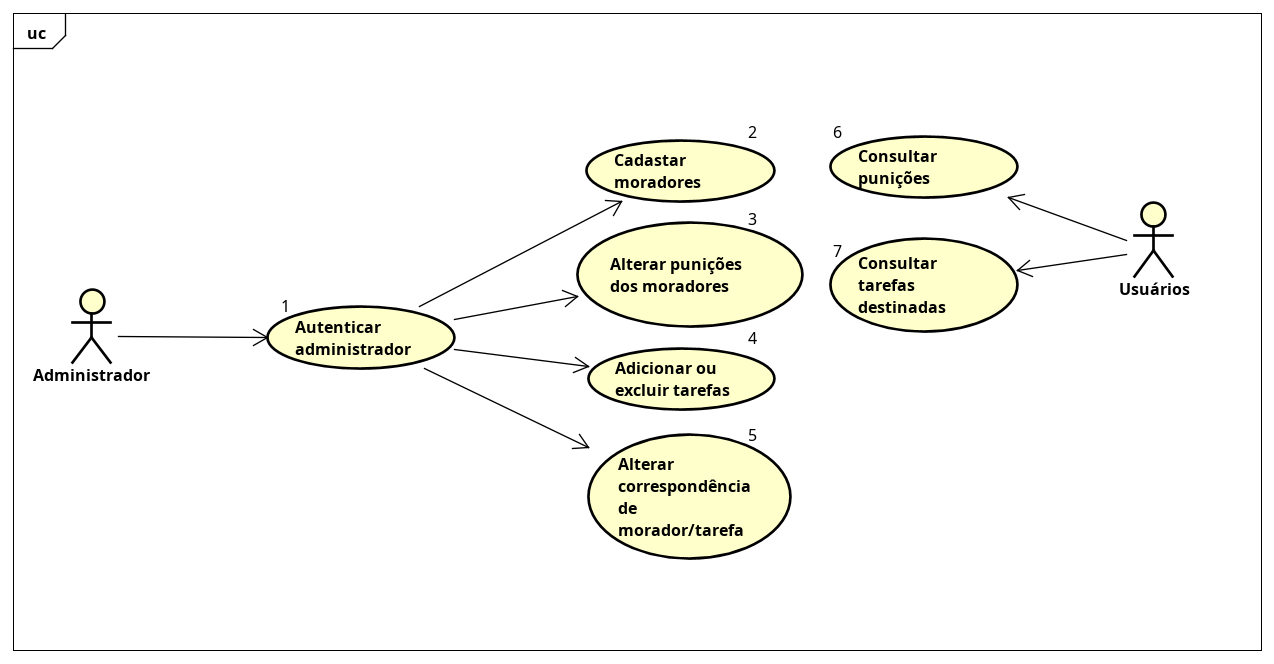
**Relatório Casos de Uso**

**EA-975 Grupo 2**

Henrique Roberto da Cunha Junior - 174638 | Leonardo Rodrigues Marques - 178610

Diagrama de Casos de Uso:



1. **Autenticar Administrador:**

* Atores envolvidos:
  + Primário: Administrador
* Fluxo de eventos:
  + Principal: O administrador insere seu usuário e senha.
  + Alternativo: Inserir um usuário que não existe no banco

Inserir uma senha incorreta para um usuário

* + Exceção: Impossibilidade de se comunicar com o banco para inserir novo usuário administrador
* Pós-condições: Administrador autenticado com sucesso

1. **Cadastrar moradores:**

* Atores envolvidos:
  + Primário: Administrador
  + Secundário: Moradores
* Pré-condições:
  + Administrador (Ator primário) deve estar autenticado.
* Fluxo de eventos:
  + Principal: O administrador cadastra um novo usuário com Nome, Ra, Curso que devem ser diferentes de vazio. E coluna punição, inteiro começando em 0.
  + Alternativo: Inserir algum Ra, diferente de números.
  + Exceção: Impossibilidade de se comunicar com o banco para inserir o morador.
* Pós-condições: Usuário cadastrado com sucesso, e sua coluna punição com valor 0.

1. **Alterar punições dos Moradores:**

* Atores envolvidos:
  + Primário: Administrador
  + Secundário: Moradores
* Pré-condições:
  + Administrador (Ator primário) deve estar autenticado.
  + Devem existir pelo menos um Morador (Ator secundário) cadastrado.
* Fluxo de eventos:
  + Principal: O administrador incrementa a coluna de punições de um determinado morador.
  + Exceção: Impossibilidade de se comunicar com o banco para atualizar a tabela.
  + Pós-condições: Dados do morador atualizado com sucesso.

1. **Alterar ou excluir tarefas:**

* Atores envolvidos:
  + Primário: Administrador
* Pré-condições:
  + Administrador (Ator primário) deve estar autenticado.
* Fluxo de eventos:
  + Principal: O administrador cadastra um novo tipo de tarefa na tabela Tarefas, com Nome, Id.
* Exceção: Impossibilidade de se comunicar com o banco para inserir a nova linha na tabela
* Pós-condições: Linha inserida na tabela e tabela atualizada com sucesso.

1. **Alterar correspondência de morador/tarefa:**

* Atores envolvidos:
  + Primário: Administrador
  + Secundário: Moradores
* Pré-condições:
  + Administrador (Ator primário) deve estar autenticado.
  + Deve haver uma relação entre morador e tarefa pré-estabelecida.
* Fluxo de eventos:
  + Principal: O administrador seleciona o morador e altera a tarefa correspondente a ele.
  + Exceção: Impossibilidade de se comunicar com o banco para alterar a correspondência entre morador e tarefa.
* Pós-condições:
  + Alteração de tarefa para determinado morador realizada com sucesso

1. **Consultar punições:**

* Atores envolvidos:
  + Primário: Morador (usuário comum)
  + Secundário: Moradores
* Pré-condições:
  + Deve haver pelo menos um morador para consultarmos as punições
* Fluxo de eventos:
  + Principal: O usuário solicita uma tabela informando todos os moradores com suas respectivas punições (número)
  + Alternativo: Sem moradores a página viria em branco.
  + Exceção: Impossibilidade de se comunicar com o banco para consultar as punições.

Tabela de moradores vazia causa uma exceção, pois não seria possível acessar nenhum.

* Pós-condições: A tabela de punições é exibida com sucesso para o usuário.

1. **Consultar tarefas destinadas:**

* Atores envolvidos:
  + Primário: Morador (usuário comum)
* Pré-condições:
  + Deve haver pelo menos um morador e pelo menos alguma tarefa para esse morador
* Fluxo de eventos:
  + Principal: O usuário solicita uma tabela que informa todos os moradores e suas respectivas tarefas.
  + Alternativo: Se não houver moradores com tarefas respectivas, a tabela viria vazia
  + Exceção: Usuário vazio causaria um null pointer, pois tentariamos acessar as tarefas de um morador que não existe.
* Pós-condições: A tabela de correspondências é exibida com sucesso.