***Mechanical Manager***

TODO

1. ~~Criar o Mockups da App.~~
2. ~~Dicionário de dados com a base de dados SQLite.~~
   1. ~~Definir o default value nos campos de data com o CURRENT\_TIMESTAMP. Ex: create table tbl1(id int primary key, dt datetime default current\_timestamp);~~
3. ~~Criar a base de dados SQLite depois de terminar o dicionário de dados.~~
4. Estudar como aplicar a segurança (Falta aprovar a CRUD-Grid).
   1. <http://stackoverflow.com/questions/11540541/how-to-grant-permissions-to-a-row-in-oracle>
5. Criar as interfaces em Android Studio.
6. Fazer um CRON para verificar se envia alguma mensagem aos utilizadores a alertar que se aproxima a revisão, etc.
7. e o resto

DÚVIDAS

1. Como fazemos para inserir no SQLite apenas os novos registos? Com o MINUS?
2. Como fazemos para atualizar no SQLite um registo que foi atualizado no ORACLE? Ex: O meu mecânico atualizou o campo TELEMOVEL.
   1. Criar o campo VERSAO em cada tabela MECANICOS (SQLite e ORACLE) com o numero da versão. Sempre que houver uma alteração, a tabela do SQLite tem de ser atualizada.

OBSERVAÇÕES

* O utilizador só pode alterar o primeiro e último nome, e o telemóvel.
* Se o utilizador quiser alterar a password tem de estar online.

APIs públicas

* <http://fipeapi.appspot.com/>
  + Todas as marcas - http://fipeapi.appspot.com/api/1/carros/marcas.json
  + Modelos da marca 1 - http://fipeapi.appspot.com/api/1/carros/veiculos/1.json
* <https://vpic.nhtsa.dot.gov/api/>

Excell

1. Views
2. Sinónimos
3. Procedimentos
4. Trigger
   1. Consumo do veículo;
   2. Atualizar o valor total da reparação
   3. Mudar o óleo
   4. Adicionar o campo Data a tabela anomalias;
5. Criar Roles na BD
6. Transactions
   1. Atualização das BDs – no caso de não existir internet é importante que a Bd atualize com o backEnd a info;
   2. Actualização de ocorrências, por exemplo: ao termos 2 ocorrências estas teem de ser atualizadas ao mesmo tempo, para não perdermos uma...

Android app

# Sandro

* Globals.java
* DatabaseHelper.java
* MecanicoRepository.java

TODO

* Criar primeiro a classe Mecanico (ver a classe Link.java)
* Depois continuar em MecanicoRepository.insert()