Delegacia da Polícia Civil.
- Funcionário Administrativo: atender à requisição do cidadão, registrando a ocorrência ou então realizar uma alteração/busca por uma ocorrência já existente.

**Pré-condições:** Funcionário Administrativo deve estar autenticado no sistema.

**Pós-condições:** Ocorrência deve estar incluída no sistema com *status* de "ativa". Se a operação de busca for realizada, os dados são retornados ao usuário com rapidez.

### Cenário de Sucesso Principal

1. O Cidadão chega ao balcão de atendimento da Delegacia da Polícia Civil e solicita uma ocorrência ao Funcionário Administrativo.
2. O Funcionário Administrativo seleciona a opção de Gerenciar Ocorrências no menu do sistema e requisita os dados ao Cidadão.
3. O Cidadão informa ao Funcionário Administrativo os dados sobre a ocorrência que deseja registrar como: local do crime, data, horário, infração/crime.
4. O Funcionário Administrativo acrescenta os dados sobre o delegado responsável, o policial envolvido, as evidências, as pessoas envolvidas e dados adicionais.
5. O Sistema acrescenta automaticamente a horário atual e o nome do Funcionário Administrativo responsável pelo registro, como também outras informações adicionais.
6. O Funcionário Administrativo finaliza a operação, tendo o seu *status* como "ativa".

### Fluxos Alternativos

(1) Se o Funcionário Administrativo necessitar, por algum motivo, buscar ou alterar os dados de uma ocorrência.

1. O Funcionário Administrativo seleciona a opção de buscar/alterar ocorrência do sistema.
2. O Funcionário Administrativo informa os dados para a busca e recebe o formulário preenchido com os dados.
  - 2.1. Se a operação em questão for de alteração, o Funcionário Administrativo insere os dados atualizados e seleciona a opção salvar.

(1-6) A qualquer momento da operação o Cidadão pode desistir de realizar a ocorrência.

1. O Funcionário Administrativo seleciona a opção de cancelar a ocorrência.

(3-4) A qualquer momento da operação os dados forem inconsistentes.

1. É enviada uma mensagem de erro na tela ao operador do sistema, e retorna ao ponto anterior ao erro.

**Caso de Uso:** Gerenciar Evidências.

**Ator Principal:** Funcionário Administrativo.

**Interessados e Interesses**

- Funcionário Administrativo: deseja incluir, alterar ou buscar uma evidência em uma ocorrência.

**Pré-condições:** Funcionário Administrativo deve estar autenticado no sistema e existir a ocorrência.

**Pós-condições:** A evidência deve estar incluída no sistema e associada a uma ocorrência. Da mesma forma, se for realizada a operação de alterar/buscar.

**Cenário de Sucesso Principal**

1. O Funcionário Administrativo seleciona a opção de Gerenciar Evidências no menu do sistema.
2. Informa o tipo de evidência, que pode ser: Arma, Celular, Objeto, Substância ou Veículo e a providência, que pode ser: "Apreendido", "Em busca" ou "A apurar".
3. Dependendo do tipo de evidência são informados dados específicos sobre a evidência.
4. Os dados são inseridos no sistema.

**Fluxos Alternativos**

(1) Se o Funcionário Administrativo necessitar, por algum motivo, buscar ou alterar os dados de uma evidência.

1. O Funcionário Administrativo seleciona a opção de buscar/alterar ocorrência do sistema.
2. O Funcionário Administrativo informa os dados para a busca e recebe o formulário preenchido com os dados.
  - 2.1. Se a operação em questão for de alteração, o Funcionário Administrativo insere os dados atualizados e seleciona a opção salvar.

(1-4) A qualquer momento da operação o Funcionário Administrativo pode desistir de realizá-la.

1. O Funcionário Administrativo seleciona a opção de cancelar a operação.

(1-2) A qualquer momento da operação os dados forem inconsistentes.

1. É enviada uma mensagem de erro na tela ao operador do sistema, e retorna ao ponto anterior ao erro.

**Caso de Uso:** Expedir Mandados Prisão/Busca e Apreensão.

**Ator Principal:** Delegado.

**Interessados e Interesses**

- Delegado: deseja expedir um mandado de prisão/busca e apreensão sobre uma ocorrência.

**Pré-condições:** Delegado deve estar autenticado e deve existir uma ocorrência relacionada.

**Pós-condições :** O mandado de prisão/busca e apreensão deve ser expedido.

**Cenário de Sucesso Principal**

1. O Delegado seleciona a opção de Expedir Mandados de Prisão/Busca e Apreensão no sistema.
2. O Delegado seleciona a ocorrência relacionada e emite o mandado de prisão/busca e apreensão.
3. Se o mandado for de prisão e a captura do cidadão relacionado estiver concluída, a ocorrência é atualizada com a captura do cidadão e o mandado deve ser marcado como "Cumprido".
4. Se o madado for de busca e apreensão e a apreensão da evidência estiver concluída, o *status* da providência do item deve ser alterado para "Apreendido" e mandado deve ser marcado como "Cumprido".

**Fluxos Alternativos**

(1-4) A qualquer momento da operação o Delegado pode desistir de realizá-la.

