

# Sistemas Multiplataformas

## Nome do Projeto

| Nome Completo     | RA         |
|-------------------|------------|
| Nicholas Carmona  | 23.01004-5 |
| Enzo Dourado      | 23.10289-6 |
| Rian Cardoso      | 22.01292-3 |
| André Martins     | 23.10193-8 |
| Guilherme Engrich | 23.00133-0 |
| Eduardo Ferraz    | 23.01527-6 |

- **Levantamento dos Requisitos do Sistema de Software**

**Objetivo:** Apresentar o levantamento dos requisitos do Sistema de Software e a forma de extração dos Requisitos.

- **Extração de Requisitos**

- Para a extração dos requisitos foi feita até o momento duas reuniões com o parceiro. Em relação às reuniões juntamente com o cliente, todas foram realizadas via Microsoft Teams de forma Online, facilitando assim o contato e a tirada de dúvidas que se apresentaram durante a estruturação
- As perguntas feitas ao parceiro foram a seguinte:
- Qual email para contato?
- Quais são as principais funcionalidades que o software deve ter?
- Há alguma tecnologia específica que precisa ser usada?
- O software será usado por diferentes tipos de usuários?

- **Análise da Coleta de Requisitos**

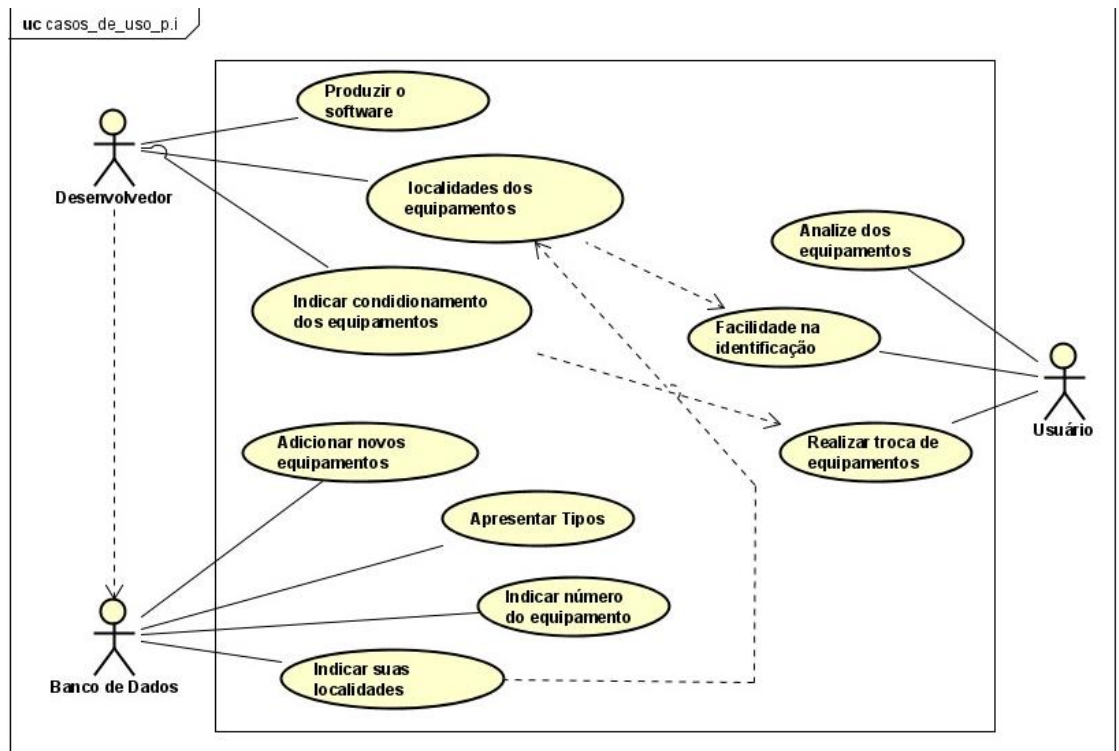
- O objetivo da aplicação será conseguir manter um bom e organizado sistema de monitoramento de todos os equipamentos de combate a incêndio do metrô e isso inclui funcionalidades como rastreabilidade dos equipamentos, monitoramento da data de validade e localização, registro de manutenções, emissão de relatórios e programação de trocas.

