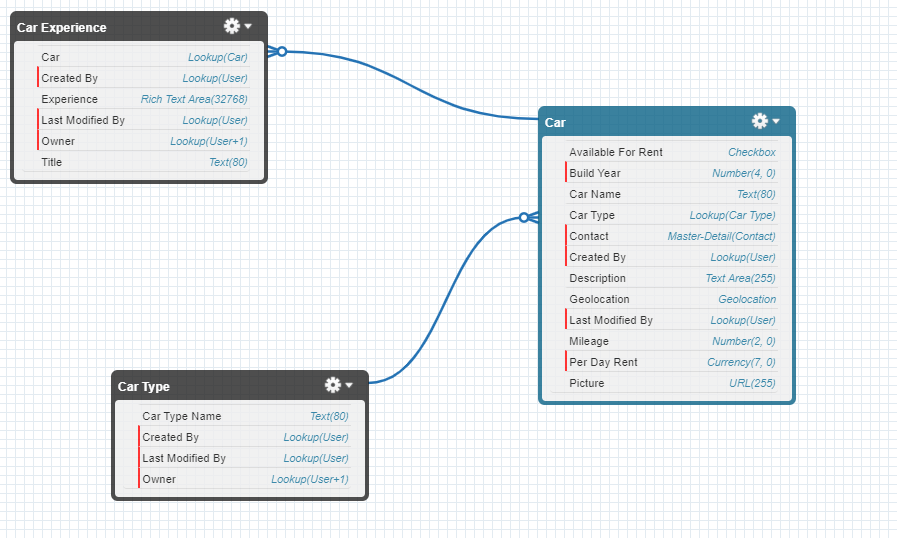
**Projeto Academia LWC – Car Rental**



**Diagrama de Entidade e Relacionamento Proposto:**



**Epic Story 001: Fluxo de Aluguel de Carros**

* **User Story 001: Preparação de ambiente**
* **User Story 002: Construção de App Page**
* **User Story 003: Apresentação da frota de Carros**
* **User Story 004: Apresentação de Detalhes do Carro**
* **User Story 005: Apresentação da localização do Carro**

**User Story 001: Preparação de ambiente**

**Descrição:**

É necessário preparar a ORG para conter os objetos descritos no diagrama acima. Isso pode ser feito realizando o deploy dos artefatos compartilhados nesse [link](https://capgemini-my.sharepoint.com/:u:/p/carlos_ferreira/EVo8lk-Pbg1PqY71iFkP17ABqfbaSv63fpD0LGB2PyPiXA?e=fB19LB).

**Descrição Técnica:**

Realizar deploy (Push to Org) com os artefatos compartilhados. Para ser possível usar para o desenvolvimento os objetos, tabs e recursos estáticos.

É necessário realizar o procedimento de pós-deploy, que consiste em atribuir as permissões de Criar, Editar e Excluir para os três objetos, junto com os acessos de leitura e escrita de todos os campos no perfil de Administrador de Sistema. Ainda, habilitar as tabs dos três objetos para o mesmo perfil.

Em seguida abrir o Developer Console e executar, no Anonymous Window, o comando “CreateData.createPackageData();”. Isso fará a carga inicial dos registros no ambiente.

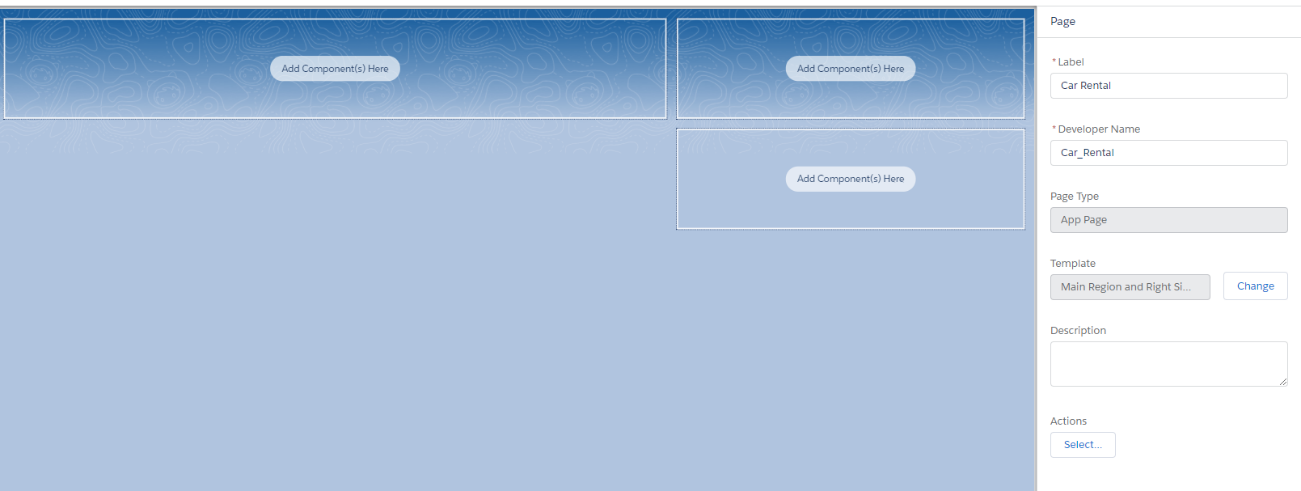
**User Story 002: Construção de App Page**