1. O Delegado seleciona o opção de cancelar a operação.

## 6.3 Descrição Concreta de Um Caso de Uso

**Caso de Uso:** Gerenciar Evidências.

**Ator Principal:** Funcionário Administrativo.

### Interessados e Interesses

- Funcionário Administrativo: deseja incluir, alterar ou buscar uma evidência em uma ocorrência.

**Pré-condições:** Funcionário Administrativo deve estar autenticado no sistema e existir a ocorrência.

**Pós-condições:** A evidência deve estar incluída no sistema e associada a uma ocorrência. Da mesma forma, se for realizada a operação de alterar/buscar.

### Cenário de Sucesso Principal

1. O Funcionário Administrativo seleciona a opção de Gerenciar Evidências no menu do sistema indicado na janela 1 pela letra A.
2. Informa o tipo de evidência selecionando umas das abas indicadas na janela 1 pela letra B, que pode ser: arma, celular, objeto, veículo ou substância; e a providência, que pode ser: "Apreendido", "Em busca" ou "A apurar" selecionada na região da janela 1 indicada pela letra C.
3. Dependendo do tipo de evidência são informados dados específicos sobre a evidência.
4. No tipo arma, como apresentado na janela 1, o Funcionário Administrativo insere sobre a arma: o número de série no campo de texto indicado pela letra D, o calibre no campo de texto indicado pela letra E, o fabricante no campo de texto indicado pela letra F, o modelo no campo de texto indicado pela letra G.
5. Para incluir os dados, o Funcionário Administrativo clica no botão “Confirmar” indicado na janela 1 pela letra H. A nova evidência é listada na grade de itens abaixo dos campos de texto.
6. Para confirmar os dados, o Funcionário Administrativo clica no botão “Salvar” indicado na janela 1 pela letra I. Os dados são inseridos no sistema.

### Fluxos Alternativos

(1) Se o Funcionário Administrativo necessitar, por algum motivo, buscar ou alterar os dados de uma evidência.

1. O Funcionário Administrativo insere o dado da busca no campo de texto de pesquisa da janela 1 e clica no botão “Buscar” indicado pela letra L. Os itens encontrados são listados na grade de itens.

2. Para alterar uma evidência já incluída, uma vez que ela está listada na grade de itens, o Funcionário Administrativo seleciona a evidência desejada e clica no botão “Editar” correspondente na coluna indicada pela letra K. Os dados aparecem preenchidos nos campos de texto acima. Após inserir os dados atualizados, o Funcionário Administrativo clica no botão “Confirmar” indicado pela letra H.

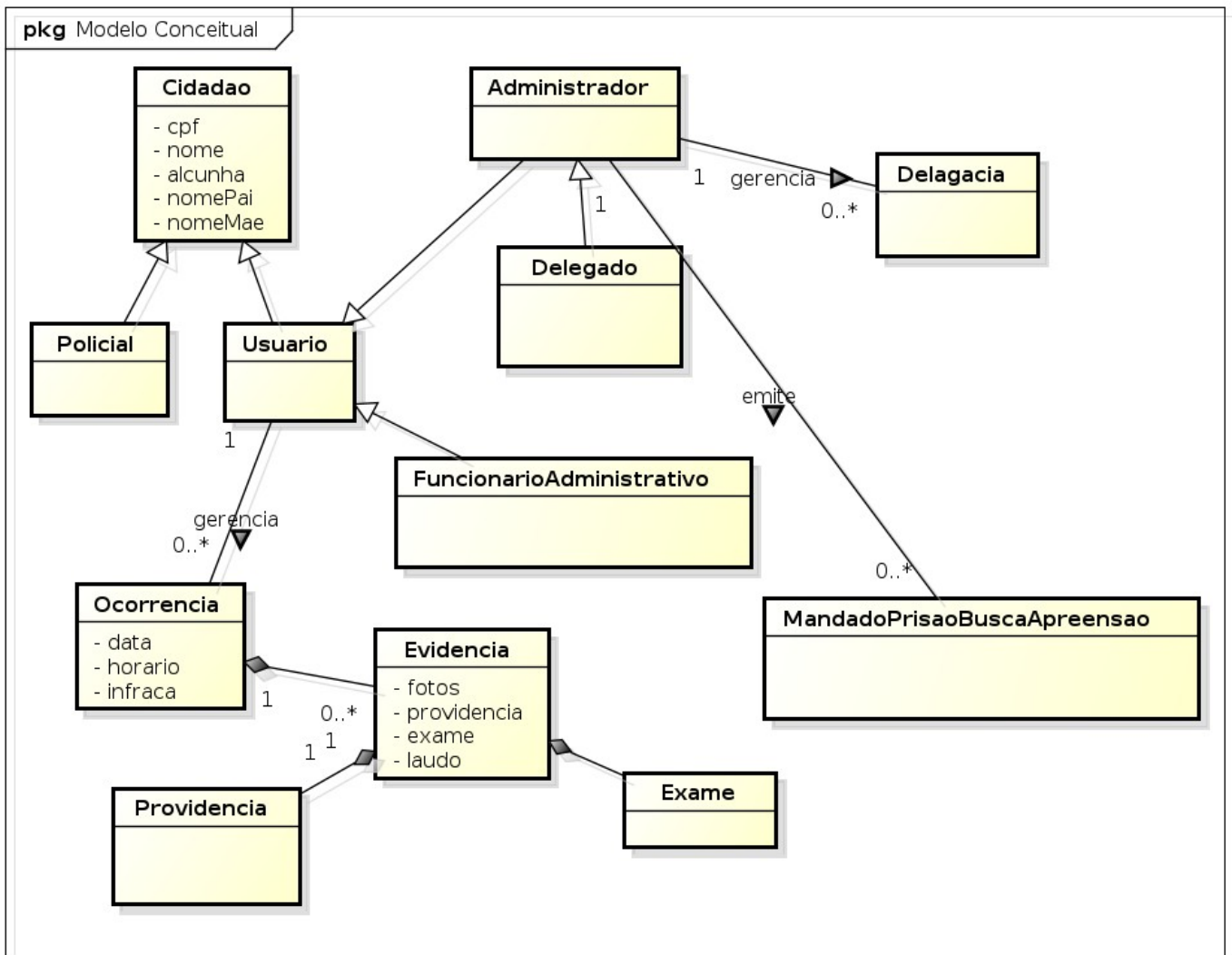
(1-6) A qualquer momento da operação o Funcionário Administrativo pode desistir de realizá-la.

