

| | | |
|--------------------------------------|------------------------------------------------------|---------------------------------------|
| Solicitante: Reserva Fácil | Projeto/Sistema: Prova de programação Java | Atividade: Desenvolvimento |
| Data: 11/09/2018 | Funcionalidade: Agendamento Eletrônico | Responsável: Samuel Santana |

Refinamento de Funcionalidade

| HISTÓRICO DE ALTERAÇÕES | | |
|-------------------------|----------------|-----------------------|
| DATA | RESPONSÁVEL | MOTIVO |
| 11/09/2018 | Samuel Santana | Criação do documento. |

| OBJETIVO |
|-------------------------------------------------------------|
| Criar um serviço de agendamento de Transferência Eletrônica |

| ATORES |
|---------------------------|
| Aplicação Clientes e API. |

| MOTIVAÇÃO |
|-------------------------------------------------------------------------------------------|
| A fim de preencher uma vaga de desenvolvedor Java, foi desenvolvido o projeto em questão. |

| DETALHAMENTO DE SERVIÇO | | |
|---------------------------------------|----------------------------------------------------------------------|------------------------------------------------------------------------------|
| Nome do Serviço: | CriarAgendamento | |
| Endereço de acesso: | http://localhost:8080/transferencia-eletronica-api/api/transferencia | |
| Versão: | 1.00 | |
| Tipo: | POST | |
| Funcionalidade | Nome da Operação | Objetivo |
| Criar agendamento das transferências. | <i>Criar Agendamento</i> | Criar o agendamento eletrônico das trãnfêrências através de requisição POST. |

| PARÂMETROS DE ENTRADA E DE SAÍDA |
|----------------------------------|
|----------------------------------|

| | | |
|--------------------------------------|------------------------------------------------------|---------------------------------------|
| Solicitante: Reserva Fácil | Projeto/Sistema: Prova de programação Java | Atividade: Desenvolvimento |
| Data: 11/09/2018 | Funcionalidade: Agendamento Eletrônico | Responsável: Samuel Santana |

Parâmetros de Entrada:

| Nome | Tipo | Requerido | Descrição |
|-------------------|--------|-----------|--------------------------------------|
| dataTransferência | String | Sim | Data da realização da transferência. |
| contaOrigem | String | Sim | Conta que sera creditado o recurso. |
| contaDestino | String | Sim | Conta que sera debitado o recurso. |
| valor | Double | Sim | Valor da Transferência |

Parâmetros de Saída (quando retorno for com erro):

| Nome | Tipo | Descrição |
|------------|---------|--------------------------------------------------|
| mensagem | String | Mensagem de retorno com o resultado da operação. |
| status | Integer | Código do retorno HTTP da requisição |
| statusCode | String | Descrição do retorno HTTP da requisição |

Parâmetros de Saída (quando retorno for sem erro):

| Nome | Tipo | Descrição |
|------------|---------|--------------------------------------------------|
| mensagem | String | Mensagem de retorno com o resultado da operação. |
| status | Integer | Código do retorno HTTP da requisição |
| statusCode | String | Descrição do retorno HTTP da requisição |

EXEMPLO DE REQUISIÇÃO

Quando retorno for com erro:

| Entrada | Saída |
|----------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------|
| <pre>{ "dataTransferencia" : "28/10/2018", "valor" : 1000.0, "contaOrigem" : "104736", "contaDestino" : "204789" }</pre> | <pre>{ "mensagem": "Não foi encontrado taxa para os parâmetros informados!", "status": "BAD_REQUEST", "statusCode": 400 }</pre> |

Quando retorno for sem erro:

| Entrada | Saída |
|-------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------|
| <pre>{ "dataTransferencia" : "11/09/2018", "valor" :10.0, "contaOrigem" : "104736", "contaDestino" : "204789" }</pre> | <pre>{ "mensagem": "Recurso criado com sucesso!", "status": "CREATED", "statusCode": 201 }</pre> |

| | | |
|--------------------------------------|------------------------------------------------------|---------------------------------------|
| Solicitante: Reserva Fácil | Projeto/Sistema: Prova de programação Java | Atividade: Desenvolvimento |
| Data: 11/09/2018 | Funcionalidade: Agendamento Eletrônico | Responsável: Samuel Santana |

