

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: [henriquebk1](#)

Geo Coordenadas

Descrição

O Geo Coordenadas possibilita que o usuário capture coordenadas geográficas, mesmo em locais remotos, permitindo, ainda, a visualização de locais salvos em um mapa, além de realizar medições de distância entre eles.

O aplicativo, também, possui uma bússola integrada, facilitando a orientação do usuário em locais remotos.

O usuário, ainda, pode escolher o melhor formato de visualização das coordenadas e de distâncias.

Público-Alvo/Intended User

Viajantes, pessoas que trabalham ou viajam para fora das cidades e quem precisar de um serviço de geomapeamento, como agrimensores, funcionários de instituições financeiras e de registros de imóveis.

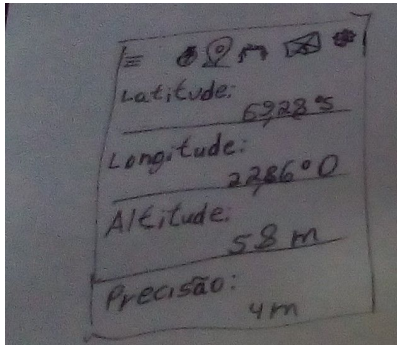
Funcionalidades/Features

- Exibe a localização atual
- Salva locais geográficos
- Exibe uma bússola virtual
- Mostra a distância entre dois locais
- Exibe um mapa com os locais do usuário

Protótipo de Interfaces do Usuário

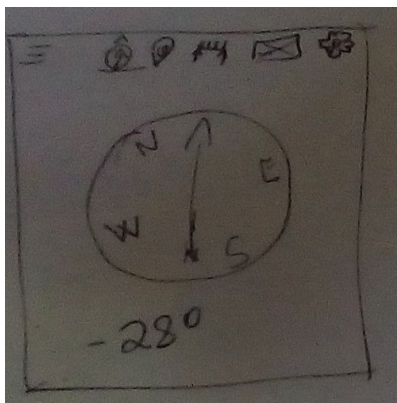
Elas podem ser feitas a mão (tire uma foto dos seus desenhos e os insira neste fluxo), ou usando um programa como o Photoshop ou Balsamiq.

Tela 1



Tela que exibe as informações de um local.

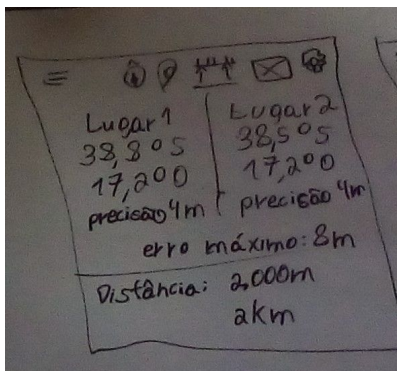
Tela 2



Tela da bússola, para o usuário se orientar em locais remotos.

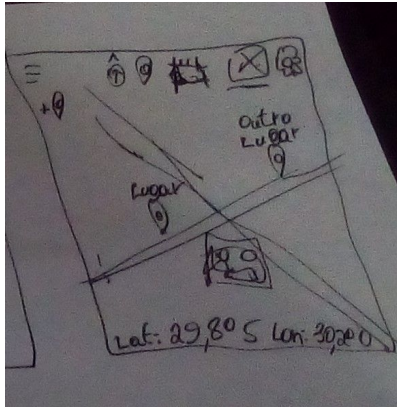
* Pode ser adaptada para mostrar a direção para um local.

Tela 3



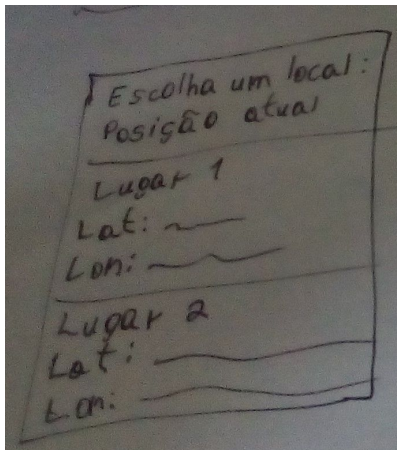
Mostra a distância entre dois locais.