1. O Funcionário Administrativo seleciona a opção de cancelar a operação clcando no botão “Cancelar” indicado pela letra J na janela 1.

(1-2) A qualquer momento da operação os dados forem inconsistentes.

1. É enviada uma mensagem de erro na tela ao operador do sistema, e retorna ao ponto anterior ao erro.

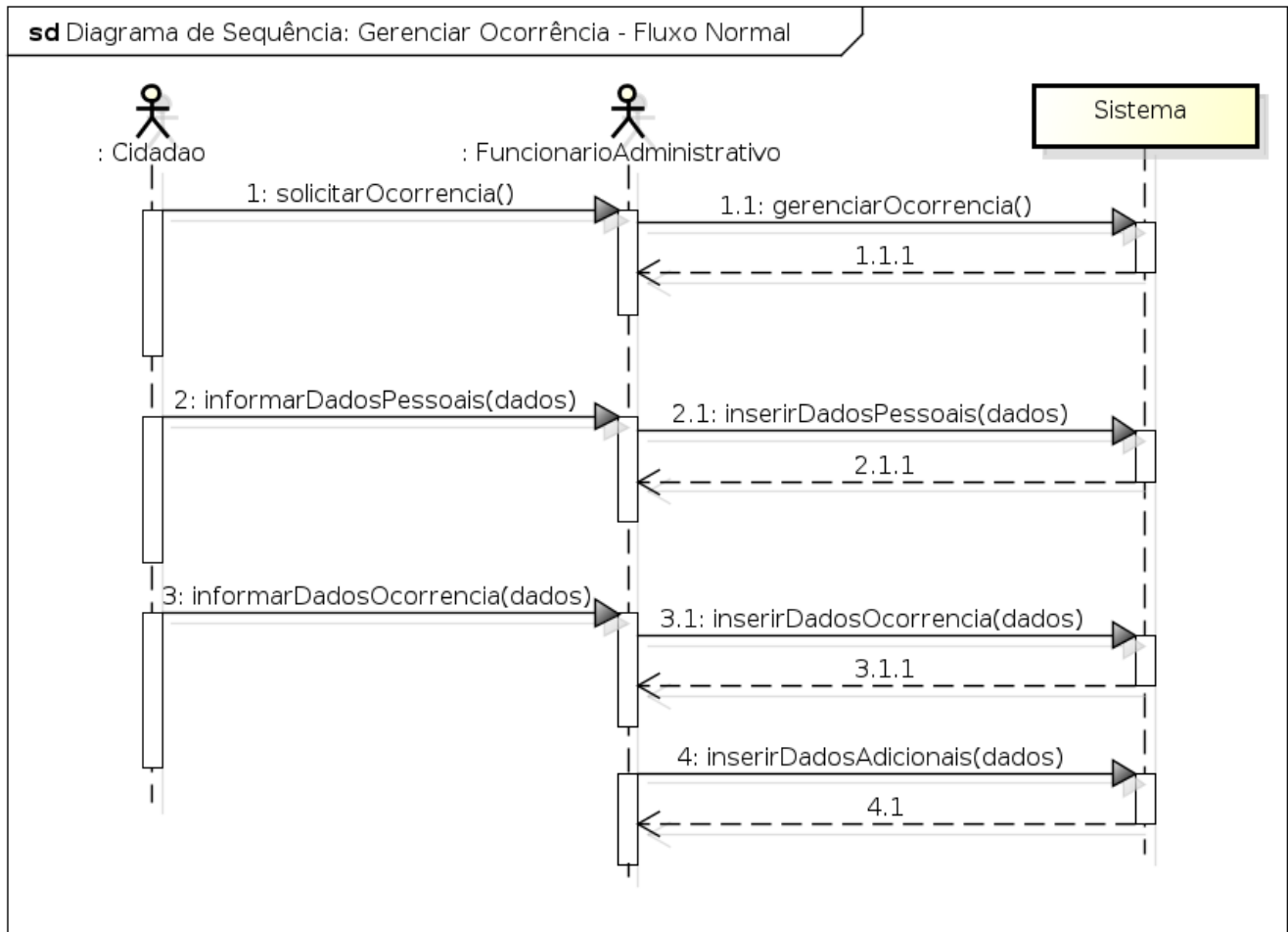
## 6.4 Modelo Conceitual



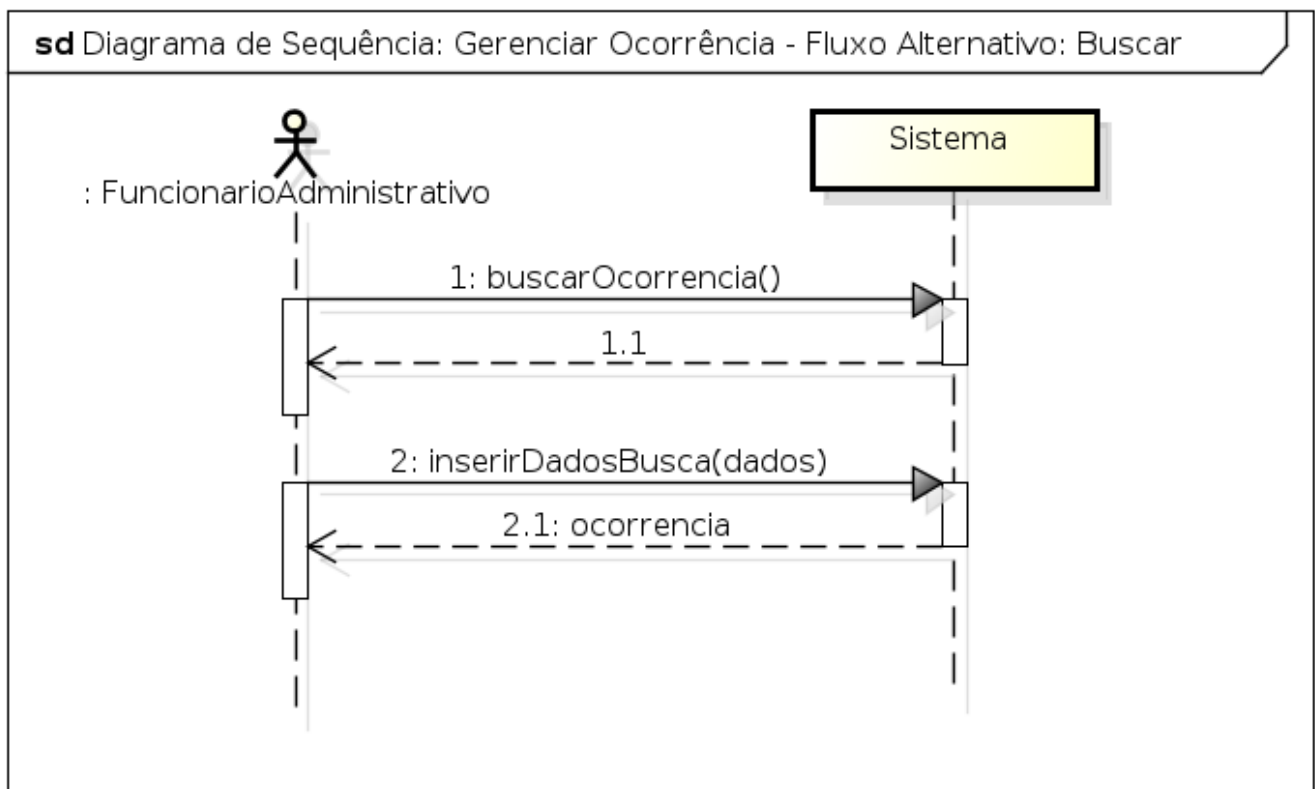
## 6.5 Diagramas de Sequência do Sistema

