

03

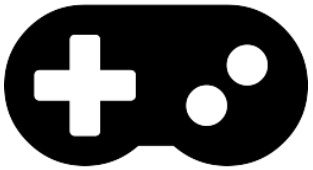
WEBAPP LIKE A BOSS

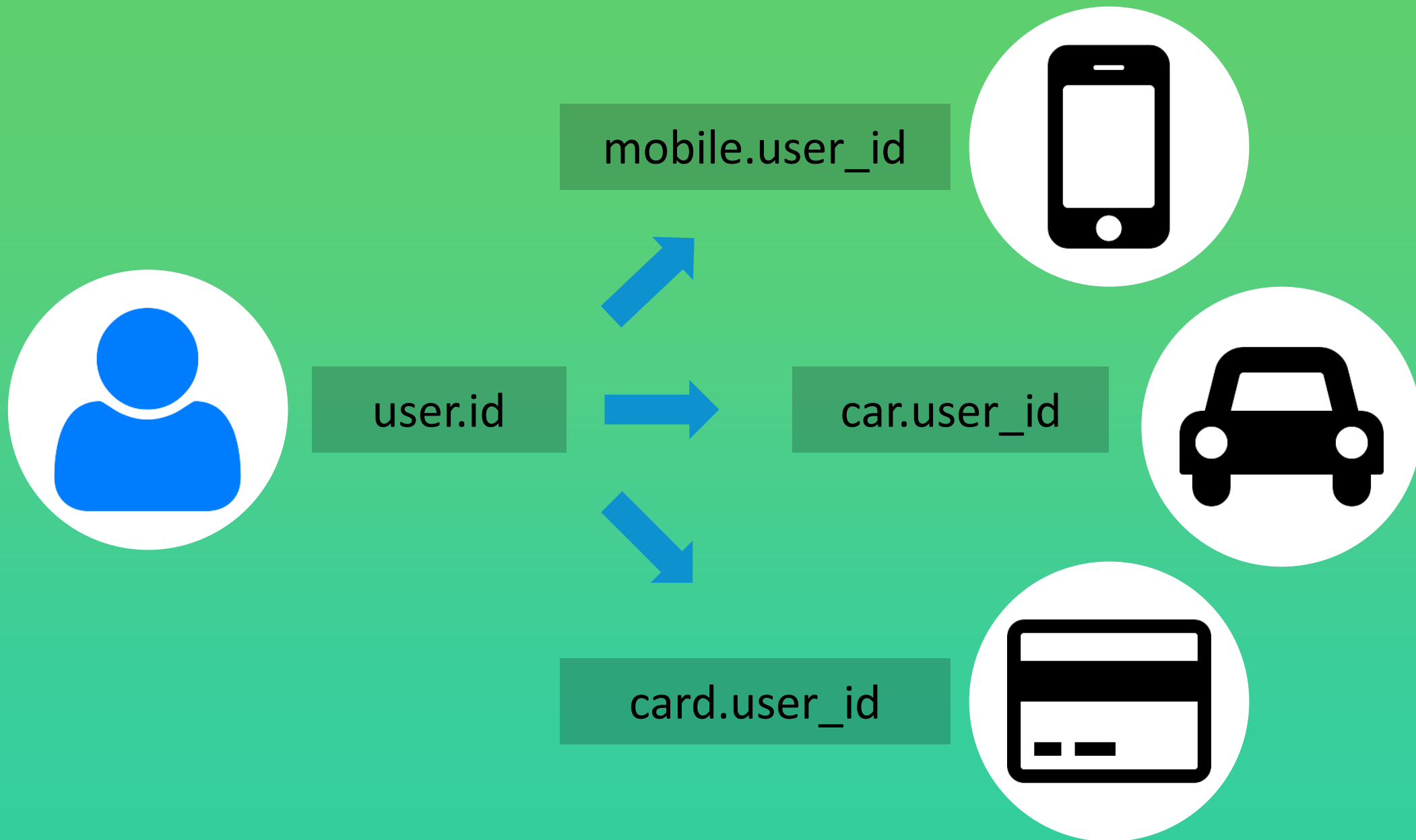
O MUNDO DAS RELAÇÕES





Como represento relações?