**Descrição:**

É necessário criar uma página Lightning para apresentar os componentes LWC desenvolvidos.

**Descrição Técnica:**

Criar, no Lightning App Builder, uma App Page com o nome Car Rental e que contenha 3 regiões, sendo uma principal à esquerda e duas (sendo uma sobre a outra) no lado direito como o exemplo abaixo:



Terá que estar disponível no aplicativo de Vendas.

**User Story 003: Apresentação da frota de Carros**

**Descrição:**

Criar alguns componentes para apresentar a frota de carros na tela. Isso envolve um filtro dos tipos de carros, uma ação para criar um tipo de carro novo e uma listagem dos carros disponíveis pelo filtro selecionado.

**Descrição Técnica:**

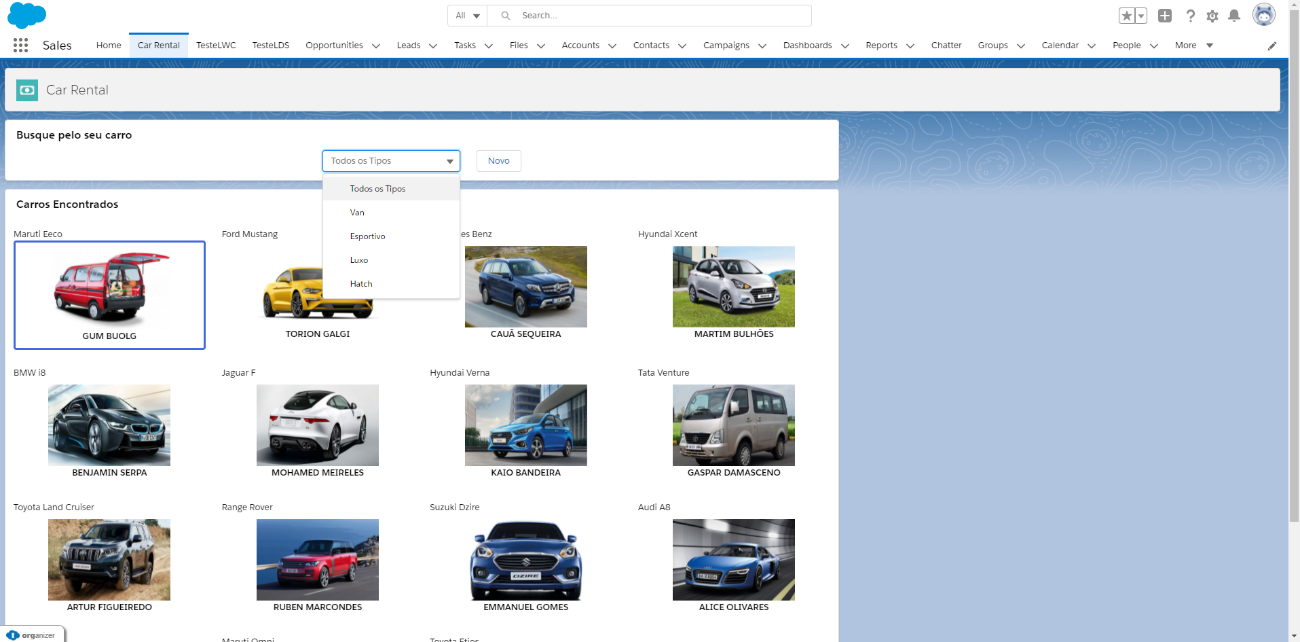
Será necessário criar de 3 a 4 componentes para atender esta US. Para isso, o principal componente deve ser um container para orquestrar as informações para os componentes filhos.