- **Especificação dos Requisitos do Sistema de Software**

**Objetivo:** Especificar os requisitos funcionais e não funcionais do sistema.

- **Requisitos Funcionais**

- As principais funcionalidades do sistema são:
- - Adicionar localidades dos extintores dentro do metrô.
- - Indicar aqueles que estão prontos para a utilização.
- - Facilitar aos trabalhadores possíveis trocas destes equipamentos.
- - Identificação de troca de equipamento para controle dos trabalhadores.

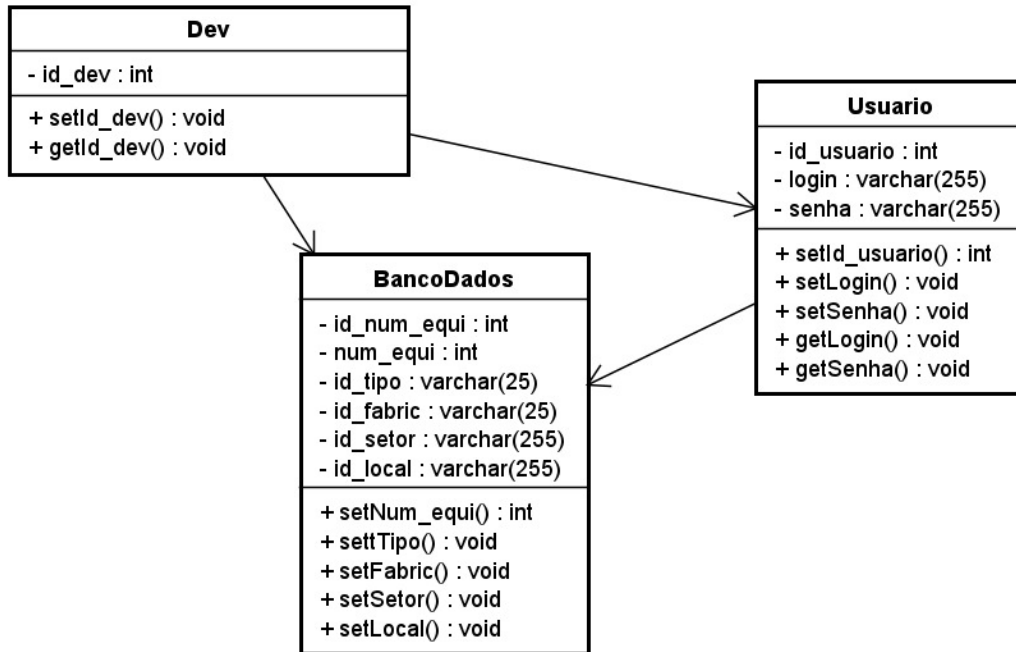


- **2.2. Requisitos Não-Funcionais**

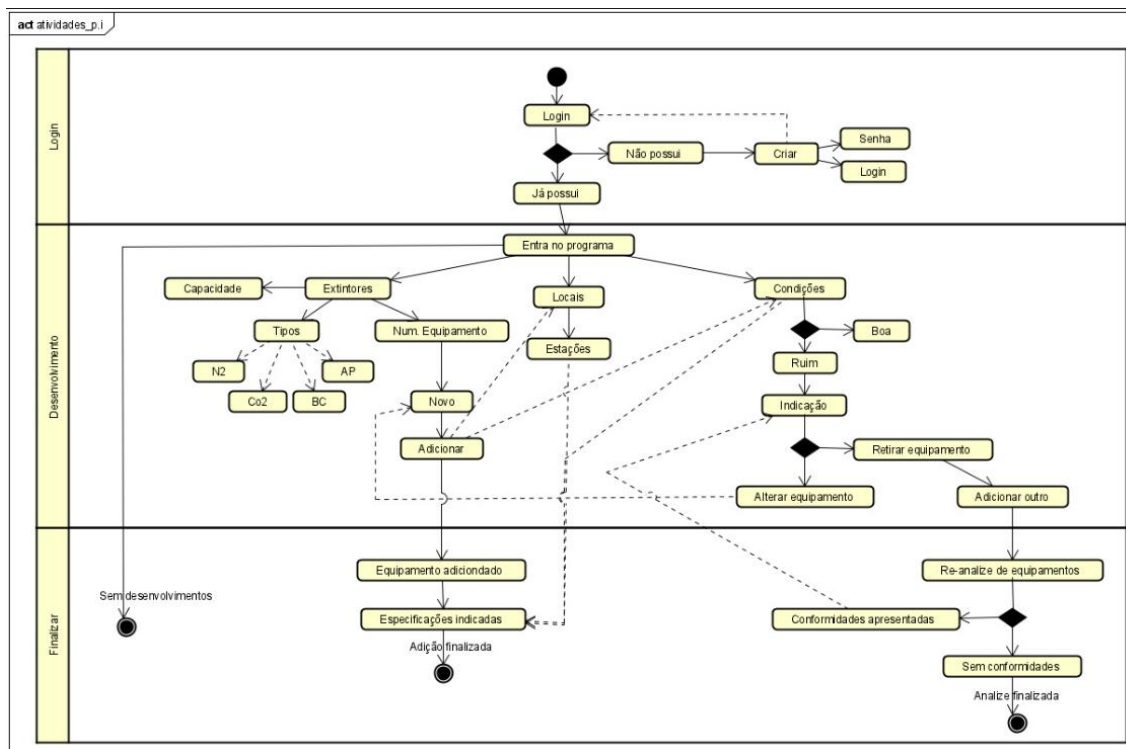
- O sistema deve ser capaz de realizar as indicações dos RF em tempo real.
- - O sistema não apresentará os funcionários cadastrados.
- - O sistema não poderá ser acessado por pessoas não cadastradas.
- - O sistema não permitirá acesso de pessoas que não são funcionárias do metrô.
- **Análise/Projeto**

