 <b>Unifor</b>	<b>UNIVERSIDADE DE FORTALEZA</b>	
<b>Curso: Ciência da Computação</b>		<b>Turma: T197-10</b>
<b>Disciplina: Desenvolvimento de plataformas moveis</b>		
<b>Aluno(s): João Guilherme de Castro, Davi Menezes, Guilherme Amaral, Camila Martins</b>		
<b>Turno: Manhã</b>	<b>Data: 09/04/2024</b>	<b>Período Letivo:</b>
<b>TRABALHO</b>		

## Entre Línguas e Sons: A Arte da Tradução de Obras Literárias do Português para o Inglês e a Inovação dos Áudio Guias

FORTALEZA  
2024

## SUMÁRIO

<b><i>VISÃO GERAL</i></b>	<b>3</b>
Introdução	3
Objetivo	3
Justificativa	3
Benchmark	3
<b><i>METODOLOGIA PARA O DESENVOLVIMENTO DE SOFTWARE</i></b>	<b>4</b>
Descrição da Metodologia	4
Processo da Metodologia	4
<b><i>ARTEFATOS DO PRODUTO</i></b>	<b>5</b>
Atores	5
Requisitos Funcionais	5
Requisitos Não-funcionais	5
Protótipo de Baixa Fidelidade	6
Diagrama de Caso de Uso	7

# VISÃO GERAL

## Introdução

No contexto atual, a tecnologia se apresenta como uma ponte vital entre o público e o acesso à cultura. Especialmente em espaços culturais como a UNIFOR, onde a riqueza de obras é vasta, existe uma oportunidade única de utilizar aplicativos móveis para enriquecer a experiência dos visitantes. Este projeto de software visa desenvolver um aplicativo dedicado a oferecer textos explicativos, audioguias, e traduções em inglês das obras expostas, promovendo não apenas a educação cultural, mas também a inclusão social de falantes não nativos e pessoas com deficiência visual.

## Objetivo

O objetivo deste trabalho é desenvolver um aplicativo móvel que facilite o acesso à informação cultural e artística no espaço cultural da UNIFOR. Pretende-se oferecer uma interface intuitiva que apresente textos informativos, audioguias detalhados e traduções em inglês, visando enriquecer a experiência dos visitantes, promover a inclusão social, e potencializar a acessibilidade das obras para um público mais amplo, incluindo visitantes internacionais e pessoas com deficiência visual.

## Justificativa

A escolha deste tema fundamenta-se na observação da lacuna existente na comunicação e na interação dos visitantes com as obras em espaços culturais. Frequentemente, a falta de informações acessíveis e a barreira linguística limitam a plena apreciação e o entendimento das peças expostas. Além disso, a importância de promover a inclusão social e acessibilidade justifica a necessidade de desenvolver um aplicativo que não só transcenda as barreiras linguísticas, mas que também torne a arte mais acessível a todos, independente de suas limitações físicas ou conhecimento linguístico.

## Benchmark

App X (Museum Guide): Oferece audioguias; não suporta múltiplos idiomas.

## VISÃO GERAL

App Z (Art Explorer): Apresenta informações textuais sobre obras; limitado em recursos de acessibilidade.

App Y(Nosso App)

	App X	App Z	App Y Nosso App
Possui informações textuais das obras		X	X
Possui tradução para inglês			X
Possui audioguias	X		X
Possui ampla acessibilidade			X

# METODOLOGIA PARA O DESENVOLVIMENTO DE SOFTWARE

## Descrição da Metodologia

O desenvolvimento do aplicativo seguirá a metodologia ágil Scrum, caracterizada por sua flexibilidade e eficiência em projetos de inovação tecnológica. Scrum é baseado em ciclos de desenvolvimento chamados Sprints, permitindo uma adaptação contínua às necessidades do projeto e do usuário final. Os papéis envolvidos incluem o Product Owner, responsável pela visão do produto; o Scrum Master, que assegura a aderência ao Scrum; e a equipe de desenvolvimento, multifuncional e autogerida.

