

**Usuário do GitHub:** [alexander.dorow](#)

# SUBReddit

## Descrição

O Reddit te dá o melhor da internet em um único lugar, com notícias, histórias divertidas, fotos, memes, vídeos e muito mais, então que tal melhorar toda essa experiência juntando todos os seus subreddits em um único lugar, o SUBReddit te ajuda com isso! Organizando todos os seus subs que deseja seguir em uma única lista.

## Público-Alvo/Intended User

Usuários do reddit.

## Funcionalidades/Features

- Listar subreddits agregados em uma única lista
- Pesquisar novos subreddits
- Adicionar novos subreddits à lista
- Remover subreddits da lista
- Visualizar os tópicos

## Protótipo de Interfaces do Usuário

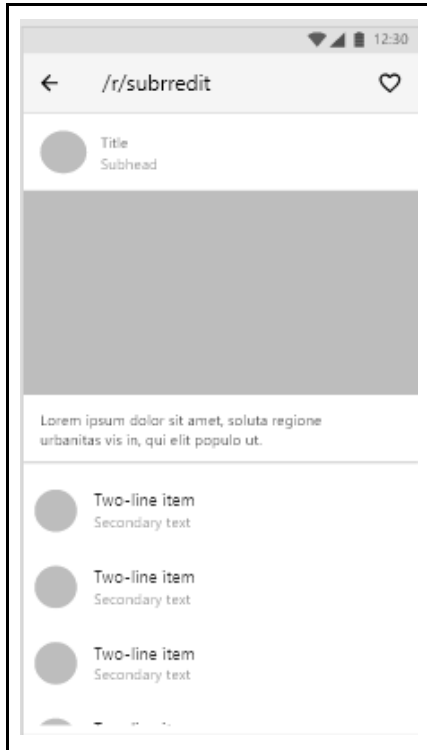
## Tela principal

	<p>Tela principal do app responsável pela listagem dos subreddits adicionados pelo usuário</p>
--	--

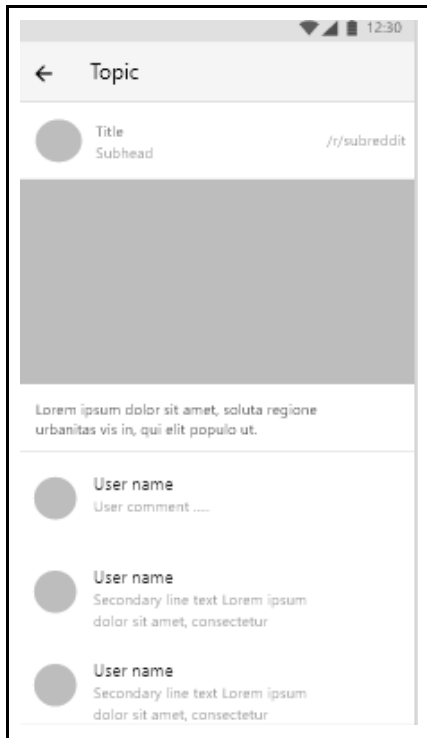
## Tela de pesquisa

	<p>Tela de pesquisa do subreddit responsável por listar todos os subreddits de acordo com a pesquisa do usuário</p>
--	---

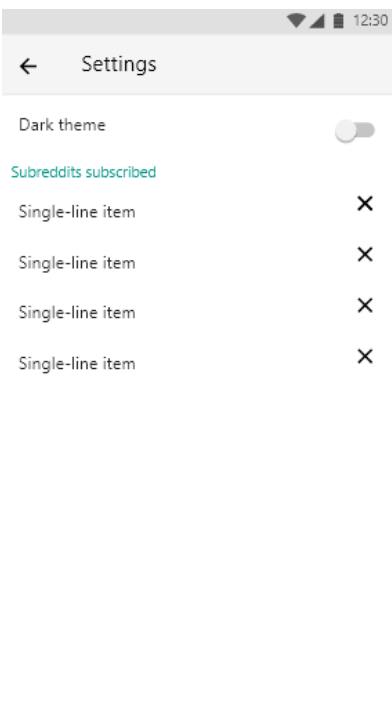
## Tela do subreddit

	<p>Tela do subreddit pesquisado, aonde irá mostrar todos os tópicos de acordo com o mesmo.</p>
--	--

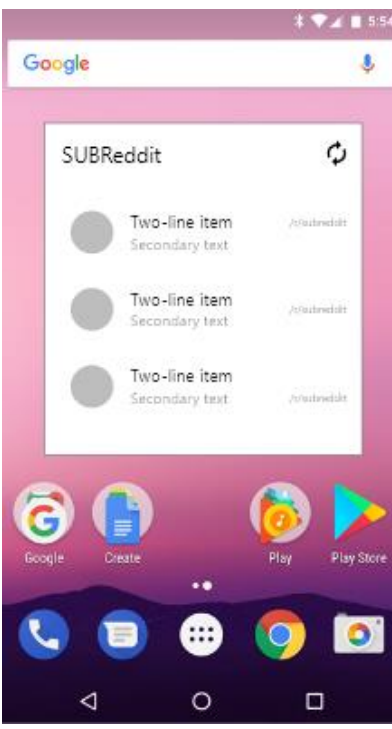
## Tela do tópico

	<p>Tela do tópico aonde irá mostrar com detalhes tudo relacionado ao tópico, permitindo a reprodução de um video, gif ou até mesmo a visualização de uma imagem e comentários.</p>
---	--

## Tela de configurações

	<p>Tela de configurações aonde o usuário poderá se desenscrever de um subreddit ou ativar o modo escuro</p>
---	---

## Widget

	<p>O widget será responsável por dar um preview dos tópicos atualizados, sendo disponibilizado um botão para permitir a atualização dos mesmos.</p>
---	---

## Considerações Chave/Key Considerations

### Como seu app vai tratar a persistência de dados?

O App irá implementar o Retrofit que será responsável por toda a comunicação para puxar os dados JSON da api REST do reddit, após isso, o json será convertido com o GSON para um objeto entity do RoomDB e salvo com os dados da biblioteca dentro do banco SQLite da aplicação. Dados mais básicos como comentários serão puxados em real time utilizando o retrofit e cache de imagens ficarão por conta do Glide.

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

Android support Library: compatibilidade retroativa de novos recursos.

Glide: tratativa de imagens

Dagger 2: injeção de dependências

RoomDB: Persistencia do banco de dados

Retrofit: Comunicação para endpoints REST

Gson: Conversão de dados JSON para Objetos

Mockito, Junit e espresso: automatizar os testes da aplicação

### Descreva como você implementará o Google Play Services.

Na aplicação será utilizado dois serviços do Google Play Services

AdMob: Que será responsável pela versão free do aplicativo permitindo a demonstração de anuncios.

Analytics: Que será responsável por demonstrar como os usuários estão usando a aplicação

## Próximos Passos: Tarefas Necessárias

### Tarefa 1: Configuração do Projeto/Project Setup

- Configurar bibliotecas
- Configurar estrutura MVP

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

- Construir a UI da MainActivity
- Construir a UI do SubredditListFragment
- Construir a UI do SubredditSearchFragment
- Construir a UI da SettingsActivity
- Construir a UI do TopicFragment
- Construir a UI do SubredditFragment

### Tarefa 3: Tratar casos de erro

- Tratar casos de erro
- Apresentar erro de forma intuitiva para o usuário

### Tarefa 4: Variações de ambientes

- Criar versão gratuita
- Criar versão paga

### Tarefa 5: Testes

- Criar testes unitários para a aplicação
- Criar testes instrumentados para aplicação