DETALHAMENTO DE SERVIÇOS

| | | |
|-------------------------------------------|----------------------------------------------------------------------|----------------------------------------------------------------------------------|
| Nome do Serviço: | recuperarAgendamento | |
| Endereço de acesso: | http://localhost:8080/transferencia-eletronica-api/api/transferencia | |
| Versão: | 1.00 | |
| Tipo: | GET | |
| Funcionalidade | Nome da Operação | Objetivo |
| Recuperar agendamento das transferências. | <i>Recuperar Agendamento</i> | Recuperar o agendamento eletrônico das transferências através de requisição GET. |

PARÂMETROS DE ENTRADA E DE SAÍDA

Parâmetros de Saída (quando retorno for sem erro):

| Nome | Tipo | Descrição |
|--------------------------|---------|-----------------------------------------------------------------------------------|
| url | String | Url do recurso solicitado |
| httpstatus | Integer | Código do retorno HTTP da requisição |
| quantidadeTransferencias | Integer | Quantidade de transferências encontradas |
| transferencias | List | Lista das transferências |
| dataTransferencia | String | Atributo do objeto transferência para a Data da transferência |
| dataAgendamento | String | Atributo do objeto transferência para a Data da transferência Data do agendamento |
| valor | Double | Atributo do objeto transferência para o Valor da transferência |
| contaOrigem | String | Atributo do objeto transferência para a conta de origem |
| contaDestino | String | Atributo do objeto transferência para a conta de destino |
| valorTaxa | Double | Atributo do objeto transferência para o valor da taxa |

| | | |
|--------------------------------------|------------------------------------------------------|---------------------------------------|
| Solicitante: Reserva Fácil | Projeto/Sistema: Prova de programação Java | Atividade: Desenvolvimento |
| Data: 11/09/2018 | Funcionalidade: Agendamento Eletrônico | Responsável: Samuel Santana |

EXEMPLO DE REQUISIÇÃO

| Entrada | Saída |
|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | <pre>{ "url":"/transferencia-eletronica-api/api/transferencia", "httpStatus":200, "quantidadeTransferencias":3, "transferencias":[{ "dataTransferencia":"11/09/2018", "dataAgendamento":"11/09/2018", "valor":10.00, "contaOrigem":"104736", "contaDestino":"204789", "valorTaxa":3.29 }, { "dataTransferencia":"11/09/2018", "dataAgendamento":"11/09/2018", "valor":10.00, "contaOrigem":"104736", "contaDestino":"204789", "valorTaxa":3.29 }, { "dataTransferencia":"21/10/2018", "dataAgendamento":"11/09/2018", "valor":1000.00, "contaOrigem":"104736", "contaDestino":"204789", "valorTaxa":40.00 }] }</pre> |

Frameworks e Tecnologias

| | |
|------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Spring: | Documentação robusta, várias referências, mais utilizado no mundo java, facilidade de implementação e manutenção, diversas soluções para diferentes tipos de contextos. |
| Strategy: | Padrão utilizado para definir qual taxa será aplicada dependendo do contexto que a transferência apresenta observado na classe <code>sls.transferenciaeletronica.core.transferencia.enums.Taxa</code> . |
| Maven: | Maior facilidade para construção de aplicações modulares, maior |

| | | |
|--------------------------------------|------------------------------------------------------|---------------------------------------|
| Solicitante: Reserva Fácil | Projeto/Sistema: Prova de programação Java | Atividade: Desenvolvimento |
| Data: 11/09/2018 | Funcionalidade: Agendamento Eletrônico | Responsável: Samuel Santana |

| | |
|---------------|--------------------------------------------------------|
| | gerenciamento das dependências e construção de builds. |
| Java: | Versão 8 |
| IDE: | Eclipse NEON 2 |
| Banco: | Derby |

