

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: mso2000

Histórias Infantis - Abobrinha Studios

Descrição

O aplicativo reunirá as melhores histórias infantis e originais do blog Histórias Infantis - Abobrinha Studios (<https://historiasinfantisabobrinha.com/>) numa interface moderna e otimizada tanto para celulares quanto tablets para divulgar o trabalho do cliente e permitir uma incrível experiência mobile para os usuários como leitura off-line, salvar as histórias preferidas numa área de favoritos, criação de widgets na homescreen e recebimento de notificações para manter o usuário sempre atualizado quando novas histórias forem lançadas.

Público-Alvo/Intended User

O aplicativo é destinado para crianças e pais que irão desfrutar momentos de entretenimento em família.

Funcionalidades/Features

- Obter todas as histórias publicadas no blog Histórias Infantis – Abobrinha Studios em formato JSON usando a WordPress API e persistir os dados localmente com um Content Provider para permitir a leitura off-line;
- O aplicativo irá esporadicamente checar o servidor em background por novas histórias e avisar o usuário com uma notificação que irá abrir a história recém obtida;
- O usuário poderá marcar as histórias que mais gostou como “favoritas” e estas serão exibidas numa aba própria da tela principal;
- A tela principal irá conter um grid com as imagens e títulos das histórias e opção para o usuário manualmente fazer um refresh das histórias do servidor;
- A tela principal também irá conter um botão que abrirá um Dialog com links para as diversas redes sociais e e-mail do cliente (Abobrinha Studios);
- A história selecionada será visualizada em outra tela que irá apresentar o título, a história, imagens e opções para salvá-la como Favorita, compartilhar com um amigo ou acessar o post original no blog;
- A tela com a história também permitirá navegar com scroll horizontal para a próxima história (ou anterior) e haverá um Navigation Drawer com todas as histórias para facilitar a navegação do usuário que não precisará retornar para a tela principal;
- Em todas as telas haverá um menu para acessar a página de configurações do aplicativo, que seriam:
 - Habilitar/Desabilitar notificações;
 - Apagar todos os favoritos.
- O aplicativo oferecerá layout otimizado para tablets e contará com efeitos de transições e animações do Material Design;

- O aplicativo será otimizado para acessibilidade
- O aplicativo contará com 2 tipos de Widgets configuráveis:
 - Ícone com a história selecionada que irá abrir a história no aplicativo;
 - Lista com todas as histórias ou apenas as favoritas.
- O aplicativo também poderá receber notificações esporádicas do Firebase Cloud Messaging;
- O aplicativo será disponibilizado em 2 flavors: versão free (com anúncios Admob, tanto em formato de banner quanto interstitial) e paga (sem anúncios)

Protótipo de Interfaces do Usuário

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

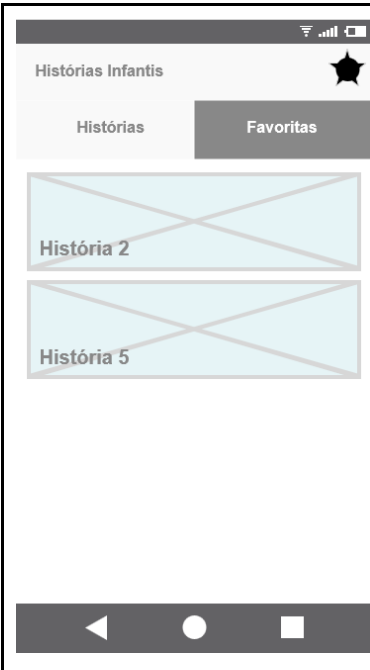
Tela 1



Tela principal do aplicativo quando aberta em celulares.

