

Utilizando o cliente de cartão de crédito para consumir o serviço externo

No *controller* `pagamentos.js`, altere a rota que recebe um pagamento para que passe a verificar o pagamento com cartão. A primeira coisa a fazer é mudar a forma como o corpo da requisição é lido, pois agora a regra evoluiu, sem quebrar o que já funcionava. O *json* recebido agora terá duas chaves principais, uma contendo os dados do pagamento em si e outra contendo os dados do cartão.

Isso implica em uma mudança em como lemos a informação do pagamento a partir do body:

```
...
app.post("/pagamentos/pagamento", function(req, res) {
  var body = req.body;
  var pagamento = body['pagamento'];
  ...
}
```

E também em como faremos a partir de agora para definir os campos a ser validados:

```
...
req.assert("pagamento.forma_de_pagamento", "Forma de pagamento é obrigatória.").notEmpty();
req.assert("pagamento.valor", "Valor é obrigatório e deve ser um decimal.").notEmpty().isFloat();
req.assert("pagamento.moeda", "Moeda é obrigatória e deve ter 3 caracteres").notEmpty().length(3);

var errors = req.validationErrors();
...
}
```

Agora implemente a verificação do tipo de pagamento antes de montar o *json* de retorno do serviço. Caso o tipo seja "cartao", o código deve instanciar um objeto `cartoesClient` para que ele invoque o serviço dos cartões.

A partir desse objeto invoque a função `autoriza` que foi criada no `cartoesClient` passando os dados do cartão que foram recebidos no body da requisição.

Reinicie o servidor e teste o pagamento com cartão usando o novo formato para o *json* de pagamento.

```
curl -X POST -v -H "Content-type: application/json"
http://localhost:3000/pagamentos/pagamento
-d '{
  "pagamento": {
    "forma_de_pagamento": "cartao",
    "valor": "10.87",
    "moeda": "BRL",
    "descricao": "descrição do pagamento"
  },
  "cartao": {
    "numero": "1234567890123456",
    "bandeira": "VISA",
    "ano_de_expiracao": "2020",
    "mes_de_expiracao": "12",
  }
}
```

```
      "cvv": "12"  
    }  
  }' ; echo
```

Envie dados inválidos no json para validar o retorno do serviço.

Teste também enviar um pagamento com `forma_de_pagamento = payfast`, como estávamos fazendo até então, para garantir que continua funcionando.