Adicionar o Método na Interface de Cupom

**public function** findByCode($code);

Adicionar o Método no CupomRepositoryEloquent

**public function** findByCode($code)  
{  
 $result = $this->**model**->where(**'code'**, $code)->where(**'used'**, 0)->first();  
  
 **if**($result)  
 {  
 *//RETORNA O ARRAY COM PRESENTER OU INSTACIA DA COLLECTION* **return** $this->parserResult($result);  
 }  
  
 *//EXCEPETION PARA REGISTRO NAO ENCONTRADO* **throw** (**new** ModelNotFoundException)->setModel(*get\_class*($this->**model**));  
}

Adicionar o Presenter

**public function** presenter()  
{  
 **return** CupomPresenter::***class***;  
}

Criar CupomPresenter e Transformer

**class** CupomPresenter **extends** FractalPresenter  
{  
 */\*\*  
 \* Transformer  
 \*  
 \** ***@return*** *\League\Fractal\TransformerAbstract  
 \*/* **public function** getTransformer()  
 {  
 **return new** CupomTransformer();  
 }  
}

**class** CupomTransformer **extends** TransformerAbstract  
{  
  
 */\*\*  
 \* Transform the \Cupom entity  
 \** ***@param*** *\Cupom $model  
 \*  
 \** ***@return*** *array  
 \*/* **public function** transform(Cupom $model)  
 {  
 **return** [  
 **'id'** => (int) $model->**id**,  
 **'code'** => $model->**code**,  
 **'value'** => (float)$model->**value**,  
*// 'created\_at' => $model->created\_at,  
// 'updated\_at' => $model->updated\_at* ];  
 }  
}