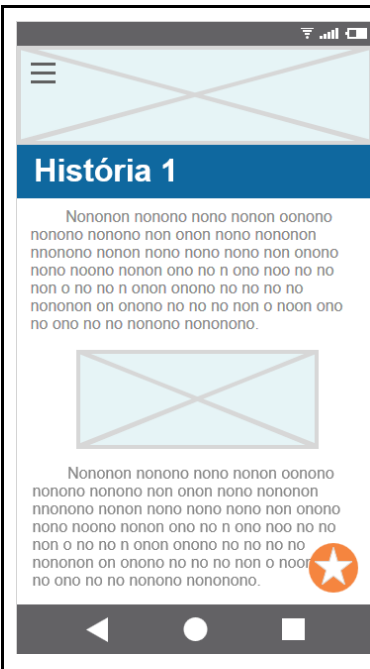
Na Toolbar existe um ícone para acesso às Configurações. Logo abaixo estão as abas para visualização de todas as histórias ou as marcadas como “favoritas” pelo usuário. Por fim, serão exibidas as imagens e títulos das histórias numa listagem rolável.

Tela 2



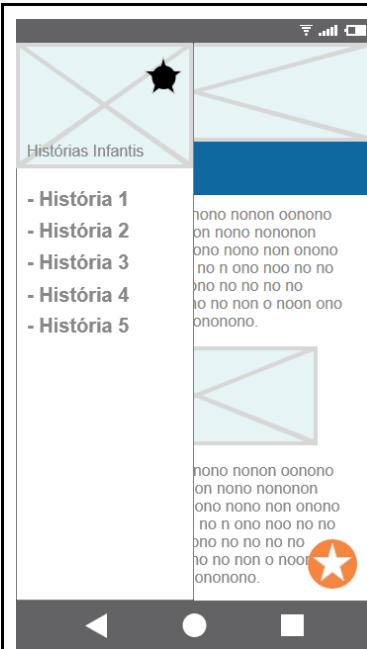
Tela principal do aplicativo, quando aberta em celulares, exibindo as histórias favoritas. O usuário poderá remover histórias desta tela deslizando as mesmas para fora da tela.

Tela 3



Tela com a história completa quando aberta em celulares. No início temos a imagem principal da história (mesma da tela principal). Em seguida temos o título e a história propriamente que pode conter outras imagens. Nesta tela temos um FAB que permitirá marcar ou desmarcar a história como “Favorita”. No final do texto da história serão exibidos links para acessar o post original no blog ou compartilhá-lo com amigos. Por fim, esta tela irá contar com um Navigation Drawer (mais detalhes na próxima imagem).

Tela 4

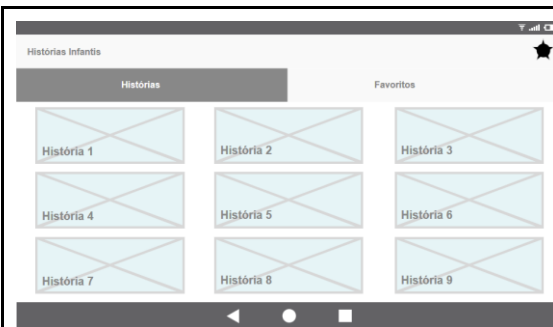


Tela com a história completa quando aberta em celulares exibindo um Navigation Drawer que irá exibir a lista de histórias (que poderão ser todas ou apenas as favoritas dependendo de qual aba estava selecionada na tela principal).

A idéia é facilitar a navegação do usuário pelas histórias sem a necessidade de retorno constante à tela principal. Além disso, o usuário também poderá acessar a próxima história deslizando a tela horizontalmente.

O Navigation Drawer também irá oferecer um botão para acessar a tela de Configurações do aplicativo.

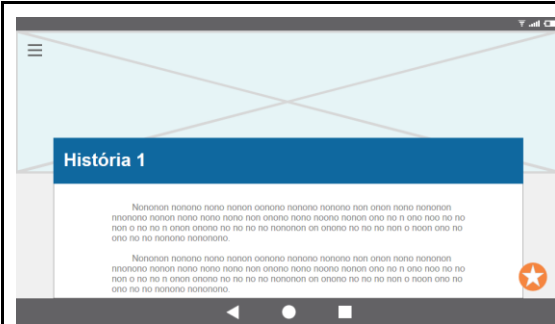
Tela 5



Tela principal do aplicativo quando aberta em tablets.

