

UPDATE

Creiamo una rotta *get* che ci porti alla pagina di modifica e un bottone dedicato nell'HTML. Deve essere una **rotta parametrica** perché dobbiamo trasportare il dato scelto per poi modificarlo.

Rotta parametrica:

```
{{route('travel.edit', compact('travel'))}}
```

```
Route::get('/edit/{travel}', [TravelController::class, 'edit'])->name('travel.edit')
```

Nel *TravelController* facciamo passare il dato attraverso la funzione *edit* che ha la *view* del form di modifica:

```
public function edit(Travel $travel){  
    return view('travel.edit', compact('travel'));  
}
```

Nel tag input del form di modifica inseriamo il **value**, e indirizziamo il *form* ad una rotta di tipo **put**.

Creiamo ora la **route put**:

```
Route::put('/update/{travel}', [TravelController::class, 'update'])->name('travel.update')
```

Nell'HTML:

```
<form method="POST" action="{{route('travel.update', $travel)}}">  
@method('PUT')  
@csrf  
<input name="title" value="{{ $travel->title }}" selected="{{ $old->title->title }}">
```

@method . Eseguiamo un *overwrite* del method POST nel form (*deve essere sempre post*)
action . Deve diventare una rotta parametrica perché la funzione *update()* necessita di un file in entrata oltre alla *\$request*.

Nel *TravelController* assegno il valore della *\$request* al valore *\$travel* della *Request*:

```
public function update(Request $request, Travel $travel){  
    $travel->title = $request->title;  
    if ($request->img){  
        $travel->img = $request->file('img')->store('public/img')  
    };  
    $travel->save();  
  
    return redirect(route('travel.show', compact('travel')));  
}
```

```
}
```

save() . Salviamo per apportare le modifiche al database usando la funzione `save()`

NB: Nel caso volessimo permettere di modificare un dato solo e non tutti, dovremmo dare alla rotta (e all'HTML) il tipo `PATCH`. Oppure ancora meglio, creare una `UpdateRequest` senza valori `'required'`.

DELETE

Per **eliminare** creiamo una rotta diventa di tipo **delete**:

```
Route::delete('/delete/{travel}', [TravelController::class, 'destroy'])->name('travel.delete');
```

Nell'HTML dove vogliamo inserire il `<button>` *Elimina* creiamo un tag `<form>` per la gestione della rotta visto che deve avere un *Method*.

```
@if (Auth::id() == $travel->user_id)
    <form method="POST" action="{{route('travel.delete', $travel)}}">
        @method('DELETE')
        @csrf
        <button type="submit">Cancella</button>
    @endif
```

button . Il form richiede esclusivamente un tag *button*. L'`<a>` non accetta il `type` come attributo.

@if / @endif . Stiamo creando una **corrispondenza** tra chi e' lo user, e le azioni che puo' compiere. Solo se l'*id* dell' *Auth::* e' uguale all'*user_id* del dato *\$travel*, puo' visualizzare il bottone *delete*.

Nel *TravelController*:

```
public function destroy(Travel $travel){
    $travel->delete();

    return redirect(route('travel.index'));
}
```