Será necessário o uso de Apex Class para realizar as buscas dos dados.

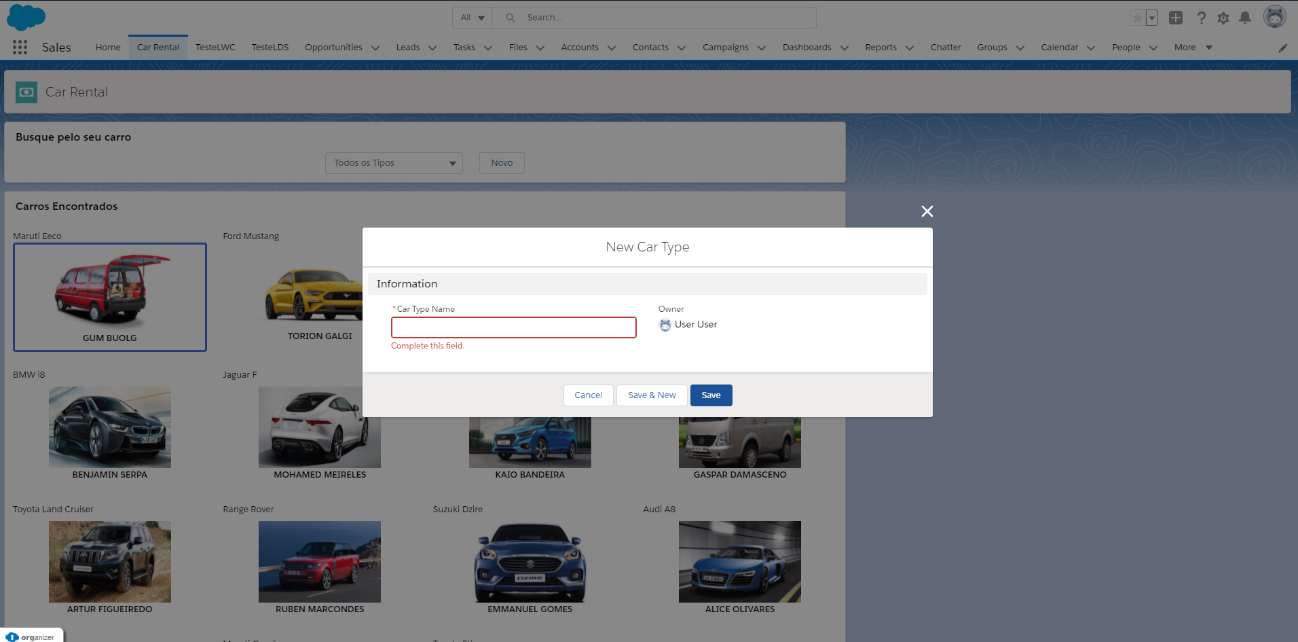
Será necessário utilizar a biblioteca de Navigation ou LDS para permitir a criação de um tipo novo de carro.

**\*Desejável ter um style aplicado para o carro selecionado.**

Protótipo:



Quando o botão novo for acionado:



**User Story 004: Apresentação de Detalhes do Carro**

**Descrição:**

Criar alguns componentes para apresentar os detalhes do carro selecionado, podendo apresentar alguns detalhes do carro na tela e uma ação para visualizar o registro completo. É possível o usuário preencher sua experiencia com o veículo selecionado e o usuário poder visualizar as experiencias postadas previamente por outros usuários do veículo selecionado. Os dados apresentados nas abas “Detalhe” e “Experiencias” devem ser recarregados toda vez que o usuário selecionar um carro novo. Toda vez que um usuário cadastrar uma experiencia nova, deverá recarregar as experiencias listadas na aba de “Experiencia” e a aba deve ficar selecionada. O componente não deve ser apresentado, caso não haja um veículo selecionado.

**Descrição Técnica:**

Criar 4 componentes para atender esta US. Para isso, o principal componente deverá ser um container para orquestrar as informações para os componentes filhos.

Para receber o Id do carro selecionado da **US003** será necessário o uso de uma biblioteca chamado [pubsub](https://capgemini-my.sharepoint.com/:u:/p/carlos_ferreira/EdaB7TfpxmtDqAa4qN7wRtABHaIUmdcg7HBWUf8pPpJVxw?e=FhakSL) e você pode verificar o exemplo de uso [aqui](https://www.jitendrazaa.com/blog/salesforce/lightning-web-component-event-handling-pub-sub/).