Relationships

Acompanhe a documentação

<http://laravel.com/docs/5.1/eloquent-relationships>



client.id

database/migrations/2014_10_12_000000_create_users_table.php



```
Schema::create('clients', function (Blueprint $table) {  
    $table->increments('id');  
    $table->integer('establishment_id')->nullable();  
    $table->string('firstname');  
    $table->string('lastname');  
    $table->string('address_state')->nullable();  
    $table->string('address_city')->nullable();  
    $table->string('address_street')->nullable();  
    $table->string('address_number')->nullable();  
    $table->string('address_zip')->nullable();  
    $table->string('contact_email')->nullable();  
    $table->string('contact_phone')->nullable();  
    $table->string('image_profile')->nullable();  
    $table->timestamps();  
    $table->softdeletes();  
});
```



card.client_id

database/migrations/2014_10_12_000000_create_cards_table.php



```
Schema::create('cards', function (Blueprint $table) {  
    $table->increments('id');  
    $table->integer('client_id')->nullable();  
    $table->string('name')->nullable();  
    $table->string('flag')->nullable();  
    $table->string('number')->nullable();  
    $table->string('expires_at')->nullable();  
    $table->string('expires_string')->nullable();  
    $table->string('cvv')->nullable();  
    $table->timestamps();  
    $table->softdeletes();  
});
```



client.id

app/Client.php



```
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Client extends Model
{
    public function establishment()
    {
        return $this->belongsTo('App\Establishment');
    }

    public function cards()
    {
        return $this->hasMany('App\Card');
    }
}
```




card.client_id

app/Card.php



```
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Card extends Model
{

    public function owner()
    {
        return $this->belongsTo('App\Client');
    }

}
```



Relações



cards()



```
App\Client::find(1)->cards()
```

Teremos uma lista com todos os cartões do cliente de ID = 1. Retorna uma collection de App\Cards.



Relações



owner()



```
App\Card::find(1)->owner()
```

Acharemos o dono do cartão de ID = 1. Retorna um model App\Client.



Exemplos

Qual o nome do dono do cartão de ID = 1?

```
App\Card::find(1)->owner()->name;
```

Quantos cartões o cliente de ID = 1 tem?

```
App\Client::find(1)->cards()->count();
```



Collections

Acompanhe a documentação

<http://laravel.com/docs/5.1/collections>

all() avg() chunk() collapse() contains() count() diff() each() every() except()
filter() first() flatten() flip() forget() forPage() get() groupBy() has()
implode() intersect() isEmpty() keyBy() keys() last() map() max() merge()
min() only() pluck() pop() prepend() pull() push() put() random() reduce()
reject() reverse() search() shift() shuffle() slice() sort() sortBy()
sortByDesc() splice() sum() take() toArray() toJson() transform() unique()
values() where() whereLoose() zip()



Exemplo: Nome do dono do cartão

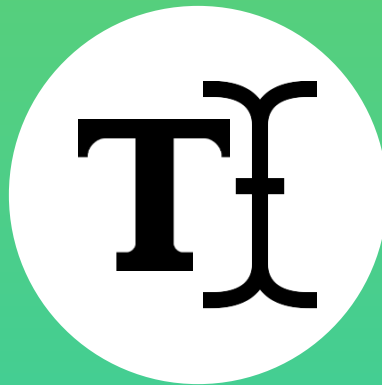
ELE ESTÁ
VIVO!



