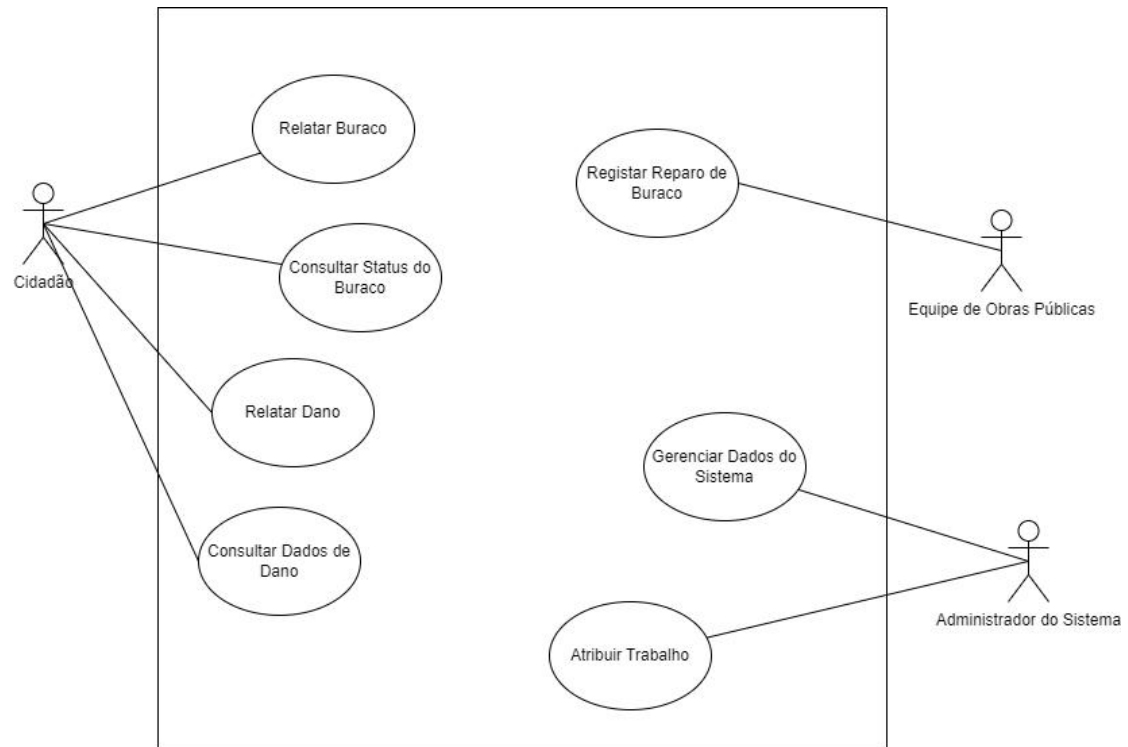
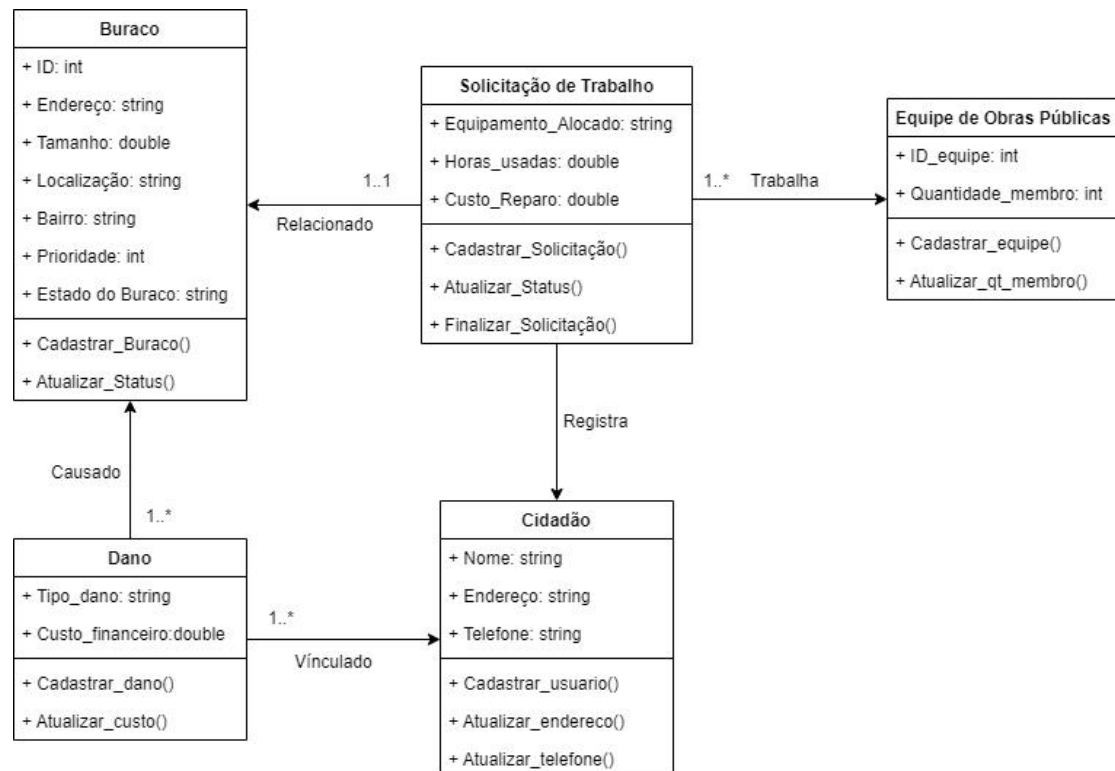