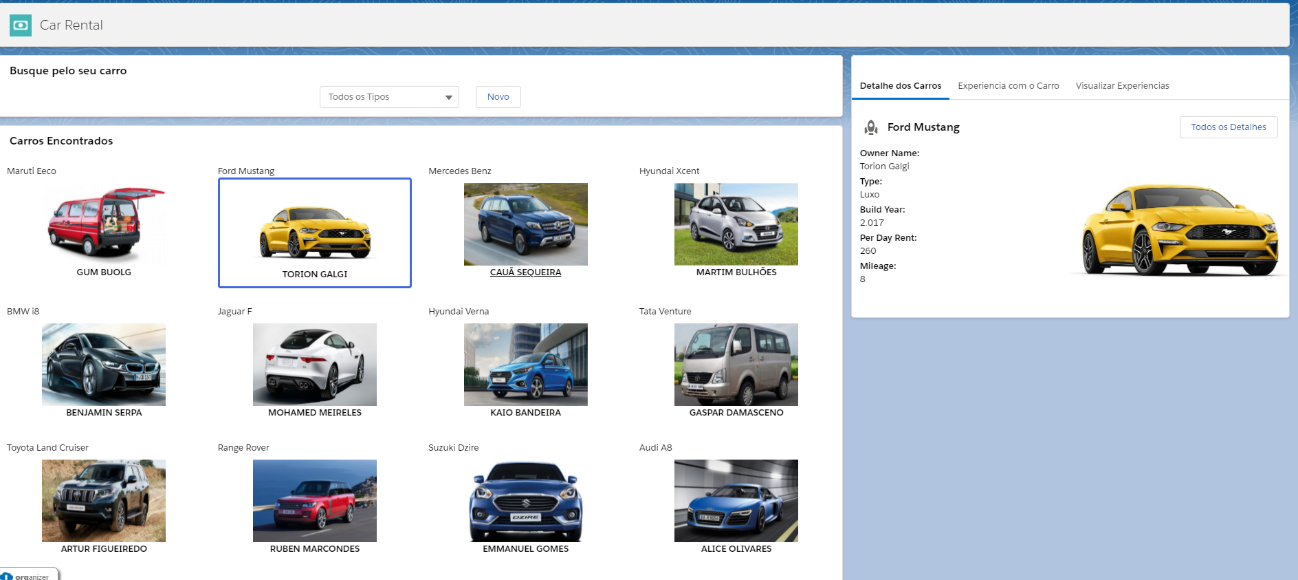
Será necessário o uso do LDS e Apex Class para realizar as buscas e persistência dos dados.

Será necessário utilizar a biblioteca de Navigation para redirecionar para a página de detalhes do carro.

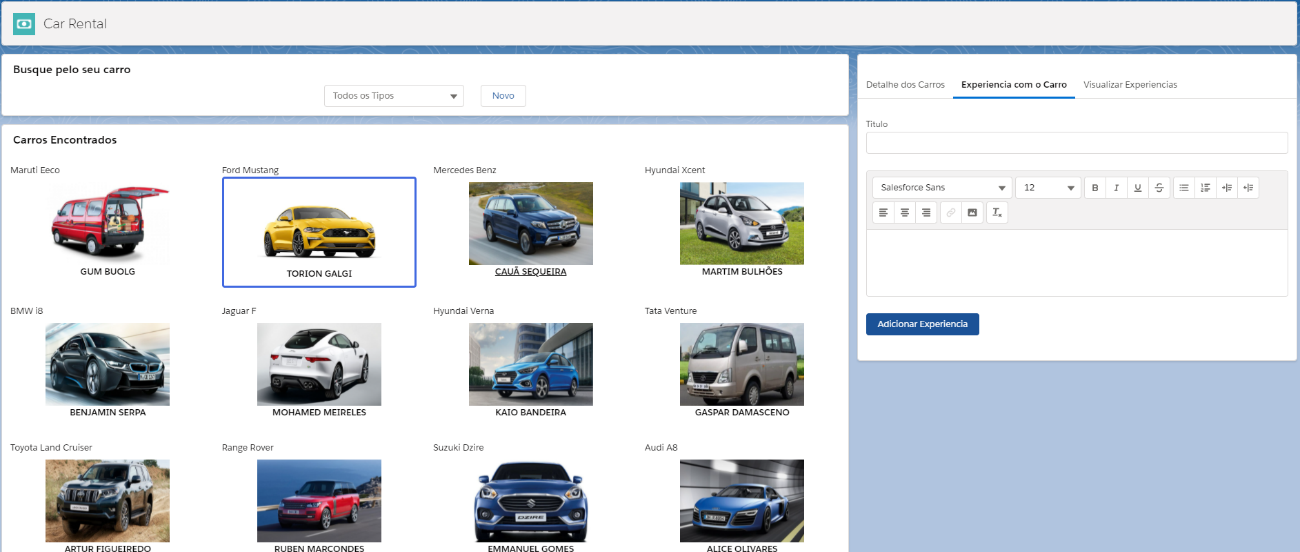
Deverá ser possível redirecionar para o usuário que postou a sua experiencia ao clicar no seu nome.