- **Diagrama de Classes**

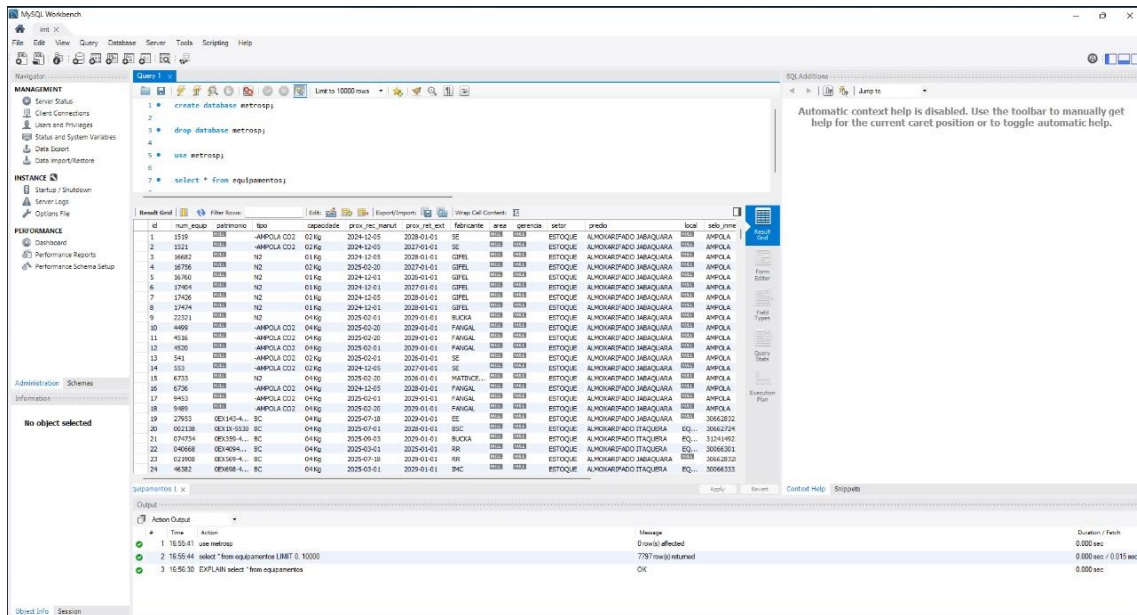
pkg classes\_p.i



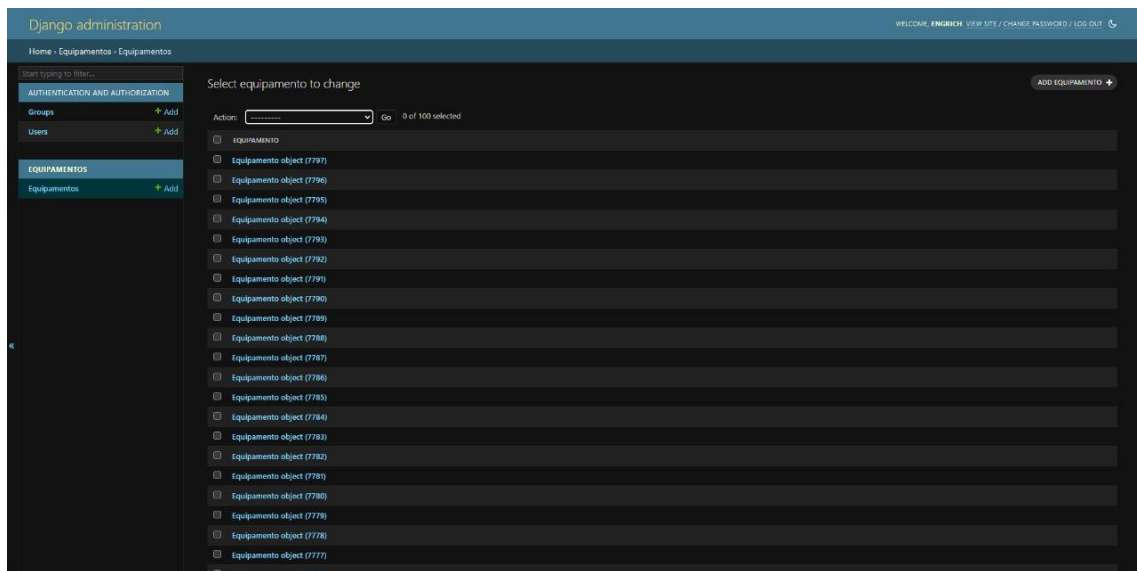
## Diagrama de Atividades



- **Diagrama de Estados (opcional)**
- Sequências de estados pelas quais o objeto pode passar durante seu ciclo de vida em resposta a eventos.
- **Modelagem de Banco de Dados**
- **Banco de Dados Relacional**



- **Banco de Dados Não-Relacional**



- **Implementação**

- Indicação do link de acesso ou repositório de toda a implementação de programação.
- Considerações sobre a implementação multiplataforma, incluindo frameworks e bibliotecas utilizadas para garantir a compatibilidade.
- Integração dos bancos de dados relacional e não-relacional com o sistema.

- **Design de Interface**

- Apresentação do design de interface por meio de mapa de navegação, design das telas em wireframes, fluxograma de interação entre telas, diálogos usuário-sistema, elementos de tela (botões, menus, campos para entrada de dados, etc.), estilos de fontes, paleta de cores, entre outros.
  - Validação da arquitetura da informação com cardsorting.

- **Protótipo de Telas**

- Apresentação das telas (formato de Wireframe ou Mockup) do projeto já com aplicação das fontes, cores, diagramação, etc.
  - Descrição das telas utilizando ferramentas como Figma, Pencil, Balsamiq, Axure, Photoshop, Adobe XD, PowerPoint, etc.

- **Validação de Usabilidade**

- Definição da técnica escolhida para validar a implementação na perspectiva da usabilidade.
  - Detalhamento do mandado de avaliação (objetivos, tarefas, métricas e questionários de coleta de dados) e descrição da percepção de avaliação (resultados), incluindo cálculos estatísticos como médias e desvios.

