



## Proposta

Como já deve saber, a Wex é focada em meios de pagamentos, e nosso desafio não poderia ser diferente!

Precisamos lançar, de forma prática, um novo produto no mercado, o Wex PREPAG, um cartão pré pago que pode ser adquirido sem muita burocracia.

Será necessário o desenvolvimento de uma nova API cuja suas principais finalidades serão emitir cartões e autorizar compras.

## Sobre o desenvolvimento

O projeto deverá ser feito em **JAVA** utilizando **Spring Boot**. O banco de dados a gosto de desenvolvedor.

### Emissão do Cartão

A emissão do cartão será de forma tão simples, que só será exigido como parâmetros de entrada o **NOME** e o **SALDO** a ser creditado.

A API REST deverá retornar os seguintes dados do cartão:

**nome:** nome do titular

**número:** número do cartão de crédito

**cvv:** código de segurança de 3 números

**validade:** data de validade no formato MM/YY

**senha:** senha numérica de 4 dígitos

**saldo:** saldo do cartão

O número do cartão de crédito deve possuir 16 números e os 6 primeiros dígitos, que representa o BIN do cartão, devem ser sempre o mesmo para esta API.

O CVV é um código de 3 dígitos que, por segurança, não pode ser armazenado na base de dados. Ele sempre deve ser calculado utilizando um algoritmo que combina o **número do cartão** e a **validade** que deve resultar em um número de 3 dígitos. Utilize da sua criatividade para construir esse algoritmo.

A senha inicial deve ser um número randômico de 4 dígitos salvo criptografado na base de dados.

A validade do cartão deve ser de 2 anos contando da data de solicitação.

## Autorização de Venda

Para que uma compra seja autorizada o requisitante deve informar os seguintes dados:



```
{  
  cartao: número do cartão de crédito,  
  validade: validade (MM/YY),  
  cvv: cvv do cartão,  
  estabelecimento: nome do estabelecimento,  
  valor: valor da venda,  
  senha: senha de 4 dígitos  
}
```

Para que a compra seja aprovada, a requisição deverá passar pelas seguintes validações:

- Se o cartão existe
- Se o cartão não expirou
- Se o cvv é válido
- Se a senha é válida
- Se o cartão possui saldo suficiente para efetivar a compra

Deverá ser criado um código de erro para cada validação. Em caso de falha o mesmo deverá ser retornado com sua respectiva mensagem.

Em caso de sucesso retornar:

```
{  
  codigo: 00,  
  saldo: <novo saldo do cartão após compra>  
}
```

## **Itens a serem considerados**

- design da solução
- legibilidade
- facilidade de evolução e manutenção da aplicação
- testes automatizados / uso de boas práticas de agilidade
- operacionalidade
- tratamento de erros
- logs
- arquitetura da aplicação

**Será considerado como diferencial o desenvolvimento do front-end e utilização de Docker.**



## **Envio da Solução**

O projeto deve estar no [GitHub](#) e após a conclusão enviar o link do repositório. No **README.md** deve conter uma breve descrição do design, detalhes de implementação e arquitetura, qual banco de dados utilizado, bem como as instruções detalhadas de como rodar sua aplicação.

Qualquer dúvida estamos à disposição.

Aguardamos seu retorno e Boa sorte!