**\*Desejável usar schema para referenciar os campos e objetos quando for realizar a busca e persistência de dados.**

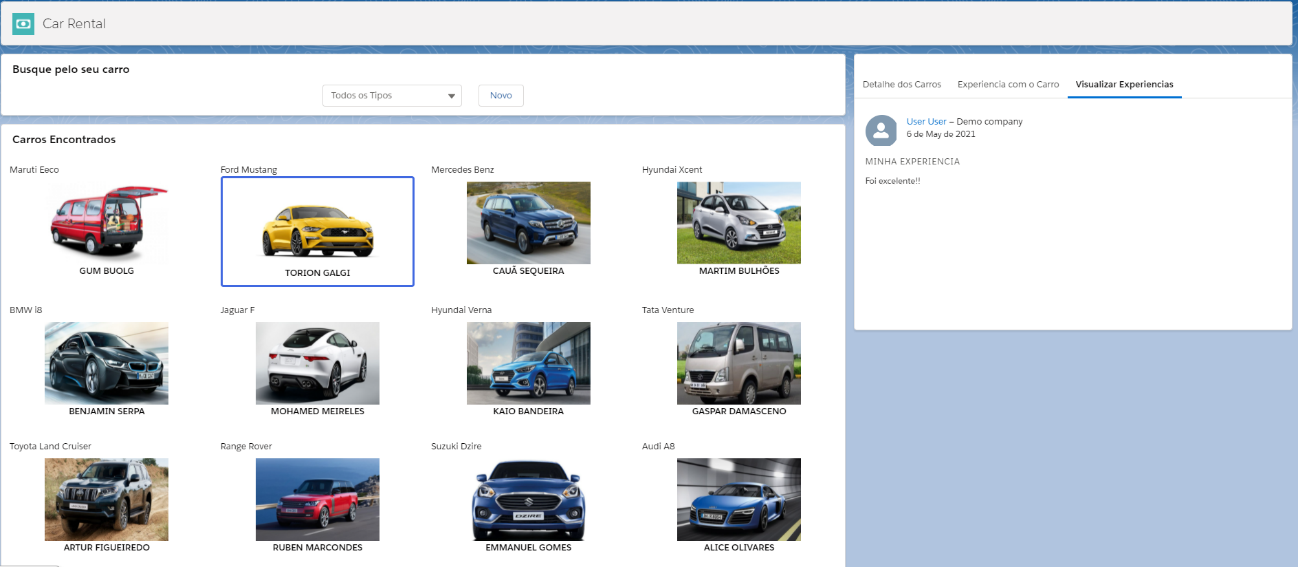
Protótipo:

**Detalhes** 

**Adicionar Experiencia**



**Visualizar Experiencia**



**User Story 005: Apresentação da localização do Carro**

**Descrição:**

Criar um componente para apresentar a localização do veículo. A localização do veículo está armazenada usando Latitude e Longitude. O usuário pode visualizar a posição do veículo pelo Google Maps. O Mapa não deve ser apresentado, caso não há veículo selecionado.

**Descrição Técnica:**

Será necessário criar 1 componente para atender essa US. Usando a mesma biblioteca do [pubsub](https://capgemini-my.sharepoint.com/:u:/p/carlos_ferreira/EdaB7TfpxmtDqAa4qN7wRtABHaIUmdcg7HBWUf8pPpJVxw?e=FhakSL) que foi definido na **US004**, podemos receber as mesmas informações passadas no desenvolvimento anterior.

Protótipo:

**Mapa**