Arquitetura

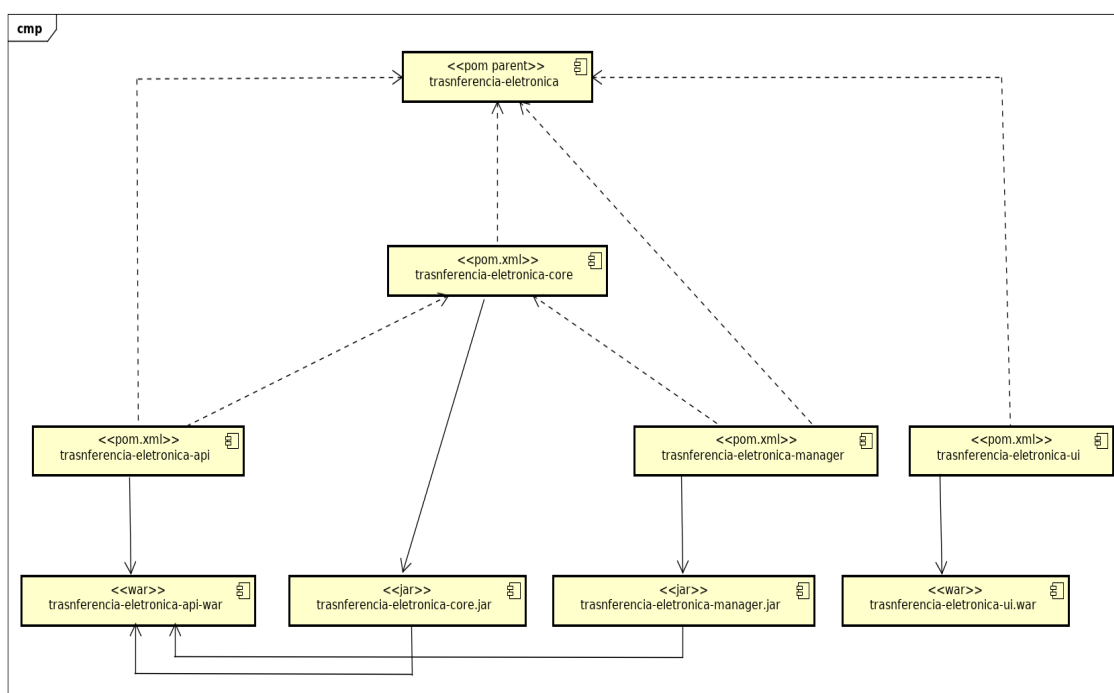


Figura 1: Diagrama de componentes

A figura 1 podemos observar a arquitetura alcançada através do maven. Existe um pom *parent* comum a todos os módulos que gerencia o projeto inteiro e existe o pom para cada módulo do projeto. O módulo `transferencia-eletronica-core` apresenta toda a regra de negócio da aplicação, o módulo `transferencia-eletronica-api` apresenta toda a interface do serviço com as aplicações clientes. O módulo `transferencia-eletronica-ui` apresenta a interface gráfica da aplicação e por fim o módulo `transferencia-eletronica-manager` apresenta o gerenciador do serviço.

| | | |
|--------------------------------------|------------------------------------------------------|---------------------------------------|
| Solicitante: Reserva Fácil | Projeto/Sistema: Prova de programação Java | Atividade: Desenvolvimento |
| Data: 11/09/2018 | Funcionalidade: Agendamento Eletrônico | Responsável: Samuel Santana |

A arquitetura gera dois artefatos .war, um para a api e o outro para interface gráfica, consequentemente existem dois artefatos que precisam ser implantados o transferencia-eletronica-api.war e o transferencia-eletronica-extranet-ui .war.
Os testes se encontram dentro do módulo transferencia-eletronica-api.