Interatividade



clicks



inputs

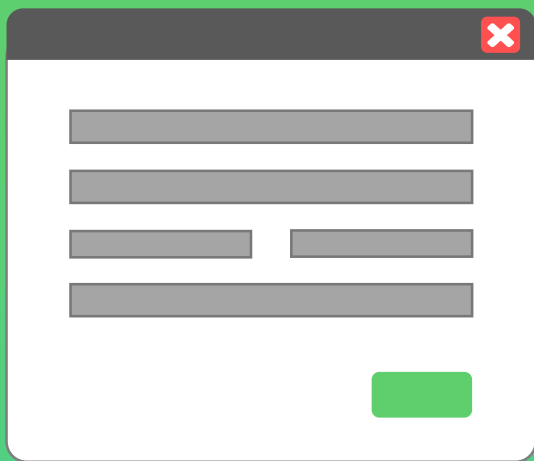


drag

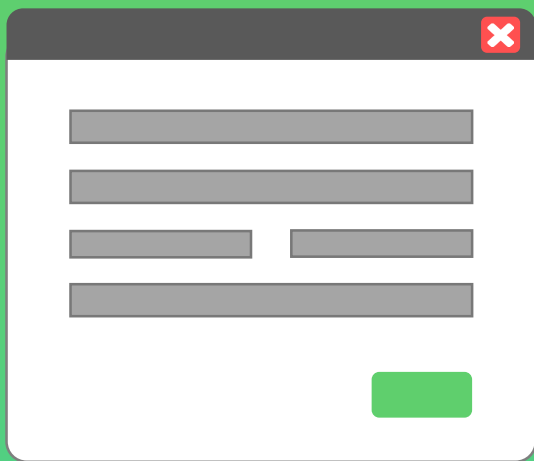


Interatividade



A stylized icon of a web form window. It has a dark gray header bar with a red square button containing a white 'X' in the top right corner. The main body is white and contains five gray rectangular input fields: three full-width fields at the top, followed by two smaller fields side-by-side, and one full-width field at the bottom. A solid green rectangular button is located in the bottom right corner of the white area.

formulário



request



controller



Authentication

Acompanhe a documentação

<http://laravel.com/docs/5.1/authentication>

FOLDERS

- easy money-simulator
 - app
 - Console
 - Events
 - Exceptions
 - Http
 - Controllers
 - Middleware
 - Requests
 - Kernel.php
 - routes.php
 - Jobs
 - Listeners
 - Policies
 - Providers
 - Card.php
 - Client.php
 - Establishment.php
 - Pinpad.php
 - Transaction.php
 - User.php
 - bootstrap
 - config
 - database
 - public
 - resources
 - storage
 - tests

<?php

/*

Application Routes

Here is where you can register all of the routes for an application.
It's a breeze. Simply tell Laravel the URIs it should respond to
and give it the controller to call when that URI is requested.

*/

Route::get('login',	'Auth\AuthController@getLogin');
Route::post('login',	'Auth\AuthController@postLogin');
Route::get('logout',	'Auth\AuthController@getLogout');
Route::get('register',	'Auth\AuthController@getRegister');
Route::post('register',	'Auth\AuthController@postRegister');
Route::get('password/email',	'Auth>PasswordController@getEmail');
Route::post('password/email',	'Auth>PasswordController@postEmail');
Route::get('password/reset/{token}',	'Auth>PasswordController@getReset');
Route::post('password/reset',	'Auth>PasswordController@postReset');

