

Como utilizar este Template

1. Faça uma cópia [File → Make a copy...]
2. Renomeie este arquivo para: **“Capstone_Stage1”**
3. Substitua todos os textos **em verde**

Instruções para Envio

1. Assim que completar todas as partes, faça o download deste documento como um PDF [File → Download as PDF]
 2. Crie um repositório novo no Github para o capstone. Dê o nome **“Capstone Project”**
 3. Adicione este documento no seu repositório. Tenha certeza que está nomeado como **“Capstone_Stage1.pdf”**
-

[Descrição](#)

[Público-Alvo/Intended User](#)

[Funcionalidades/Features](#)

[Protótipo de Interfaces do Usuário](#)

[Tela 1](#)

[Tela 2](#)

[Considerações Chave/Key Considerations](#)

[Como seu app vai tratar a persistência de dados?](#)

[Descreva qualquer caso de uso específico \(“corner case”\) da experiência do Usuário \(UX\).](#)

[Descreva quais bibliotecas você utilizará e compartilhe a razão de incluí-las.](#)

[Descreva como você implementará o Google Play Services.](#)

[Próximos Passos: Tarefas Necessárias](#)

[Tarefa 1: Configuração do Projeto/Project Setup](#)

[Tarefa 2: Implement UI for Each Activity and Fragment](#)

[Tarefa 3: Your Next Task](#)

[Tarefa 4: Your Next Task](#)

[Tarefa 5: Your Next Task](#)

Usuário do GitHub: [linekerm Souza](#)

WakeMeThere

Descrição

WakeMeThere é um aplicativo de alarme que se baseia na localização e desperta quando atingir o local definido.

Você pode tirar um cochilo em qualquer transporte público que não precisará se preocupar se passou do ponto, o aplicativo irá ajudá-lo. Apenas defina suas localizações - WakeMeThere irá cuidar do resto!

Público-Alvo/Intended User

Estudantes, trabalhadores, seus avós, você e qualquer pessoa que queira tirar um cochilo em transporte público sem se preocupar em perder o ponto de chegada. O aplicativo será disponibilizado para usuários Android.

Funcionalidades/Features

O aplicativo será desenvolvido utilizando a linguagem JAVA.

Principais funcionalidades do app:

- Listagem das localizações
- Adicionar nova localização
- Tocar o alarme quando chegar na localização definida
- Visualizar mapa para selecionar uma localização
- Permitir pesquisar no mapa e locais próximos

O aplicativo também dará suporte para acessibilidade para que todos possam utilizar o aplicativo sem nenhum problema.

Será utilizado AsyncTask para atividades que levar mais tempo assim evitará possíveis ANRs (Application Not Responding).

Protótipo de Interfaces do Usuário

Elas podem ser feitas a mão (tire uma foto dos seus desenhos e os insira neste fluxo), ou usando um programa como o Photoshop ou Balsamiq.

Tela 1

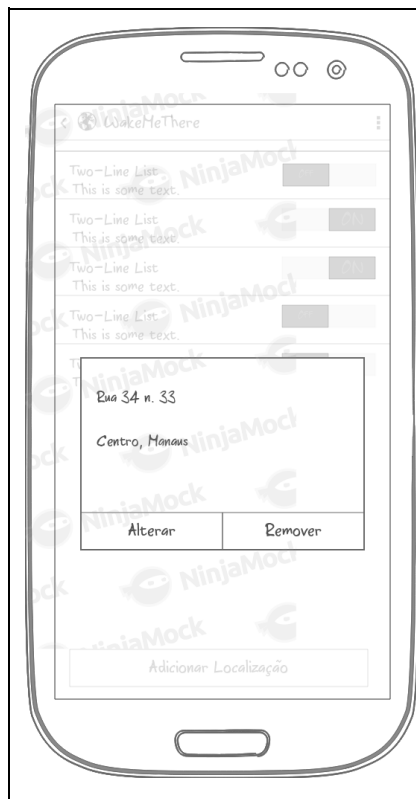


Lista as localizações definidas pelo usuário que as utiliza com frequência para ativar/desativar qual localização quer que o aplicativo o desperte.

