

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: github.com/dritoferro

Controle de Posto

Descrição

Com o app Controle de Posto, você pode controlar separadamente os gastos de combustível para cada veículo que você possua, você tem autonomia total sobre o aplicativo ficando por dentro de quanto gastou e quanto rodou em cada mês e por ano! Ainda terá a possibilidade de compartilhar o preço de um posto com seus amigos via redes sociais para que todos saiam ganhando.

Público-Alvo/Intended User

Este aplicativo destina-se a qualquer pessoa que possua um veículo e que deseja ter um controle preciso sobre seus gastos com combustível tendo a opção de receber notificações de preços baixos e o compartilhamento com seu grupo de amigos.


Funcionalidades/Features

- Salvar informações de cada veículo individualmente tendo a opção de personalizar o ícone com uma foto de acordo com o gosto do usuário
- Compartilhamento de valores de combustível
- Geração de gráficos
- Obter a localização do posto

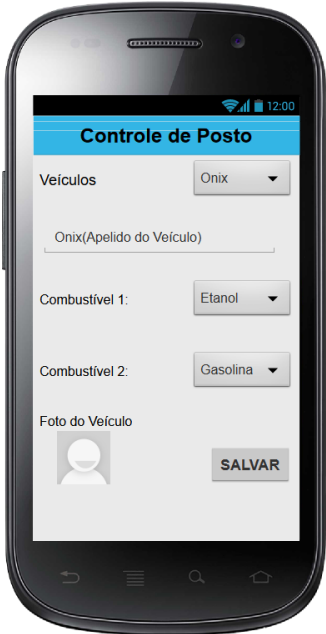
Protótipo de Interfaces do Usuário

Os protótipos de tela foram feitos utilizando o software Pencil.

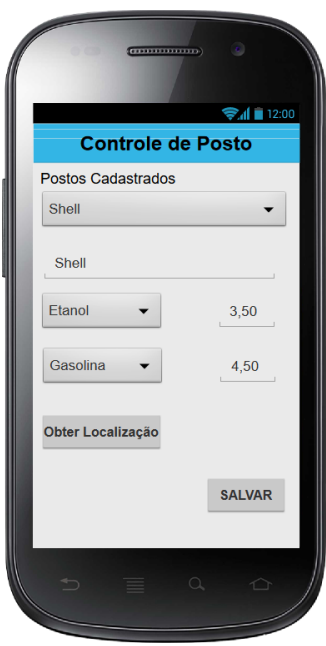
Tela 1

	<p>Esta será a tela principal do aplicativo, na qual o usuário tem acesso a uma lista de abastecimentos realizadas para cada veículo que possuir. Cada item da lista possui as seguintes informações: Data do abastecimento, o nome do posto, a km atual do veículo no momento da abastecida, e a quantidade de quilômetros rodados desde a última abastecida.</p>
--	--


Tela 2

	<p>Nesta tela é possível realizar o cadastro de um veículo para poder realizar o controle de abastecimentos. O usuário deve colocar um nome (um apelido) para o veículo e escolher no mínimo um combustível para seu veículo, tendo a opção de colocar uma foto do mesmo para facilitar a identificação.</p>
---	--

Tela 3

	<p>Nesta tela é possível realizar o cadastro de um posto tendo a opção de incluir o nome do posto, os valores de até dois combustíveis diferentes e obter a localização do posto para que você receba notificações sobre futuras quedas de preço.</p>
---	---

Tela 4

	<p>Após realizar o cadastro de um posto e um veículo, o usuário poderá realizar o cadastro de um novo abastecimento, tendo que informar o dia, veículo abastecido, a km atual no momento, e o posto ao qual abasteceu o veículo. Ainda há a possibilidade de atualizar o preço do litro do combustível na hora da abastecida.</p>
---	---

Considerações Chave/Key Considerations

Como seu app vai tratar a persistência de dados?

