

Universidade do Estado do Amazonas

Angélica - Sistema de controle e gerenciamento de taxistas

**Manaus - AM
2019**

SUMÁRIO

1. Identificação
2. Objetivo geral do projeto
3. Modelagem
4. Projeto de Banco
 - 4.1 Mapeamento
 - 4.2 Padrões Utilizados
5. Projeto
 - 5.1 Motorista
 - 5.2 Permissão
 - 5.3 Usuário
 - 5.4 Táxi
6. Spawner Data Generator
 - 6.1 Motorista
 - 6.2 Permissão
 - 6.3 Usuário
 - 6.4 Táxi
7. PostgreSQL
8. Tutoriais de instalação
 - 8.1 Toad Data Modeler
 - 8.2 PostgreSQL
 - 8.3 Spawner Data Generator
9. Referências

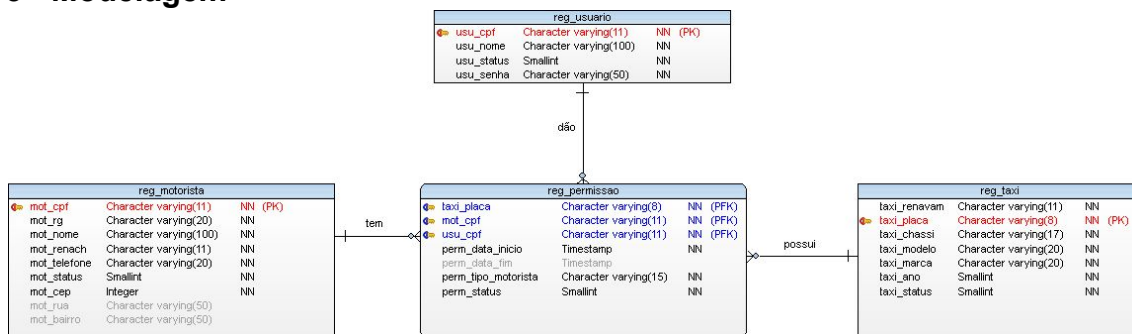
1 - Identificação

Angélica - Sistema de controle e gerenciamento de taxistas

2 - Objetivo geral do projeto

Projetar um sistema integrado para controle e gerenciamento de motoristas de táxi a fim de oferecer mais segurança e integridade para aqueles que utilizam dos serviços oferecidos, bem como o dos gerentes do sistema para facilitar administração de tais dados.

3 - Modelagem



4 - Projeto de Banco de Dados

4.1 - Mapeamento

1. reg_motorista = {mot_cpf#, mot_rg, mot_nome, mot_renach, mot_telefone, mot_status, mot_cep, mot_ua, mot_bairro}

PK = mot_cpf

2. reg_taxi = {tax_placa#, tax_renavam, tax_chassi, tax_modelo, tax_marca, tax_ano, tax_status}

PK = tax_placa

3. reg_usuario = {usu_cpf#, usu_nome, usu_senha, usu_status}

PK = usu_cpf

4. reg_permissao = {tax_placa#*, mot_cpf#*, usu_cpf#*, per_data_inicio, per_data_fim, per_tipo_motorista, per_status}

FK = tax_placa ref reg_taxi, mot_cpf ref reg_motorista, usu_cpf ref reg_usuario

4.2 - Padrões utilizados

- Notação para chave primária é “#” e para chaves estrangeiras “*”
- Para os nomes dos atributos das tabelas: três primeiras letras do nome da entidade, seguido do nome do atributo e todas as sequências separadas por uma _ (underline).
- Palavras no singular, sem acentuação e todas minúsculas;

- Para os nomes das entidades: três primeiras letras da palavra registro, seguido do nome da entidade, e todas as sequências separadas por uma _ (underline).

5 - Projeto

5.1 - Motorista

Descrição = Manter informações pertinentes dos motoristas, um motorista pode ser proprietário de um táxi e auxiliar de outro táxi.

Item	Nome do Atributo	Descrição	Formato (para as máscaras dos campos)	Exemplo
1	mot_cpf	Cpf do motorista	XXX.XXX.XXX-XX	"12345678910"
2	mot_rg	Rg do motorista		
3	mot_nome	Nome do motorista		"Fulano de tal"
4	mot_renach	Registro nacional de carteira de habilitação do motorista	AMXXXXXXXXXX	AM123456789
5	mot_telefone	Telefone do motorista	"(XX) XXXXX-XXXX"	"(99) 99999-9999"
6	mot_status	Se refere ao estado atual do motorista 1 - ativo 0 - inativo		"0" ou "1"
7	mot_cep	Cep do motorista	"XXXXX-XXXXX"	"69500-69999"
8	mot_rua	Se refere ao endereço do motorista		"Rua Amazonas"
9	mot_bairro	Se refere ao endereço do motorista		"Aleixo"

5.2 - Permissão

Descrição = Se refere ao relacionamento entre motoristas e táxis, mantém quais motoristas tem permissão de acesso a um táxi.

Item	Nome do Atributo	Descrição	Formato (para as máscaras dos campos)	Exemplo (no banco)
1	tax_placa	Placa do carro	XXX-XXXX	"ABC1234"
2	mot_cpf	CPF do motorista	XXX.XXX.XXX-XX	"12345678910"
3	usu_cpf	CPF do usuário	XXX.XXX.XXX-XX	"12345678910"
4	per_data_inicio	Data e horário de início do registro	DD/MM/AAAA hh:mm:ss	"28/03/2019 02:46:30"
5	per_data_fim	Data e horário da finalização do registro	DD/MM/AAAA hh:mm:ss	"28/03/2019 02:46:30"
6	per_tipo_motorista	Tipo do motorista, pode ser "auxiliar" ou "principal" que seria o proprietário do veículo		"auxiliar" ou "principal"
7	per_status	Status da permissão 1 - ativo 0 - inativo		"0" ou "1"

5.3 - Usuário

Descrição = Manter informações pertinentes dos usuários do órgão público.