Geração do Build

Para a geração do build é necessário utilizar o comando **mvn clean install** e perceber o seguinte retorno:

```
[INFO]transferencia-eletronica ..... SUCCESS [ 1.376 s]
[INFO] transferencia-eletronica-core ..... SUCCESS [ 3.755 s]
[INFO] transferencia-eletronica-manager ..... SUCCESS [ 1.392 s]
[INFO] transferencia-eletronica-api ..... SUCCESS [ 3.131 s]
[INFO] transferencia-eletronica-extranet-ui ..... SUCCESS [ 0.403 s]
```

Após a execução do comando, será necessário implantar o artefato transferencia-eletronica-api.war, caso seja desejado utilizar a interface gráfica o artefato transferencia-eletronica-extranet-ui.war deverá ser implantado também.

INTERFACE GRÁFICA

Figura 2: Aplicação sem Agendamento

A home da aplicação gráfica, sem agendamento, é exibida na figura 2.

| | | |
|--------------------------------------|------------------------------------------------------|---------------------------------------|
| Solicitante: Reserva Fácil | Projeto/Sistema: Prova de programação Java | Atividade: Desenvolvimento |
| Data: 11/09/2018 | Funcionalidade: Agendamento Eletrônico | Responsável: Samuel Santana |

Transferência Eletrônica

| Nova Transferência | | | | | |
|--------------------|--------------|------------------|--------------------|-------|---------------|
| Conta Destino | Conta Origem | Data Agendamento | Data Transferencia | Valor | Valor da Taxa |
| 54321y | 12345x | 10/09/2018 | 10/09/2018 | 10 | 3.3 |
| 54321y | 12345x | 10/09/2018 | 20/09/2018 | 10 | 120 |
| 54321y | 12345x | 10/09/2018 | 30/09/2018 | 10 | 0.8 |
| 12345y | 12345x | 10/09/2018 | 10/10/2018 | 10 | 0.6 |
| 54321y | 12345x | 10/09/2018 | 20/10/2018 | 10 | 0.4 |
| 54321y | 12345x | 10/09/2018 | 21/09/2018 | 10 | 0.8 |

Figura 3: Aplicação com Agendamento

A home da aplicação gráfica, com agendamento ,é exibida na figura 3.

EXTRA

A aplicação apresenta um módulo de gerenciamento, que resumidamente, monitora todas as requisições e exceções que ocorrem durante o processamento da requisição. O módulo utiliza programação orientada a aspectos, ou seja, atua de forma transversal ao fluxo principal da aplicação. A figura 4 exhibe o comportamento do módulo.

| | | |
|--------------------------------------|------------------------------------------------------|---------------------------------------|
| Solicitante: Reserva Fácil | Projeto/Sistema: Prova de programação Java | Atividade: Desenvolvimento |
| Data: 11/09/2018 | Funcionalidade: Agendamento Eletrônico | Responsável: Samuel Santana |