## Processo da Metodologia

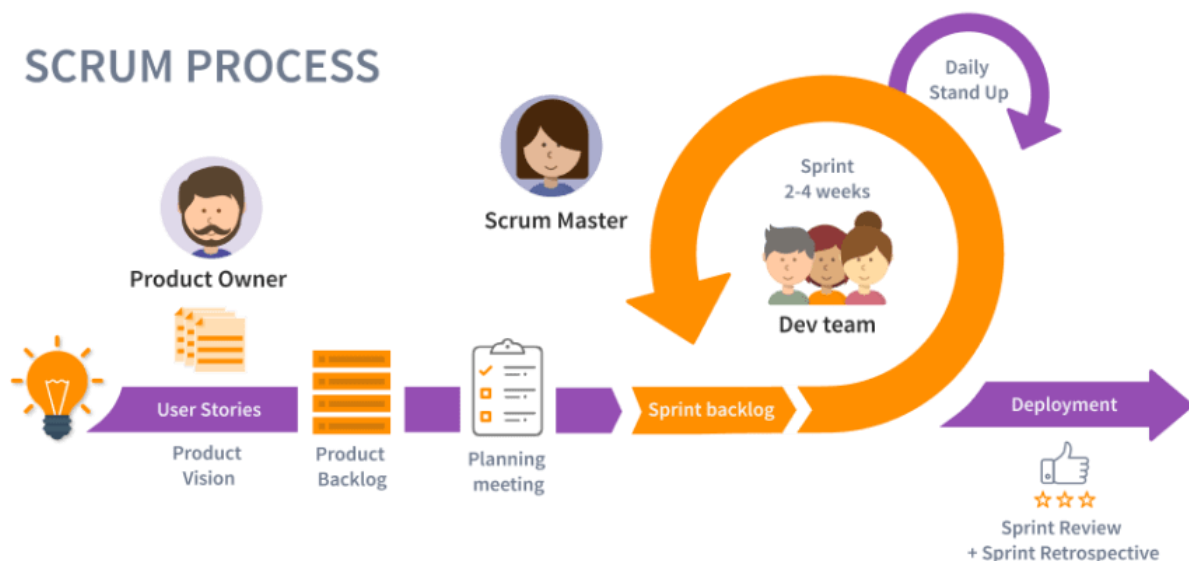
O ciclo do Scrum será estruturado em Sprints de duas semanas, cada um iniciando com um Sprint Planning para definição de metas, seguido por Daily Scrums para monitoramento do progresso, um Sprint Review para avaliação do que foi desenvolvido, e um Sprint Retrospective para identificar melhorias. Este processo facilitará a entrega contínua de valor, permitindo ajustes rápidos e eficazes ao aplicativo.

**Daily Scrum:** Reuniões diárias para sincronização do trabalho.

**Sprint Planning:** Planejamento das tarefas para o próximo Sprint.

**Sprint Review:** Avaliação do trabalho realizado.

**Sprint Retrospective:** Reflexão sobre o processo e adaptações necessárias.



## ARTEFATOS DO PRODUTO

### Atores

Usuário Final: Visitantes do espaço cultural interessados nas obras.

Administrador do Conteúdo: Responsáveis pela atualização e manutenção do conteúdo do aplicativo.

### Requisitos Funcionais

Cod	Requisito	Prioridade
RF01	O sistema deve permitir a visualização de informações detalhadas sobre as obras em português e inglês.	1
RF02	O sistema deve oferecer audioguias para cada obra, com opções em português e inglês.	1
RF03	O sistema deve ser acessível para usuários com deficiência visual, oferecendo leitura de tela e audioguias.	1
RF04	O sistema deve permitir que o administrador adicione, exclua e edite as obras.	1
RF05	O sistema deve ter uma barra de pesquisa para procurar as obras.	2
RF06	O sistema deve ter uma tela de login para administrador.	2
RF07	O sistema deve ter um botão na tela principal para ir para a tela de login do administrador.	2