Funcionalmente será como a dos celulares, mas com layout em formato de grid para melhor visualização

Tela 6



Tela com a história completa quando aberta em tablets.

Funcionalmente será como a dos celulares, mas com layout otimizado para exibição em telas grandes.

Tela 7

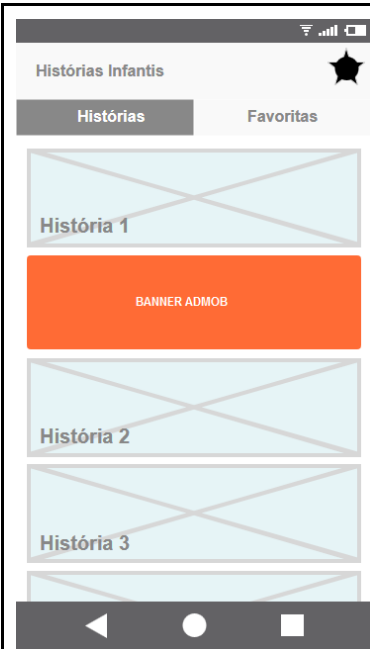


Exemplos de widgets na homescreen

O primeiro seria um widget com lista de histórias (todas ou favoritas) de acordo com a escolha do usuário ao adicionar o widget. Ao clicar numa história da lista, a mesma é aberta para leitura no app.

Os demais são widgets simples com a história selecionada pelo usuário ao adicionar o widget. Exibem a imagem e título da história. Ao serem clicados, abrem a história correspondente no app.

Tela 8



Versão free do app que exibe banners Admob aleatórios na lista de histórias. Também serão exibidos aleatoriamente banners interstitial antes de uma história ser carregada na tela secundária.

Tela 9



Exemplo de notificações do app. Podem ser tanto notificações originadas do Firebase Cloud Messaging com mensagens do cliente ou originadas pelo próprio app em background quando novas histórias são detectadas no blog. Neste caso, a história é aberta para leitura quando a notificação é clicada.

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 aplicativo irá implementar um Content Provider para armazenar os dados JSON obtidos da WordPress API que será sincronizado esporadicamente em background com tarefas agendadas (por ex: 1 vez por dia) ou manualmente pelo usuário na tela principal. As imagens carregadas serão obtidas e cacheadas com o uso de libraries como o Picasso ou Glide. Também serão utilizados SharedPreferences para guardas as diversas configurações do aplicativo e outras variáveis de ambiente.

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

O aplicativo irá conter diversos elementos de transições de tela e animações de views para tornar a experiência mais divertida para o público alvo, essencialmente crianças. Além disso, a criação de widgets na homescreen torna a experiência do usuário com o aplicativo ainda melhor.

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

Será utilizado o Picasso ou Glide para tratar o carregamento e cache das imagens e a biblioteca de Admob para exibição de anúncios diversos na versão free do aplicativo.

Descreva como você implementará o Google Play Services.

O aplicativo será integrado com o Firebase Cloud Messaging que permitirá ao cliente (Abobrinha Studios) enviar notificações diversas com promoções, novidades no blog, etc., para permitir uma maior aproximação com o público alvo.

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

- Configurar bibliotecas (Support, Design, RecyclerView, Constraint Layout, Admob, Picasso ou Glide).

Tarefa 2: Implementar a Interface de Usuário (UI) inicial para cada Activity e implementar acesso à API

- Tela principal (MainActivity)
 - Construir a UI com uma RecyclerView simples (apenas TextViews);
 - Implementar rotinas para fazer a query na WordPress API e o parsing no JSON com os resultados;
 - Carregar a RecyclerView com os títulos das histórias e implementar interface para abertura da tela secundária com a história selecionada.
- Tela secundária (HistoriaActivity)
 - Obter dados da história selecionada na MainActivity;
 - Exibir demais dados da história (Texto e URL das imagens).