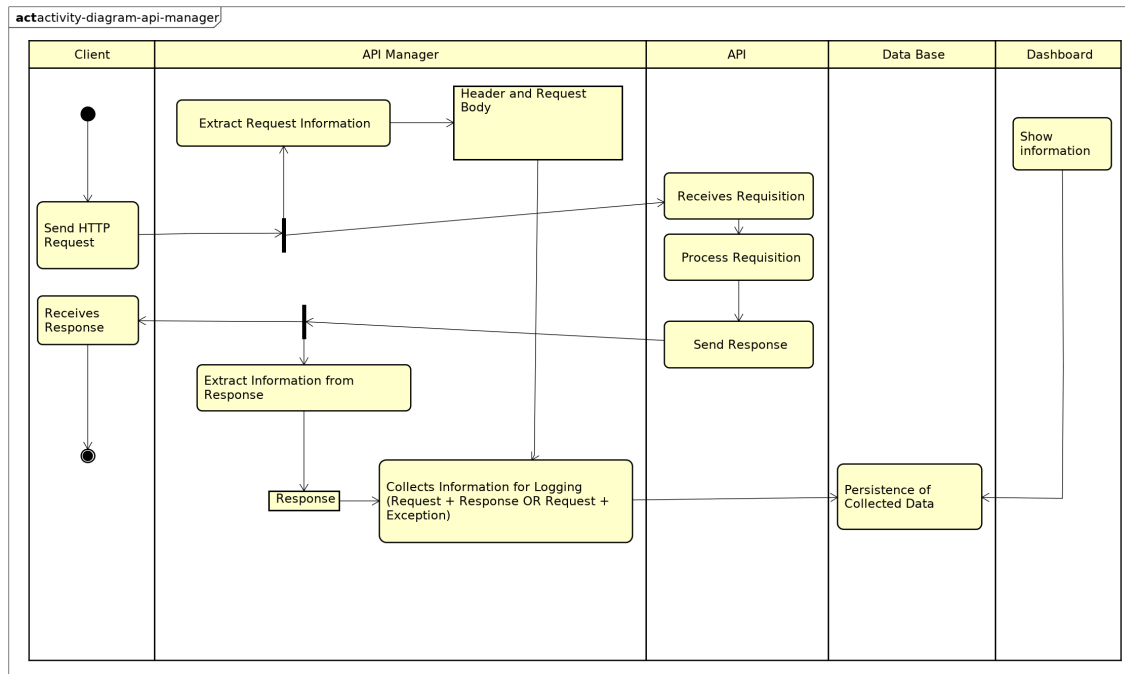


Figura 4: Diagrama de atividade do monitoramento

DETALHAMENTO DE SERVIÇOS DE MONITORAMENTO

| | | |
|----------------------------------|------------------------------------------------------------------------|-----------------------------------------------------------------------------------------|
| Nome do Serviço: | recuperarMonitoramento | |
| Endereço de acesso: | http://localhost:7001/transfereencia-eletronica-api/api/administrativo | |
| Versão: | 1.00 | |
| Tipo: | GET | |
| Funcionalidade | Nome da Operação | Objetivo |
| Recuperar informações coletadas. | <i>Recuperar Monitoramento</i> | Recuperar as informações das requisições que foram monitoradas durante o processamento. |

| | | |
|--------------------------------------|------------------------------------------------------|---------------------------------------|
| Solicitante: Reserva Fácil | Projeto/Sistema: Prova de programação Java | Atividade: Desenvolvimento |
| Data: 11/09/2018 | Funcionalidade: Agendamento Eletrônico | Responsável: Samuel Santana |

PARÂMETROS DE ENTRADA E DE SAÍDA

Parâmetros de Saída (quando retorno for sem erro):

| Nome | Tipo | Descrição |
|-----------------------|---------|-------------------------------------------------------------------------------------|
| url | String | Url do recurso solicitado |
| httpstatus | Integer | Código do retorno HTTP da requisição |
| quantidadeRequisicoes | Integer | Quantidade de transferências encontradas |
| requisicoes | List | Lista das requisições |
| data | String | Atributo do objeto requisição para a Data da transferência |
| tempoExecucao | String | Atributo do objeto requisição que indica o tempo de processamento da requisição |
| ipOrigem | String | Atributo do objeto requisição que indica o IP de origem da requisição |
| tipo | String | Atributo do objeto requisição que indica o tipo de requisição (GET,POST,PUT,DELETE) |
| parametro | Object | Atributo do objeto requisição |
| header | String | Atributo do objeto parametro que indica o header da requisição |
| entrada | String | Atributo do objeto parametro que indica os dados de entrada da requisição |
| saida | String | Atributo do objeto parametroque indica os dados de saída da requisição |
| metodoInvocado | String | Atributo do objeto parametro que indica o método invocado da requisição |
| classeInvocada | String | Atributo do objeto parametro que indica a classe invocada da requisição |
| erro | Object | Atributo do objeto requisição |
| motivo | String | Atributo do objeto erro que indica o motivo do erro |
| classe | String | Atributo do objeto erro que indica a classe em que ocorreu o erro |
| metodo | String | Atributo do objeto erro que indica o método que deu erro |
| stacktrace | String | Atributo do objeto erro que indica o stack trace do erro |