- **Redesign**

- Apresentação do redesign nos casos em que os testes de usabilidade apontarem alterações. Caso não exista alteração, este item pode ser excluído.

- **Questões Legais**

- Apresentação das leis de acessibilidade e de software que podem ser aplicadas no projeto.

- **Testes**

- Indicação do link de acesso ou repositório de todas as evidências de testes unitários realizados no projeto de acordo com os casos de uso especificados.
- Estratégias de teste para garantir a funcionalidade em diferentes plataformas.
- Testes específicos para verificar a integração e desempenho dos bancos de dados relacional e não-relacional.

- **Resultados e Considerações**

- Principais “prints” das telas do sistema de software desenvolvido, com uma breve explicação de cada tela.
- Considerações gerais do projeto, focando nos requisitos implementados e nos resultados obtidos.
- Desafios e soluções específicas para garantir a funcionalidade multiplataforma.
- Discussão sobre a eficiência e desempenho dos bancos de dados relacional e não-relacional.

## **7. Apêndice I**

- Roteiro de entrevista ou questionário respondido.

### **Observações Adicionais**

Para garantir a funcionalidade multiplataforma, é recomendável utilizar frameworks como React Native, Flutter, Xamarin, Electron, entre outros, que facilitam a criação de aplicações que funcionam em diversas plataformas a partir de um único código-base. Além disso, a escolha de bancos de dados não-relacionais pode ser baseada em suas vantagens específicas, como escalabilidade e flexibilidade, para lidar com grandes volumes de dados e estruturas variáveis.