### Requisitos Não-funcionais

Cod	Requisito	Prioridade
RNF01	O aplicativo deve ser capaz de suportar um crescimento significativo no número de usuários e de solicitações simultâneas sem degradar a qualidade do serviço.	1
RNF02	O código do aplicativo deve seguir padrões de codificação e documentação que facilitem atualizações, correções e a adição de novas funcionalidades com eficiência e eficácia.	3
RNF03	Design responsivo para adequação a diferentes tamanhos	3

Classificação para Prioridade dos requisitos:

1 - (Deve ter): São os requisitos essenciais e de alta prioridade, considerados fundamentais para o sucesso do

## ARTEFATOS DO PRODUTO

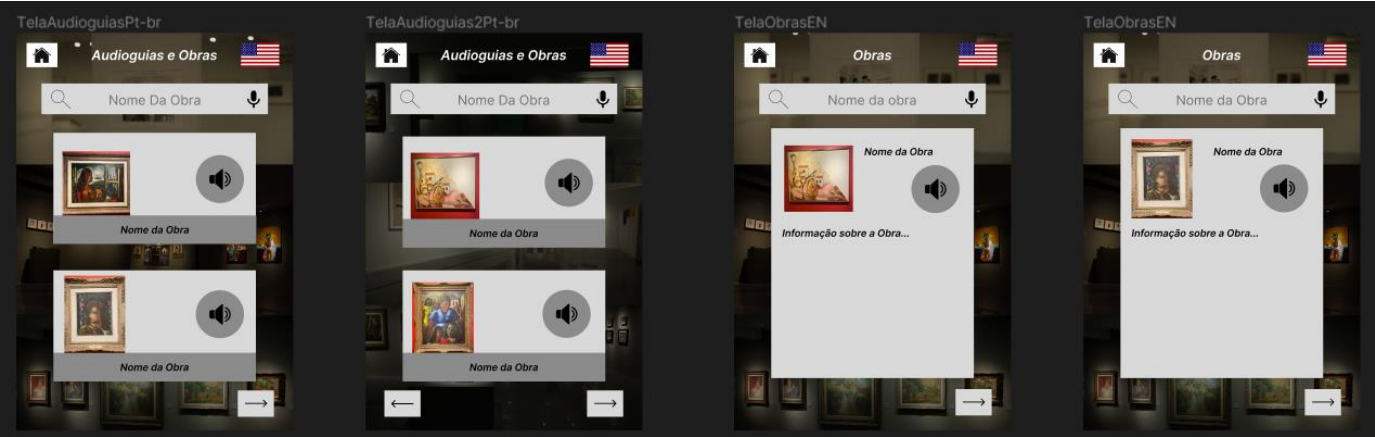
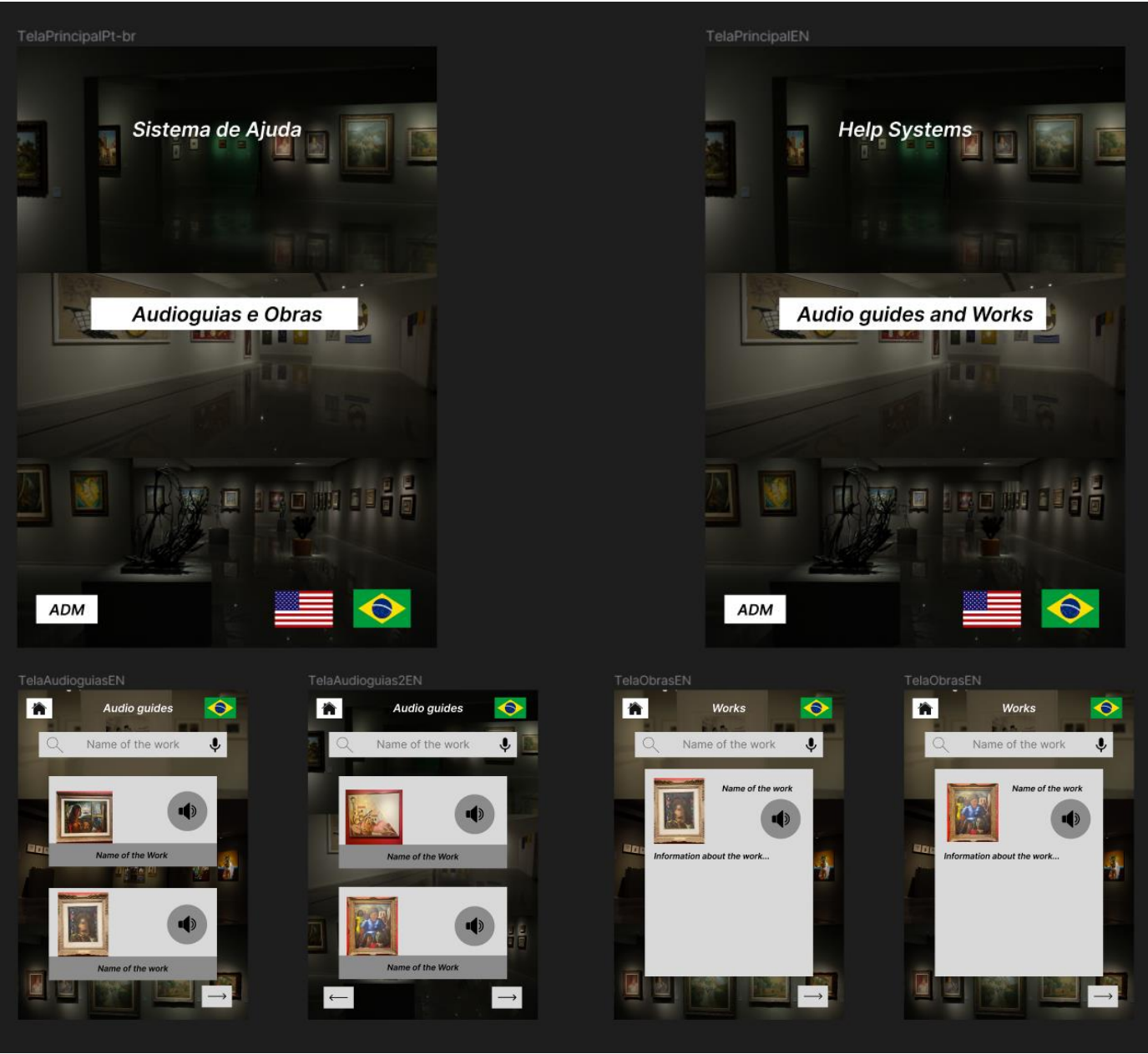
sistema. São os requisitos que devem ser implementados obrigatoriamente, pois sem eles o sistema não atenderia às necessidades básicas dos usuários finais ou não cumpriria os objetivos do projeto.