Os dados serão salvos em um banco de dados localmente, SQLite. Será feito um ContentProvider para realizar a manipulação dos dados. Também haverá uma conexão com o Firebase para que sejam enviadas informações sobre os postos e com isso enviar notificações para os usuários sobre o preço do combustível e também a localização do posto.

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

- Se o usuário estiver realizando qualquer processo de cadastro e/ou alteração e pressionar o botão de voltar, será exibida uma mensagem se o mesmo deseja salvar os dados antes de sair da tela.
- Quando o usuário estiver realizando o cadastro de um novo abastecimento e o posto não estiver na lista, o mesmo pode ser adicionado selecionando no menu lateral e após isso quando clicar em retornar, a tela de cadastro de abastecimento será exibida novamente.

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

Serão utilizadas as seguintes bibliotecas:

- ✓ Glide: para carregamento e cache de imagens dos veículos.
- ✓ Firebase Realtime Database: para compartilhar informações sobre os postos e seus valores e localização.
- ✓ Firebase Authentication: para realizar a autenticação dos usuários para o envio de informações para o Firebase Database.
- ✓ Firebase UI: para realizar a criação das telas de autenticação do usuário.
- ✓ Firebase Notifications: para receber notificações sobre postos com preços baixos.
- ✓ RecyclerView: Será utilizado para listar os abastecimentos de cada veículo.
- ✓ MPAndroidChart: Será utilizada para a geração dos gráficos sobre o gasto de combustível.

Descreva como você implementará o Google Play Services.

Serão utilizados o serviço de AdMob para publicidades e o serviço de Location para obter as informações precisas sobre o posto de combustível.

Próximos Passos: Tarefas Necessárias

Tarefa 1: Configuração do Projeto/Project Setup

- Configurar bibliotecas e suas respectivas versões
- Configurar versão mínima suportada do app.

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

- Construir a UI da MainActivity que será a tela que contém dois fragments para a lista dos abastecimentos e os veículos que o usuário possui.
- Construir a UI para o cadastro e alteração dos dados do veículo.
- Construir a UI para manter as informações de cada posto que o usuário já utilizou.
- Construir a UI para a geração de gráficos sobre o abastecimento com opções de tipo de gráfico.
- Construir a UI para exibir as notificações sobre promoção de postos.

Tarefa 3: Criação das classes de Modelagem e Acesso aos Dados via ContentProviders

Neste momento serão feitas as classes para a construção e conexão ao banco de dados local e posteriormente a criação de um content provider para conectar-se às informações.

- Criação do banco de dados, utilizando SQL Open Helper.
- Criação das classes de Contrato e dos métodos de um CRUD.
- Criação da classe de implementação do ContentProvider.

Tarefa 4: Criação das classes sobre Regras de Negócio

Nesta tarefa serão codificadas as classes para lidar com as regras de negócio do aplicativo bem como a validação dos dados que serão exibidos na UI e que serão salvos no banco de dados.

- Criação de uma classe genérica para lidar com as validações, de modo que a mesma possa ser reaproveitada em projetos futuros.

- Criação das classes para realizar a comunicação e validação entre a interface gráfica e o banco de dados, também gerenciando os dados que irão para o Firebase Realtime Database.

Tarefa 5: Criação das classes para lidar com o Google Play Services - AdMob e Location

- Criação da classe para exibir os anúncios no aplicativo via AdMob.
- Criação da classe para obter a localização do usuário via Location para ser utilizada na divulgação de preços baixos de combustível.

Tarefa 6: Criação das classes para comunicação com o Firebase Database e notificação de preços baixos

- Criação e modelagem das classes para conectar ao Firebase Database
- Obtenção dos dados usando o ContentProvider previamente criado e enviar os mesmos para a Firebase Database.
- Criação da classe para gerenciar as notificações que serão exibidas de acordo com os postos que o cliente utiliza.

Tarefa 7: Criação e configuração do envio de notificações no painel do Firebase Notifications

- Configurar o Firebase Notifications para enviar uma notificação para os usuários sobre preços baixos em determinado posto.