EXEMPLO DE REQUISIÇÃO

| Entrada | Saída |
|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | <pre>{ "url": "/transferencia-eletronica-api/api/administrativo", "httpStatus": 200, "quantidadeRequisicoes": 6, "requisicoes": [{ "data": 1536634800000, "tempoExecucao": 102, "ipOrigem": "0:0:0:0:0:0:1", "tipo": "POST", "parametro": { "header": "[{\\n \\\"descricao\\\" : \\\"Content-Type\\\",\\n \\\"valor\\\" : \\\"application/json\\\"\\n}, {\\n \\\"descricao\\\" : \\\"cache-control\\\",\\n \\\"valor\\\" : \\\"no-cache\\\"\\n}, {\\n \\\"descricao\\\" : \\\"Postman-Token\\\",\\n \\\"valor\\\" : \\\"ea1feb8b-792f-4719-bd74-6e7d42aaff73\\\"\\n}, {\\n \\\"descricao\\\" : \\\"User-</pre> |

| | | |
|--------------------------------------|------------------------------------------------------|---------------------------------------|
| Solicitante: Reserva Fácil | Projeto/Sistema: Prova de programação Java | Atividade: Desenvolvimento |
| Data: 11/09/2018 | Funcionalidade: Agendamento Eletrônico | Responsável: Samuel Santana |

```

Agent\",\\n  \\\"valor\\\" : \\\"PostmanRuntime/7.1.1\\\"
\\n\\n},
{\\n  \\\"descricao\\\" : \\\"Accept\\\",\\n  \\\"valor\\\" : \\\"*/
*\\\"\\n}, {\\n  \\\"descricao\\\" : \\\"Host\\\",\\n  \\\"valor\\\" :
\\\"localhost:7001\\\"\\n},
{\\n  \\\"descricao\\\" : \\\"cookie\\\",\\n  \\\"valor\\\" : \\\"JS
SESSIONID=6409929E7DC04C3204964DD971
7C39C3\\\"\\n}, {\\n  \\\"descricao\\\" : \\\"accept-
encoding\\\",\\n  \\\"valor\\\" : \\\"gzip, deflate\\\"\\n}, {\\n
\\\"descricao\\\" : \\\"content-length\\\",\\n  \\\"valor\\\" :
\\\"115\\\"\\n},
{\\n  \\\"descricao\\\" : \\\"Connection\\\",\\n  \\\"valor\\\" :
\\\"keep-alive\\\"\\n} ]\",

      \"entrada\":
      \"{\\n  \\\"dataTransferencia\\\" : \\\"09/09/2018\\\",\\n  \\
\\\"dataAgendamento\\\" : null,\\n  \\\"valor\\\" : 10.0,\\n
\\\"contaOrigem\\\" : \\\"104736\\\",\\n  \\\"contaDestino
\\\" : \\\"204789\\\",\\n  \\\"valorTaxa\\\" : null\\n}\",
      \"saida\": \"{\\n  \\\"headers\\\" :
{ },\\n  \\\"body\\\" : {\\n    \\\"mensagem\\\" : \\\"Não foi
encontrado taxa para os parâmetros
informados!\\\",\\n    \\\"status\\\" : \\\"BAD_REQUEST\\\",\\n
    \\\"statusCode\\\" : 400\\n  },\\n  \\\"statusCode\\\" : \\\"BAD_REQUEST\\\"
\\\"\\n}\",

      \"metodoInvocado\":
      \"criarAgendamento\",

      \"classeInvocada\":
      \"sls.transferenciaeletronica.api.rest.transferencia
.TransferenciaRecurso\"
    },
    \"erro\": null
  }
}

```

OBS: Por fim, segue em anexo o arquivo, *transferencia-eletronica.postman_collection.json*, que contém a collection do Postman para facilitar os testes, lembrando que será necessário alterar o valor do atributo data de transferência.