2 - (Deveria ter): São os requisitos importantes, mas não tão críticos quanto os de prioridade 1. Esses requisitos devem ser implementados se possível, mas podem ser adiados caso haja restrições de tempo ou recursos.

3 - (Poderia ter): São os requisitos desejáveis, mas não essenciais. São requisitos que podem trazer benefícios adicionais ao sistema, mas sua ausência não afetaria significativamente a funcionalidade principal.

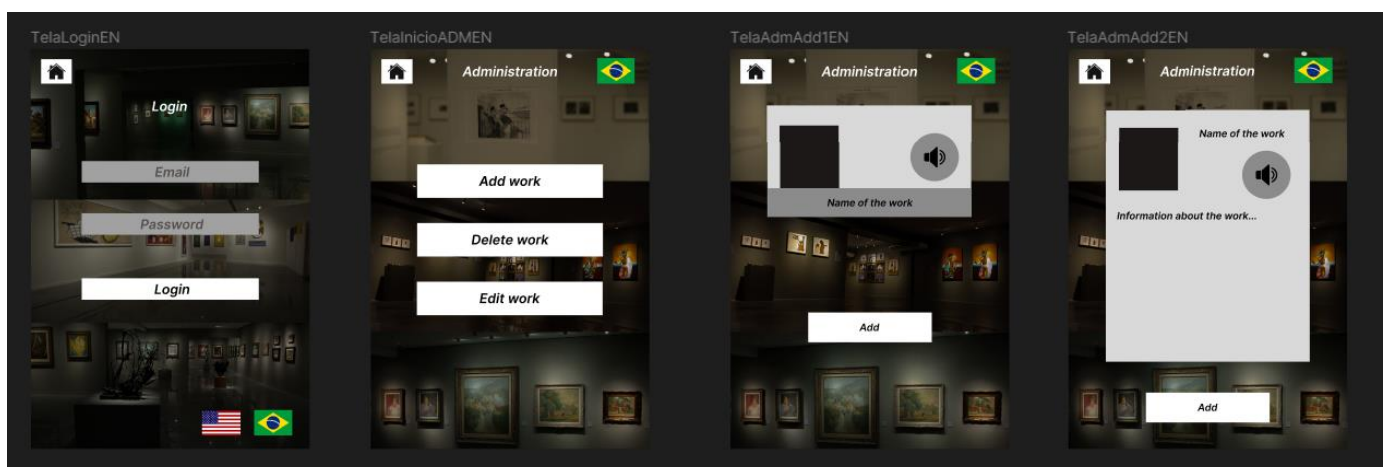
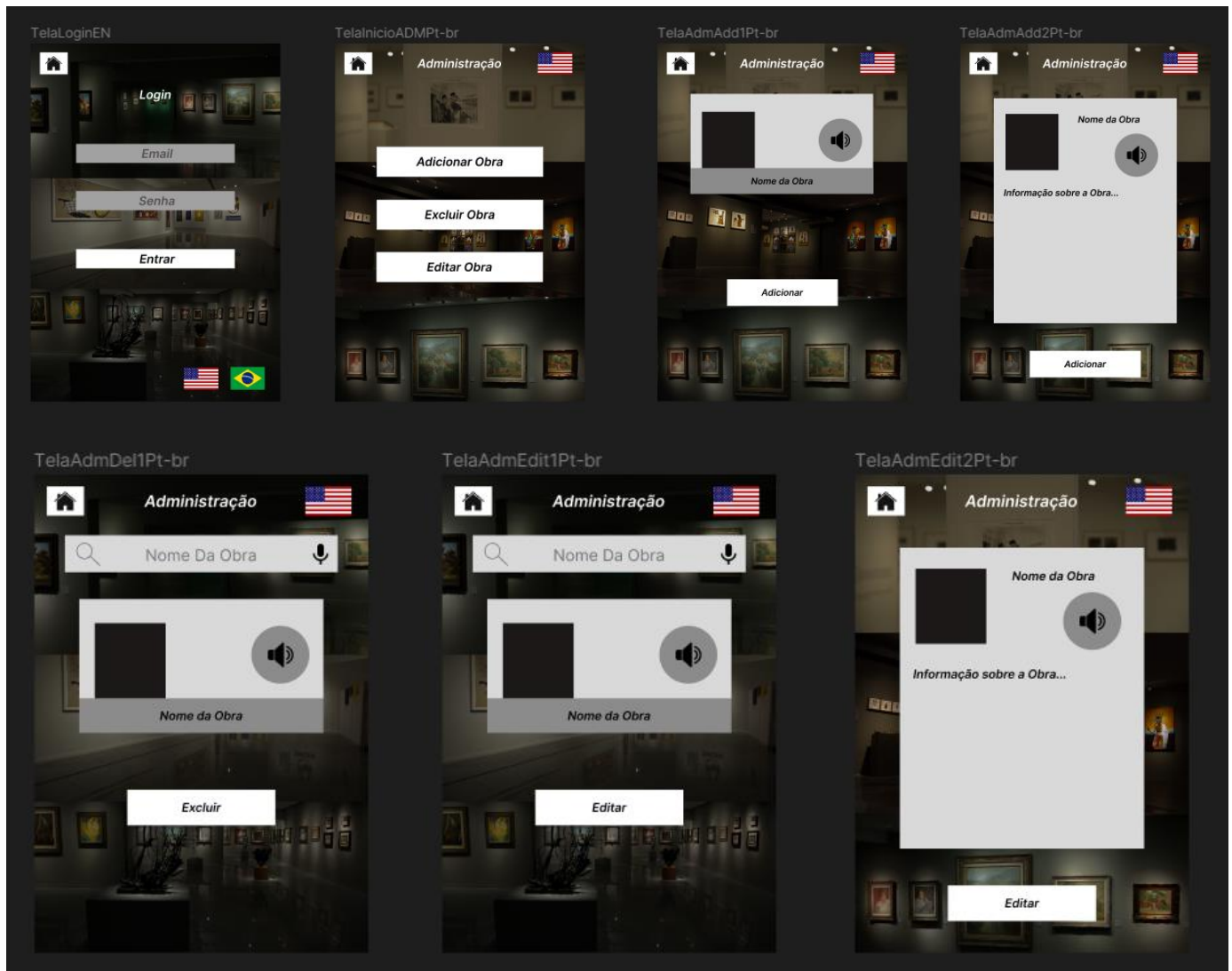
Protótipo de Baixa Fidelidade