Item	Nome do Atributo	Descrição	Formato (para as máscaras dos campos)	Exemplo
------	------------------	-----------	--	---------

1	usu_cpf	Cpf do usuário	XXX.XXX.XXX-XX	"12345678910"
2	usu_nome	Nome do usuário		"Fulano de tal"
3	usu_senha	Senha do usuário, no mínimo oito caracteres, 1 letra maiúscula, 1 letras minúscula, números e um caracter		"Senha1234@"
4	usu_status	Status do usuário 1 - ativo 0 - inativo		"1" ou "0"

5.4 Táxi

Descrição = Mantém os dados pertinentes dos carros dos motoristas.

Item	Nome do Atributo	Descrição	Formato (para as máscaras dos campos)	Exemplo (no banco)
1	tax_placa	Placa do carro	XXX-XXXX	"ABC1234"
2	tax_renavam	Registro Nacional de Veículos Automotores, é o número identificador único de cada veículo	XXXXXXXXXXXX	"12345678910"
3	tax_chassi	É a forma de registro universal dos veículos automotivos produzidos.	XXX.XXXXXX.XX.XXXXXX	"9BG.RD08X0.4G.117974"
4	tax_modelo	Modelo do carro		"Camaro"
5	tax_marca	Marca do carro		"CHEVROLET"
6	tax_ano	Ano do carro	XXXX	"1995"
7	tax_status	Status do carro 1 - ativo 0 - inativo		"0" ou "1"

6 - Spawner Data Generator

Spawner foi o executável utilizado para gerar valores aleatórios para popular as tabelas.

Baseado nos padrões citados acima, utilizou-se o spawner para simular resultados aleatórios semelhantes para popular a base de dados.

6.1 - Motorista

Campo	Tipo	Subtipo	Exemplo
mot_cpf	Text	masked string	44099505621
mot_rg	Text	masked string	19083868
mot_nome	Human	Full name	Nell Ballard
mot_renach	Text	masked string	46309106606
mot_telefone	Text	masked string	00016596
mot_status	Set	From fixed list	1
mot_cep	Number	Integer	69027838
mot_rua	Human	City	Yakima
mot_bairro	Human	State	Wisconsin

6.2 - Permissão

Campo	Tipo	Subtipo	Exemplo
tax_placa	Text	masked string	FCX5905
mot_cpf	Text	masked string	44099505621
usu_cpf	Text	masked string	45015612215
per_data_inicio	Time	date	2018-03-26 00:00:00
per_data_fim	Time	date	2020-03-26 00:00:00
per_tipo_motorista	Set	From fixed list	auxiliar
per_status	Set	From fixed list	0

6.3 - Usuário

Campo	Tipo	Subtipo	Exemplo
usu_cpf	Text	masked string	45015612215
usu_nome	Human	Full name	Lucas Petty
usu_senha	Text	masked string	GuvWg9N
usu_status	Set	From fixed list	1

6.4 - Táxi

Campo	Tipo	Subtipo	Exemplo
tax_placa	Text	masked string	FCX5905
tax_renavam	Text	masked string	17530660643
tax_chassi	Text	masked string	5NOHJQH1092YH2536
tax_modelo	Text	masked string	Yebmajumm
tax_marca	Text	masked string	Lztetttsl
tax_ano	Number	integer	2014
tax_status	Set	From fixed list	0

7 - PostgreSQL

Para criar as tabelas, foi necessário entrar na ferramenta de “Query Tool” da respectiva base de dados criada. A partir desta, é possível executar comandos sql para criar tabelas e modificar elementos da base de dados.

O arquivo sql gerado pelo Toad Data Modeler foi utilizado para criar as tabelas. Os arquivos gerados pelo Spawner foram utilizados para inserir dados aleatórios nessas tabelas. Todos pela ferramenta “Query Tool”.

8 Tutoriais de instalação

Os programas utilizados foram Toad Data Modeler para modelagem, PostgreSQL para manter a base de dados e Spawner Data Generator para gerar

8.1 - Toad Data Modeler

Instalação e configuração do software pode ser visto através do link:

<https://docs.google.com/presentation/d/1AftzuOf6OyY5GEoVzc7FN5HfHqPxcuxvNae7naCpE9Q/edit#slide=id.p>

8.2 - PostgreSQL

Instalação e configuração do software pode ser visto através do link:

https://docs.google.com/presentation/d/1E3VaxYO0eKJvomFVsq7f-9Ew7qPNto1mlkPu_a7LDckl/edit?usp=sharing

8.3 - Spawner Data Generator

Instalação do software Spawner Data Generator pode ser visto através do link:

<https://docs.google.com/presentation/d/1Lkgant0khzNb9vnKff1msWQ2937txd3Y1JBgku01Ckg/edit?usp=sharing>

Referências

- Serviço Renach. Disponível em <<https://servicos.detran.am.gov.br/>> Acesso em 02 de abril de 2019;
- Placas de identificação de veículos no Brasil. Disponível em <<https://pt.wikipedia.org/>> Acesso em 02 de abril de 2019;
- Registro Nacional de Veículos Automotores. Disponível em <<https://pt.wikipedia.org/>> Acesso em 02 de abril de 2019;
- Número de Identificação do Veículo. Disponível em <<https://pt.wikipedia.org/>> Acesso em 02 de abril de 2019.