Tela 2

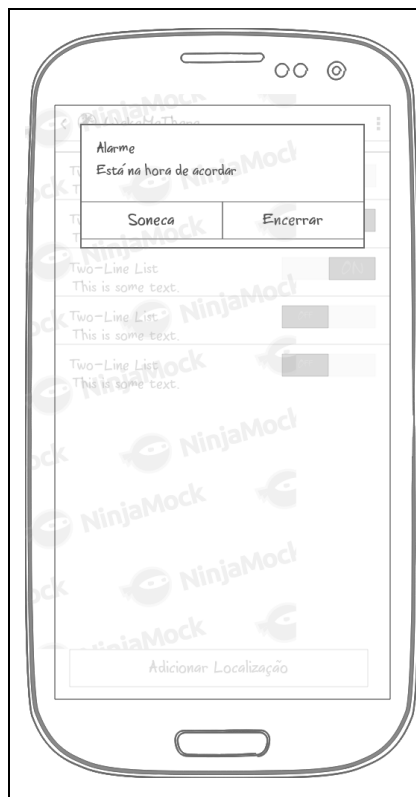


Tela 3



Ao pressionar sobre uma localização na lista será mostrado detalhes do que foi selecionado e exibido opção de remover ou alterar a localização.

Tela 4



O alarme será acionado automaticamente quando estiver próximo do local independente se o celular estiver bloqueado ou não.



Considerações Chave/Key Considerations

Como seu app vai tratar a persistência de dados?

Será criado um `ContentProvider` (com o `SQLite Database`) para acessar dados locais e também a utilização do `ContentResolver` para fazer as operações de query e insert ou através de `Loader`.

Descreva qualquer caso de uso específico (“corner case”) da experiência do Usuário (UX).

O usuário poderá adicionar uma localização clicando no botão de Adicionar abaixo da listagem, será encaminhado para a tela onde terá um mapa, onde ele poderá pesquisar ou escolher algum ponto do mapa e clicando em Selecionar essa Localização. Se pressionar o botão Voltar na tela onde o mapa é exibido, a operação é cancelada.

Em casos como o usuário não tenha dado permissão de localização ou problemas para recuperar informações de localização, o aplicativo informará o usuário sobre o problema para que ele possa tomar alguma ação.

O aplicativo terá suporte landscape e portrait mas não possuirá nenhum comportamento diferente.

Descreva quais bibliotecas você utilizará e compartilhe a razão de incluí-las.

Espresso para testes, recyclerview para listagem e appcompat para as activities. Será utilizado uma biblioteca de terceiro chamada SimpleAlarmManager para agendar os alarmes. As bibliotecas utilizadas farão uso de versões estáveis (release version)

Descreva como você implementará o Google Play Services.

Utilizarei as bibliotecas Places e Location do Google Play Services para identificar pontos de localização (latitude e longitude) onde o usuário atualmente se encontra e verificar se combina com alguma localização adicionada pelo usuário anteriormente.

Próximos Passos: Tarefas Necessárias

Esta é a parte onde você falará sobre as principais funcionalidades do seu app (mencionadas acima) e as dividirá em tarefas técnicas tangíveis que você pode completar de forma incremental até finalizar o app.

Tarefa 1: Configuração do Projeto/Project Setup

Sub Tarefas:

- Criação do projeto e configurações no gradle
- Configuração da chave de API no Google API Console
- Configurar bibliotecas

Tarefa 2: Implementar a Interface de Usuário (UI) para cada Activity e Fragment

Sub Tarefas:

- Construir a UI da Listagem de Localizações
- Construir a UI que utilizará o mapa para adição de localização
- Construir dialog de edição/remoção de localização

Tarefa 3: Configurar Google API Client

Sub Tarefas:

- Configurar metadados e permissões
- Adicionar dependências do Google Play Services
- Instanciar o Google API Client e implementar os callbacks

Tarefa 4: Tratar permissões de localização em tempo de execução

Sub Tarefas:

- Inicializar permissões
- Implementar callbacks para permissões exibidas para o usuário
- Tratar qual android sdk está sendo utilizado pelo usuário

Tarefa 5: Implementar PlacePicker

Sub Tarefas:

- Configurar PlacePicker
- Tratar exceções do GooglePlayServices
- Tratar callbacks de resultado
- Recuperar informação de local e salvar no banco

Tarefa 6: Tocar alarme quando estiver próximo do local definido

Sub Tarefas:

- Implementar permissões do Alarme
- Implementar callbacks de Alarme
- Utilizar callbacks de localização para executar o alarme
- Tocar o alarme

Instruções para Envio

4. Assim que completar todas as partes, faça o download deste documento como um PDF [File → Download as PDF]
5. Crie um repositório novo no Github para o capstone. Dê o nome “**Capstone Project**”
6. Adicione este documento no seu repositório. Tenha certeza que está nomeado como “**Capstone_Stage1.pdf**”