Tarefa 3: Implementar Content Provider e Serviços para sincronização dos dados

- Content Provider
 - Definir o Contract (URI's e tabelas necessárias do BD);
 - Definir o Helper do DB;
 - Implementar o matcher e métodos necessários do Provider (select, insert, delete, update);
 - Configurar o ContentProvider no AndroidManifest.xml;
 - Implementar testes diversos (Espresso) com o Provider.

- Sync Service
 - Definir estratégia de sincronização dos dados;
 - Implementar os mecanismos necessários para executar a sincronização:
 - IntentService;
 - Tarefas sincronizadas;
 - JobService / JobDispatcher;
 - Notificações.
 - Implementar testes diversos (Espresso) com a estratégia de sincronização definida.
- Tela Principal (MainActivity)
 - Inicializar o serviço de sincronia de dados;
 - Implementar Loader para carregar dados do Content Provider;
 - Ajustar RecyclerView para ler dados do Cursor e enviar a URI da história selecionada;
 - Implementar um listener para atualizar a UI com os refreshs manuais do usuário.
- Tela Secundária (HistoriaActivity)
 - Obter a URI da história selecionada na tela principal;
 - Implementar Loader para carregar dados do Content Provider usando a URI obtida.

Tarefa 4: Polir a interface

- Tela Principal (MainActivity)
 - Implementar ViewPager e TabLayout com 2 abas para exibição de todas as histórias ou apenas as favoritas;
 - Criar Fragment para ser utilizado no ViewPager e mover o código da MainActivity que monta o grid e ajustar para usar a URI apropriada (todas as histórias ou apenas favoritas);
 - Na aba de “Favoritos” implementar o deslize horizontal de itens na RecyclerView para remover a história dos favoritos.
- Tela Secundária (HistoriaActivity)
 - Implementar FAB para marcação/desmarcação de Favoritos;
 - Implementar Navigation Drawer e carregar lista de histórias conforme a última aba selecionada pelo usuário na tela principal.

- Implementar Transições e animações
 - Shared Element Transitions (imagem do grid para a imagem inicial da tela secundária);
 - Interpolator animations (título e corpo da história deslizando de fora da tela para suas posições. O mesmo para os itens do grid na tela principal);
 - View animations (FAB button, refresh button).
- Criar tela de configurações
 - Recursos XML com as preferências necessárias
 - (Checkbox) Ativar notificações;
 - Excluir favoritos.
 - Activity/Fragment que estende PreferenceFragmentCompat;
 - Ajustar sumários das preferências de acordo com os ajustes feitos pelo usuário;
 - Ajustar serviços de sincronização de acordo com a opção configurada pelo usuário.

Tarefa 5: Widgets

- Widget simples
 - Criar Activity de setup com a lista de histórias para o usuário selecionar uma história;
 - Exibir imagem e título da história selecionada que irá abrir a tela secundária do app passando a URI da história selecionada.
- Widget com lista
 - Criar Activity de setup com opção para exibir todas as histórias ou apenas as favoritas;
 - Exibir lista de histórias de acordo com a seleção feita no setup;
 - Cada item da lista irá abrir a tela secundária do app passando a URI da história selecionada.

Tarefa 6: Flavors e FCM

- Firebase Cloud Messaging
 - Criar conta e projeto no Firebase;
 - Adicionar JSON do Firebase no projeto;
 - Adicionar dependências no Gradle;
 - Enviar mensagens de teste pelo console do Firebase;
 - Configurar Custom Data no console do Firebase;
 - Ajustar MainActivity para obter o extra do intent recebido;
 - Realizar novos testes.

- Flavors
 - Criar novo flavor para a versão free do app no Gradle;
 - Criar novos layouts para a versão free da tela principal que irão incluir banners Admob;
 - Criar novas Activities para a versão free que irão invocar anúncios interstitial aleatoriamente antes de carregar uma história.

Adicione quantas tarefas que achar necessárias para concluir seu app.

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**”