Projeto PHTR

1) Diagrama de Caso de Uso



2) Diagrama de Classes



3) Sistema de Contexto

- Cidadãos: Usuários que relatam buracos e danos.
- Equipes de Obras Públicas: Responsáveis pela execução dos reparos nos buracos.
- Administradores do Sistema: Gerenciam e supervisionam o sistema.
- Banco de Dados: Armazena informações sobre buracos, solicitações de trabalho, danos e outras informações relevantes.
- Recursos de Hardware e Software: Incluem servidores web, bancos de dados, navegadores web para acesso dos cidadãos e aplicativos para equipes de obras públicas.

4) Arquitetura de Alto Nível

- Camada de Apresentação: Responsável pela interface do usuário web. Inclui páginas web para cidadãos relatarem buracos, consultar status e relatar danos, bem como uma interface de administração para os administradores do sistema gerenciarem dados.
- Camada de Lógica de Negócios: Contém a lógica de processamento central do sistema, incluindo a gestão de buracos, atribuição de trabalhos, registro de reparos, cálculo de custos, e gerenciamento de dados do sistema. Esta camada é acessada pela camada de apresentação e interage com a camada de dados.
- Camada de Dados: Responsável pelo armazenamento e recuperação de dados. Inclui um banco de dados para armazenar informações sobre buracos, solicitações de trabalho, danos e outras informações relacionadas.

5) Refinamento da Arquitetura de Software

Camada de Apresentação:

- Interface do Usuário Web para Cidadãos.
- Interface do Usuário Web para Administradores.
- Páginas para Relatar Buracos, Consultar Status, Relatar Danos e Gerenciar Dados.
- Módulos de Autenticação e Autorização.

Camada de Lógica de Negócios:

- Controladores para a lógica de negócios.
- Classes para Gerenciamento de Buracos.

- Classes para Atribuição de Trabalho.
- Classes para Registro de Reparo.
- Cálculo de Custos.
- Gerenciamento de Dados do Sistema.
- Lógica para Priorização de Reparos.

Camada de Dados:

- Banco de Dados Relacional (por exemplo, MySQL, PostgreSQL) para armazenar informações de buracos, solicitações de trabalho e danos.
- ORM (Object-Relational Mapping) para interação com o banco de dados.
- Repositórios para acesso aos dados.

Interfaces de Comunicação:

- APIs REST para interações entre a camada de apresentação e a camada de lógica de negócios.
- Interfaces de comunicação interna entre os componentes da camada de lógica de negócios.