Protótipo de Baixa Fidelidade





## Protótipo de Baixa Fidelidade



## Protótipo de Baixa Fidelidade

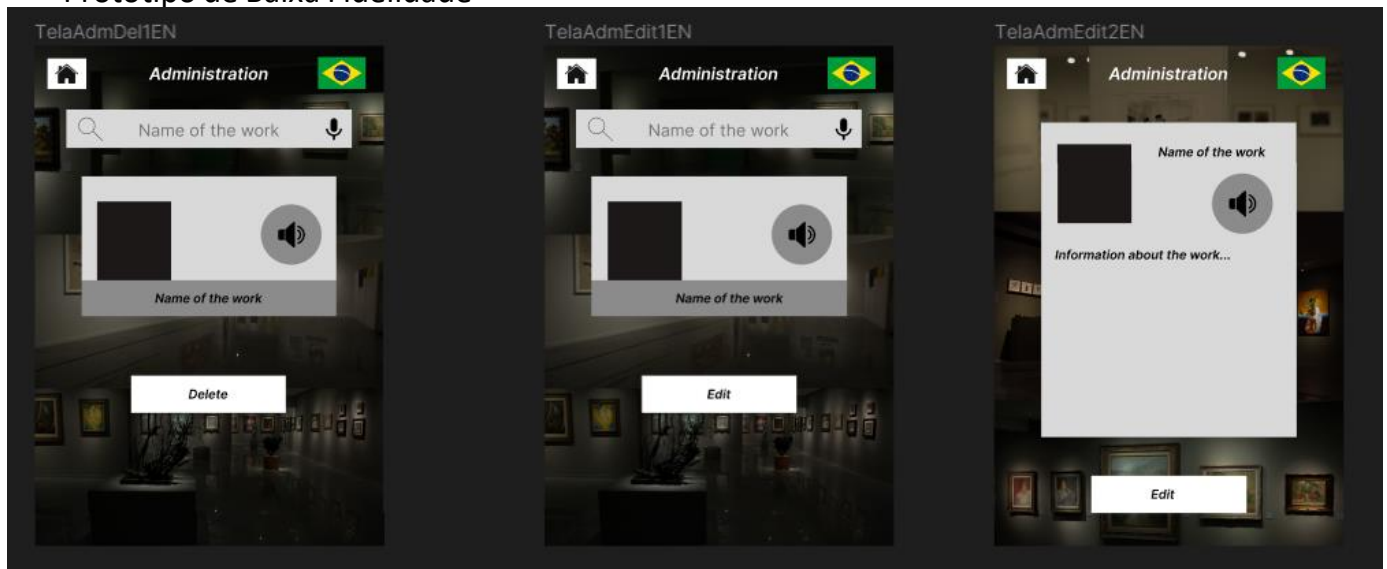


Diagrama de Caso de Uso

Diagrama de Caso de Uso

## Diagrama de Caso de Uso

### Diagrama de caso de uso

