

Descrição da atividade:

- Arquitetura da aplicação

- Mapeamento de classes de análise em elementos de projeto.
- Diagrama de classes da arquitetura, organização em pacotes.
- Pelo menos um dos casos de uso deve envolver interação com um sistema externo (de forma que surjam subsistemas na arquitetura).
- O projeto deve explorar o máximo possível o uso de padrões de projeto e arquiteturais.

# Arquitetura da aplicação

## Deskconnect

<b>Mapeamento das classes de análise em elementos de projeto.....</b>	<b>2</b>
Identificar classes de projeto.....	2
Identificar subsistemas.....	3
<b>Organização em pacotes.....</b>	<b>3</b>
<b>Visão geral da arquitetura.....</b>	<b>4</b>

## Mapeamento das classes de análise em elementos de projeto

### Identificar classes de projeto

*Uma classe de análise simples será mapeada para uma classe de projeto.*

#### **Conta**

A classe de análise Conta será uma única classe de Projeto, contendo todos os atributos que definimos previamente para esta classe, sendo estes *login* e *senha*.

#### **Funcionário**

A classe de análise Funcionário será dividida em três classes: Funcionario, Gerente e Endereco. A classe Endereco contém os atributos de *estado*, *cidade*, *bairro*, *rua*, *CEP* e *número*. Todos os Funcionários e Gerentes devem conter uma referência a Endereco, especificando seus respectivos endereços. As classes de Funcionário e Gerente devem conter os atributos: *Nome*, *CPF*, *Função*, *Salário* e *Endereço*.

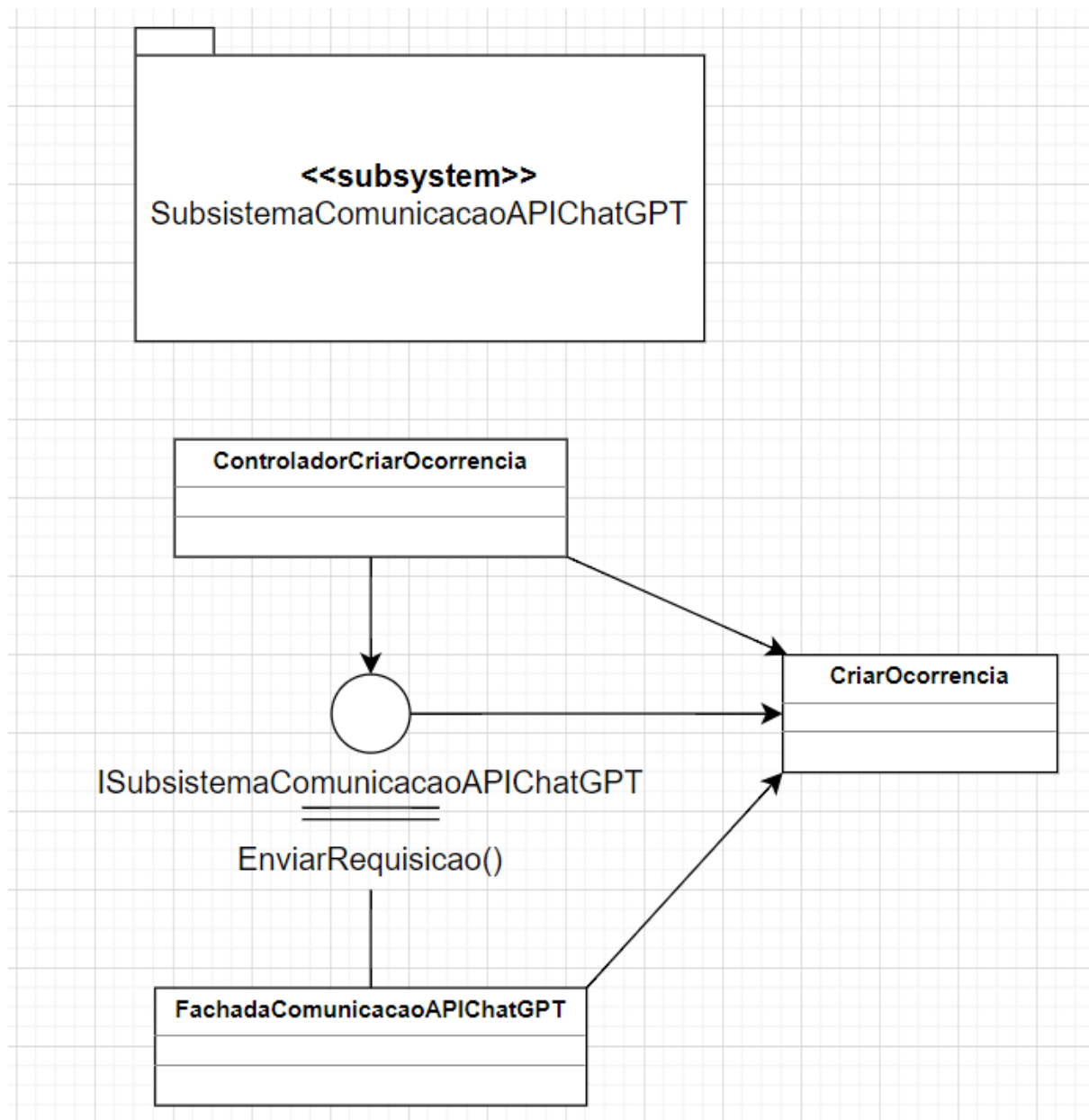
#### **Ocorrência**

A classe de análise Ocorrência corresponde a uma classe de projeto. Esta classe possui os seguintes atributos: *Título*, *Responsável*, *Setor*, *Descrição*, *Data de criação*, *Data de início* e *Data de fim*.

#### **Histórico**

A classe de análise Histórico não deverá ser refletida em uma classe de projeto, dado que o histórico é apenas a coleção ou subconjunto da coleção de ocorrências.

## Identificar subsistemas



## Organização em pacotes

Implementação de padrões de projeto planejadas

1. Observer: Notificação do Deskconnect quando uma nova ocorrência é criada.
2. Simple factory ou builder: Criação de instâncias de Ocorrência.

3. Singleton: talvez para criar uma única instância do Dashboard. (Atenção: o Singleton não contará no total de patterns solicitados).

## Visão geral da arquitetura