### 6.5.1 Gerenciar Ocorrência

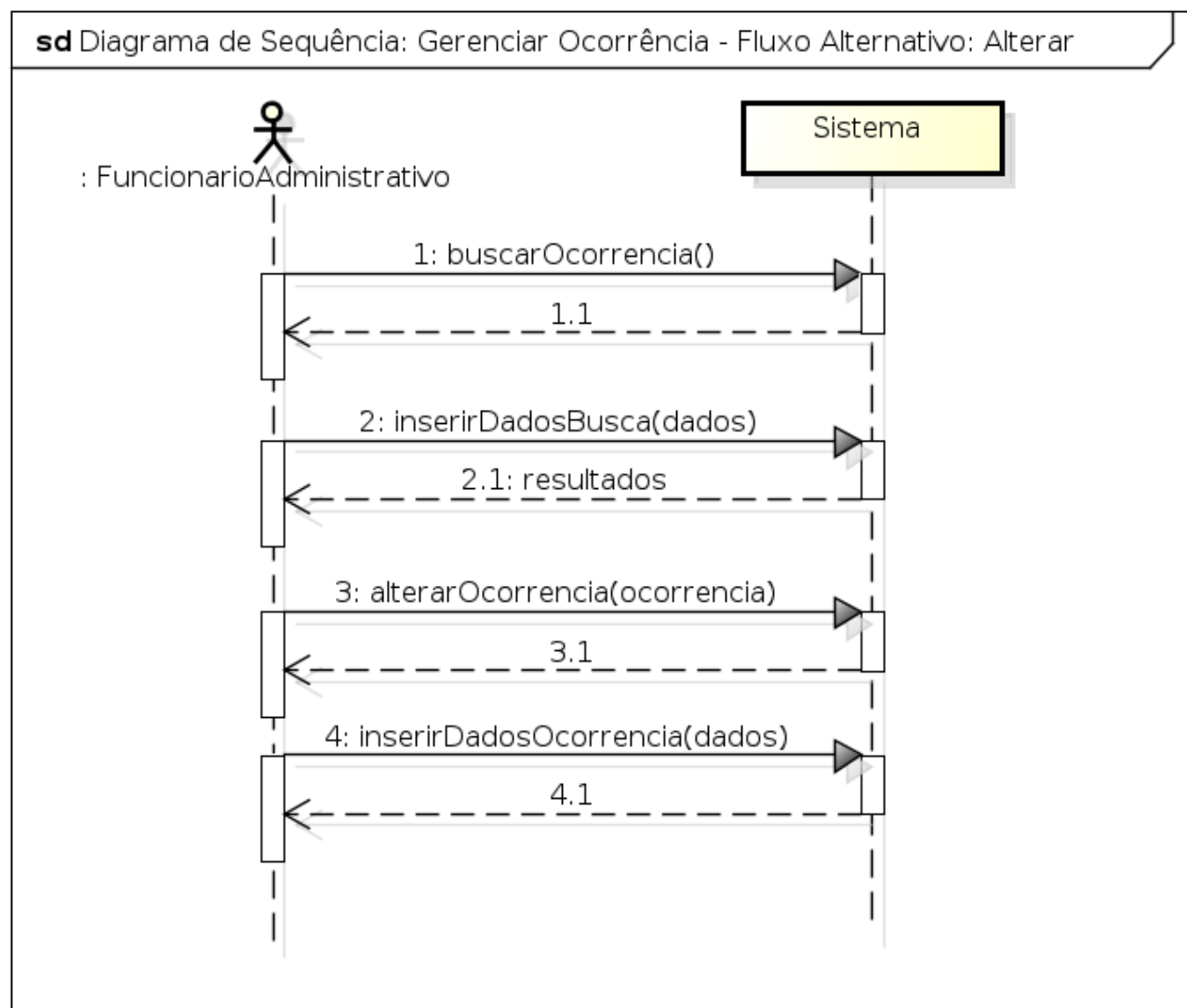
#### 6.5.1.1 Fluxo Normal



### 6.5.1.2 Fluxo Alternativo: Buscar



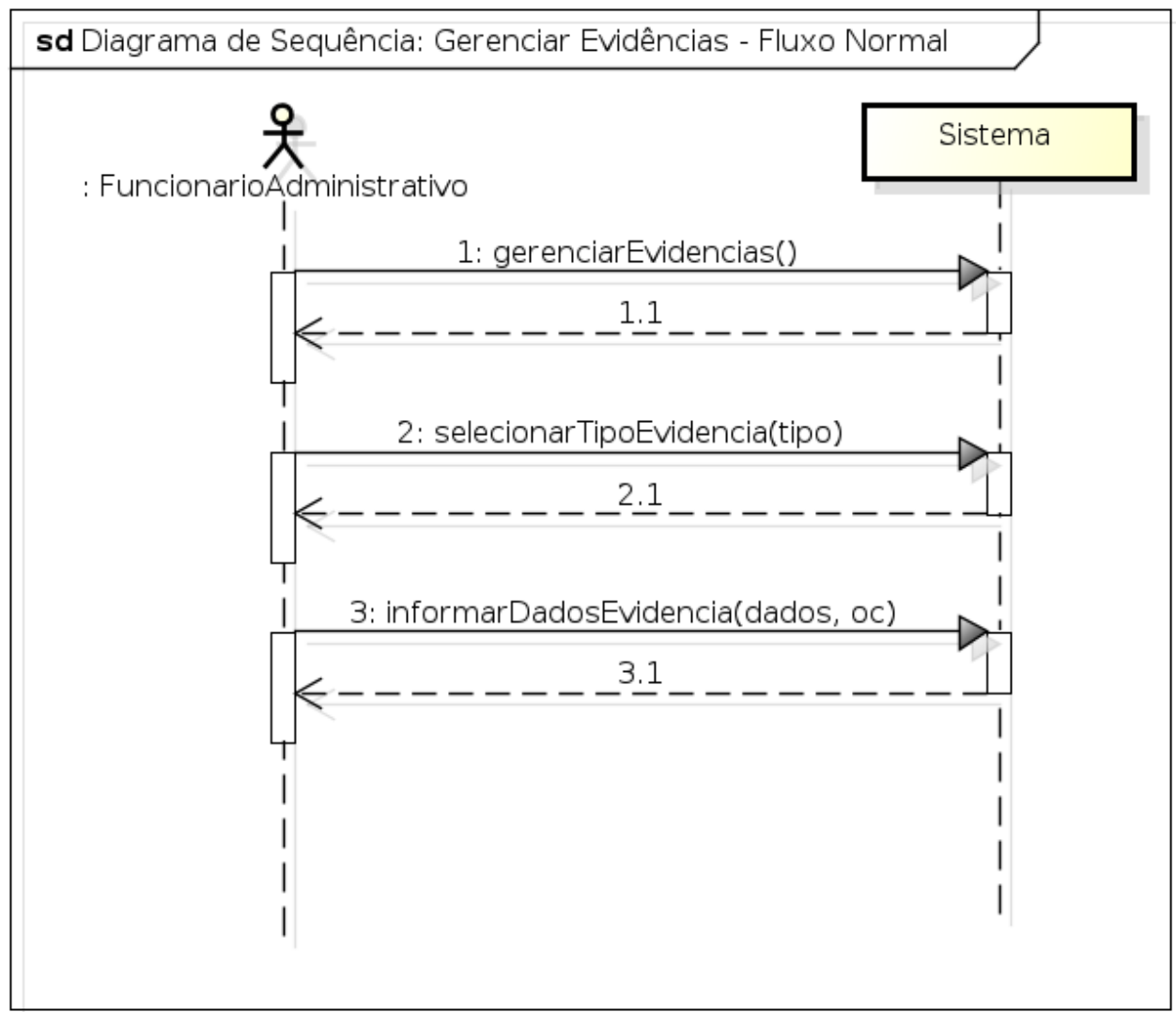
### 6.5.1.3 Fluxo Alternativo: Alterar



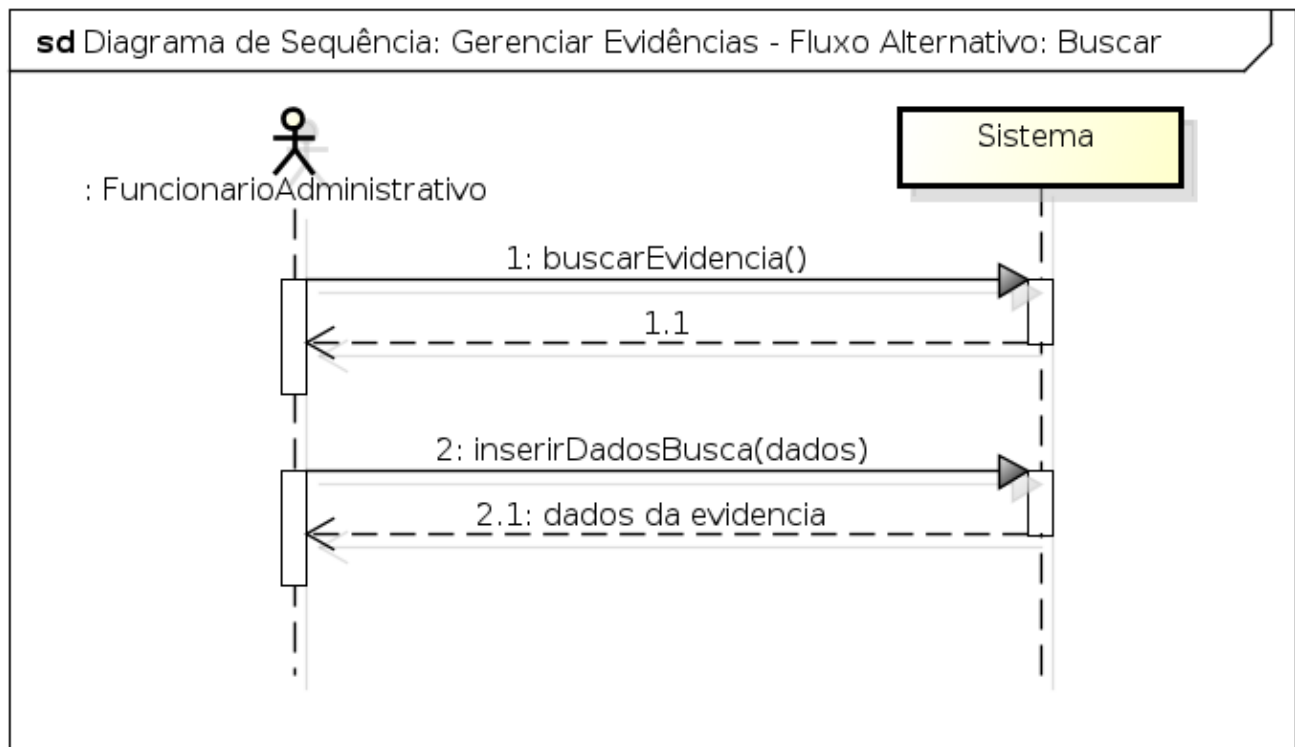


## 6.5.2 Gerenciar Evidência

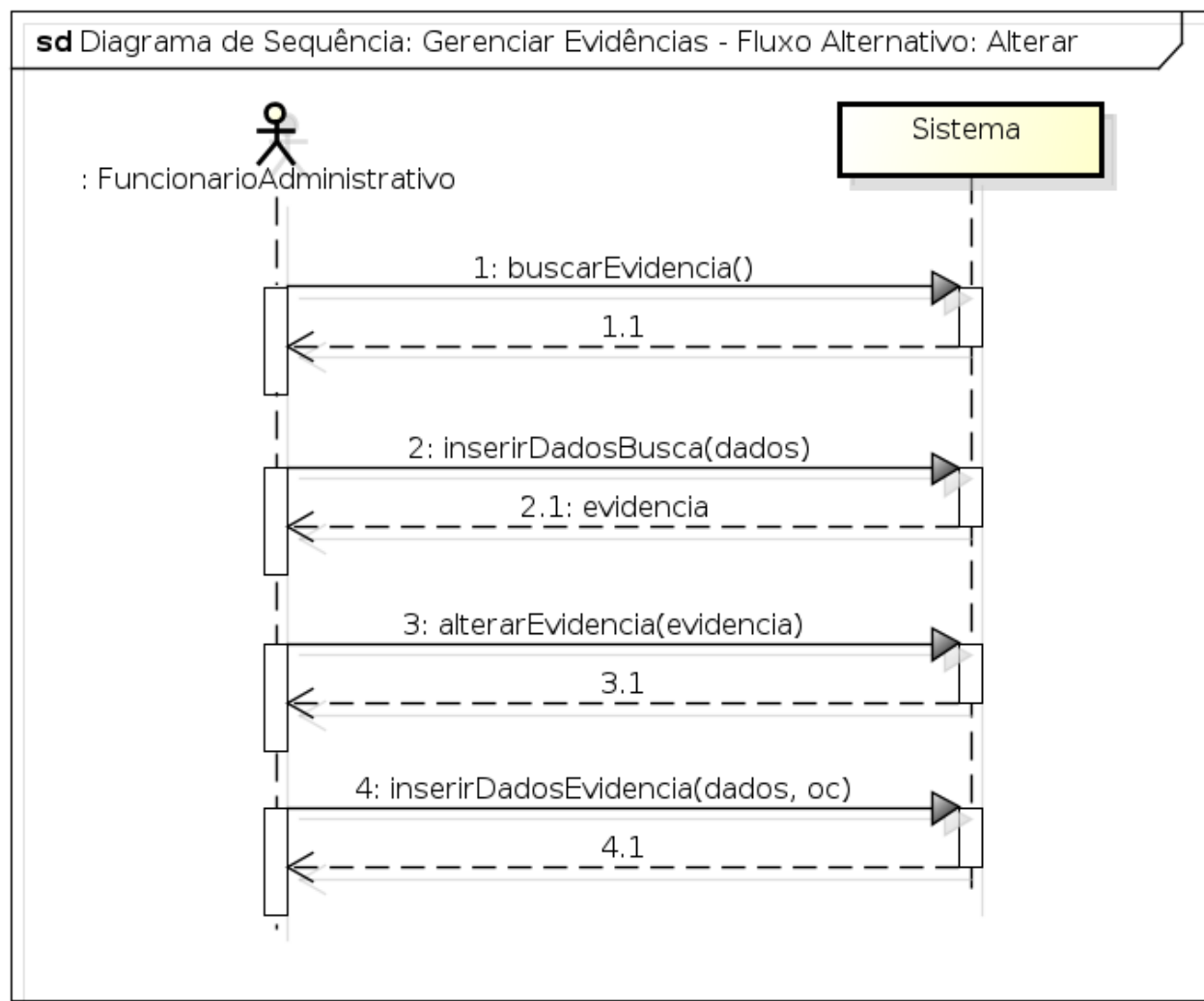
### 6.5.2.1 Fluxo Normal



### 6.5.2.2 Fluxo Alternativo: Buscar

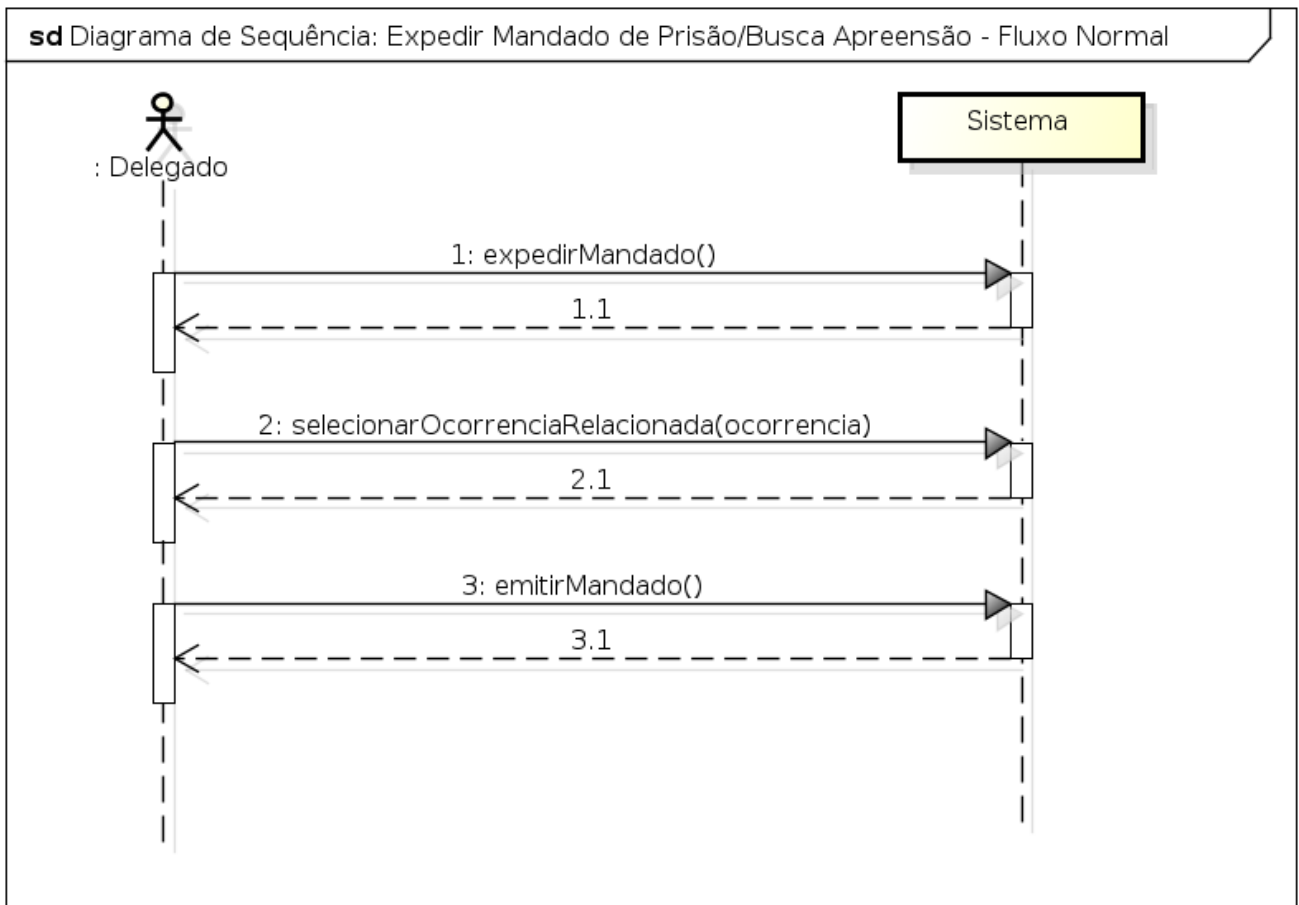


### 6.5.2.3 Fluxo Alternativo: Alterar



## 6.5.3 Expedir Mandado de Prisão/Busca e Apreensão

### 6.5.3.1 Fluxo Normal



## 6.6 Contratos de Operação

### 6.6.1 InserirDadosPessoais(dados)

**Operação:** inserirDadosPessoais(dados)

**Parâmetros de Entrada:** nome, alcunha (vulgo), RG, CPF, data de nascimento, naturalidade, nacionalidade, endereço, telefone, nome da mãe, nome do pai.