Register

Name

Email

Password

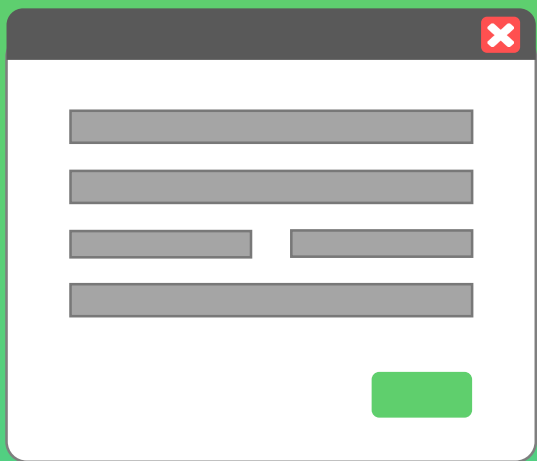
Confirm Password

Register

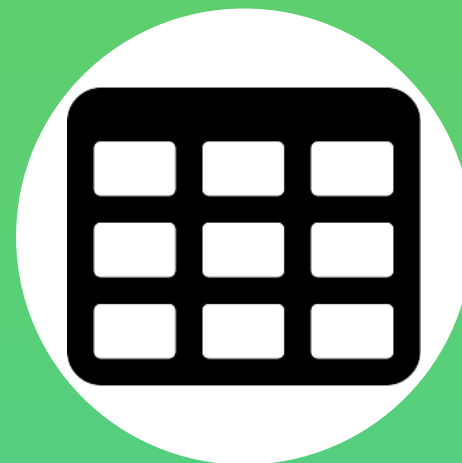
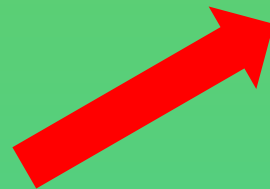


```
@extends('layouts.public')

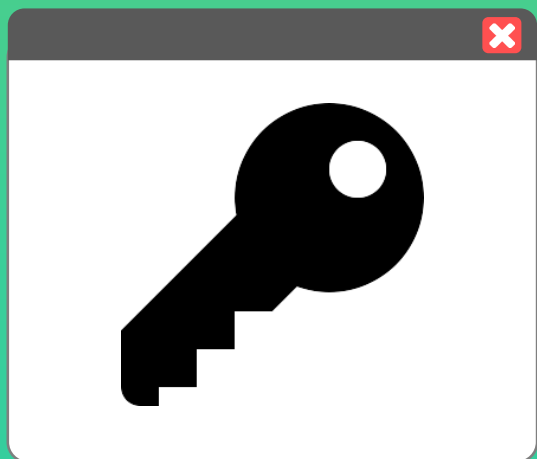
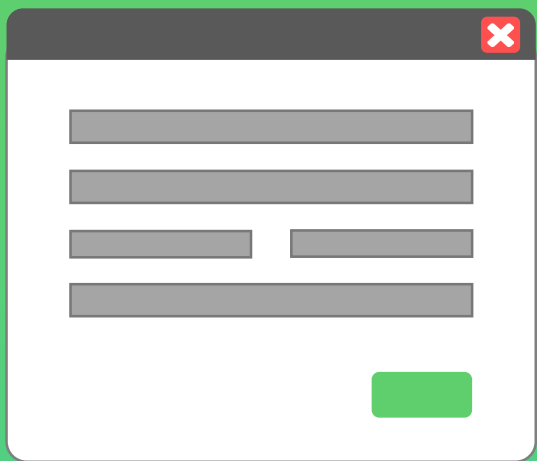
@section('title', 'Register')
@section('content')
<form method="POST" action="/register">
    {!! csrf_field() !!}
    <div class="row">
        <div class="col-md-2"><p class="pull-right">Name</p></div>
        <div class="col-md-10"><input type="text" name="name" value="{{ old('name') }}"></div>
    </div>
    <div class="row">
        <div class="col-md-2"><p class="pull-right">Email</p></div>
        <div class="col-md-10"><input type="email" name="email" value="{{ old('email') }}"></div>
    </div>
    <div class="row">
        <div class="col-md-2"><p class="pull-right">Password</p></div>
        <div class="col-md-10"><input type="password" name="password"></div>
    </div>
    <div class="row">
        <div class="col-md-2"><p class="pull-right">Confirm Password</p></div>
        <div class="col-md-10"><input type="password" name="password_confirmation"></div>
    </div>
    <div class="row">
        <div class="col-md-offset-2 col-md-10"><button type="submit">Register</button></div>
    </div>
</form>
@stop
```



User::create()



model



Login

Email

Password

☐ Remember me

Login



```
@extends('layouts.public')

@section('title', 'Login')
@section('content')
<form method="POST" action="/login">
  {!! csrf_field() !!}
  <div class="row">
    <div class="col-md-2"><p class="pull-right">Email</p></div>
    <div class="col-md-10"><input type="email" name="email" value="{{ old('email') }}"></div>
  </div>
  <div class="row">
    <div class="col-md-2"><p class="pull-right">Password</p></div>
    <div class="col-md-10"><input type="password" name="password" id="password"></div>
  </div>
  <div class="row">
    <div class="col-md-offset-2 col-md-10">
      <p class=""><input type="checkbox" name="remember"> Remember me</p>
    </div>
  </div>
  <div class="row">
    <div class="col-md-offset-2 col-md-10"><button type="submit">Login</button></div>
  </div>
</form>
@stop
```