Tela 4



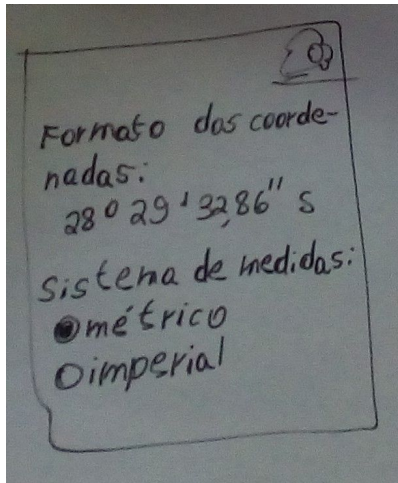
Visualização dos locais do usuário em um mapa

Tela 5



Permiti que o usuário escolha um local.

Tela 6



Tela de configurações do aplicativo, onde o usuário seleciona o modo de exibição das coordenadas e o sistema de medidas.

Adicione quantas telas achar necessário para demonstrar o fluxo de UI do seu app.

Considerações Chave/Key Considerations

Como seu app vai tratar a persistência de dados?

O app usará o Realm para armazenar os dados dos locais salvos pelo usuário. Um objeto referente a cada local (equivalente a um marcador do Google Maps) será armazenado localmente. Ele poderá ser atualizado ou excluído pelo usuário através do aplicativo. Além disso, penso em possibilitar que o usuário possa compartilhar um arquivo .kml (conteúdo locais salvos pelo usuário), permitindo a visualização futura dos locais em programas como o Google Earth.

*Talvez, conforme evoluir o desenvolvimento, posso adicionar a possibilidade, além de marcadores, formas como círculos e polígonos, que também poderão ser visualizados no mapa e exportados no arquivo .kml.

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

- Android annotations (<https://github.com/androidannotations/androidannotations/wiki>) para diminuir o tamanho do código que será criado.
- Realm (<https://realm.io>) para armazenar dados persistentes, escolhido por permitir um rápido desenvolvimento, junto de um bom desempenho.

Descreva como você implementará o Google Play Services.

A api de localização será utilizada para obter a localização geográfica do dispositivo. Já a api de mapas permitirá a exibição do mapa.

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

O projeto será criado com a “wizard” do Android Studio, que fará todo o trabalho com a estrutura do projeto.

As bibliotecas necessárias serão adicionadas no código do Gradle.

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

- Desenhar a tela de exibição de uma localização
- Criar o layout da exibição da bússola
- Criar o layout da tela de medidas
- Criar o layout da exibição do mapa
- Criar o layout da tela para seleção de uma localização
- Criar a activity de configurações

Tarefa 3: Implementar a tela de configuração do dispositivo.

- Adicionar as configurações do aplicativo usando SharedPreferences.
- Verificar se as configurações são corretamente aplicadas a todas as telas do aplicativo.

Tarefa 4: Implementar a api de localização

Implementar a api de localização do Google Play Services, para obter a localização atual.

- Popular o layout da tela de exibição de uma localização com a atual
- Criar alertas para o caso de falhas (ex.: gps desligado)

Tarefa 5: Implementar o banco de dados

Implementar o Realm no projeto, permitindo o salvamento de locais.

- Criar os objetos Realm que serão armazenados.
- Testar o banco de dados.

Tarefa 6: Implementar a bússola

Criar a exibição da bússola no aplicativo

- Obter o ângulo do norte magnético através dos sensores do dispositivo.
- Implementar a rotação da bússola através da mudança do valor lido.

Tarefa 7: Implementar a seleção de um local.

Permitir que o usuário selecione um local

- Implementar a busca pelos locais salvos no armazenamento interno.
- Implementar a criação de um local através da digitação das coordenadas.

Tarefa 8: Implementar a tela de medição de distância entre locais.

- Implementar o cálculo e visualização da distância entre dois locais.

Tarefa 9: Implementar a Maps api.

- Adicionar a exibição do mapa.
- Permitir a adição de locais ao mapa.
- Tornar o mapa interativo.

Tarefa 10: Verificar responsividade

- Verificar o comportamento do aplicativo em diferentes orientações de tela, fazendo os ajustes necessários.
- Verificar o funcionamento do aplicativo em diferentes tamanhos de tela, fazendo os ajustes necessários.

Tarefa 11: Realizar testes unitários

- Realizar diversos testes unitários, com o intuito de eliminar possíveis bugs.