**Referências Cruzadas:** Caso de Uso Gerenciar Ocorrências.

**Pré-Condições:** O Funcionário Administrativo deve estar autenticado no sistema.

**Pós-Condição:** O cidadão deve estar cadastrado no sistema e apto para estar relacionado a uma ocorrência.

### 6.6.2 InserirDadosOcorrencia(dados)

**Operação:** inserirDadosOcorrencia(dados)

**Parâmetros de Entrada:** local do crime (logradouro, número, complemento, CEP, bairro, cidade, estado, referência), a data, o horário, a infração/crime, o delegado responsável, a equipe policial envolvida (no mínimo um policial), as evidências e as pessoas envolvidas.

**Referências Cruzadas:** Caso de Uso Gerenciar Ocorrências.

**Pré-Condições:** O Funcionário Administrativo deve estar autenticado no sistema assim como o cidadão que deseja registrar a ocorrência.

**Pós-Condição:** A ocorrência deve estar cadastrada no sistema.

### 6.6.3 InformarDadosEvidencia(dados, oc)

**Operação:** informarDadosEvidencia(dados)

**Parâmetros de Entrada:** tipo de envolvimento (arma do crime, objeto subtraído, a apurar) e a providência (apreendido, em busca, a apurar), fotos, exame e laudo. Se a evidência for do tipo Arma: número de série, o calibre, o fabricante e o modelo. Se a evidência for do tipo Celular: o número do IMEI, o número do celular, o fabricante, o modelo, o nome e o CPF do proprietário. Se a evidência for do tipo Objeto: o nome e a descrição. Se a evidência for do tipo Substância: o tipo de substância, o peso total, a embalagem (pacote, capsula, frasco), a quantidade unitária e a dimensão (largura, altura, comprimento) em centímetros da unidade. Se a evidência for do tipo Veículo: número do chassi, a placa, o ano de fabricação, o ano do modelo, o fabricante, o modelo e a cor. Ocorrência a evidência será relacionada.

**Referências Cruzadas:** Caso de Uso Gerenciar Evidências.

**Pré-Condições:** O Funcionário Administrativo deve estar autenticado no sistema e existir a ocorrência.

**Pós-Condição:** Uma instância de evidência existe no sistema e foi associada a uma ocorrência.

### 6.6.4 EmitirMandado(ocorrencia)

**Operação:** emitirMandado(ocorrencia)

**Parâmetros de Entrada:** Ocorrência relacionada.

**Referências Cruzadas:** Caso de Uso Expedir Mandados de Prisão/Busca e Apreensão.

**Pré-Condições:** O Delegado deve estar autenticado no sistema e deve existir uma ocorrência relacionada.