Agência para a Modernização Administrativa I.P.

Fatura Sem Papel

Documento de Integração

Versão 1.3

AGÊNCIA PARA A MODERNIZAÇÃO ADMINISTRATIVA

Referências a outros Documentos

Ref.	Descrição	Autor
-	-	-

Registo de Revisões

Data	Versão	Descrição	Autor
28-10-2022	1.0	Documento Inicial	AMA
02-11-2022	1.1	Documento com updates e correções	AMA
09-11-2022	1.2	• Adicionado fluxo típico; • Alteração ao fluxo criação de conta; • Alteração ao fluxo atualizar token.	AMA
15-11-2022	1.3	• Correções e atualizações	AMA

Índice

1 Introdução

O projeto Fatura Sem Papel (FSP) visa o desenvolvimento, implementação e suporte ao funcionamento de uma aplicação que permita a todos comerciantes, que reúnam as condições necessárias, enviar faturas eletronicamente, via e-mail, a todos os cidadãos/empresas que tenham demonstrado interesse em assim as receber ao aderir ao serviço.

Pretende-se assim contribuir para a desmaterialização das faturas, contribuindo não só para uma redução de custos evidente como também para a diminuição do consumo de papel, com claras vantagens e impactos positivos no contexto ambiental.

Este documento detalha os fluxos e especifica os serviços da FSP. Para além disso, aborda também outros temas importantes como o processo de integração.

1.1 Definições, Acrónimos e Abreviações

FSP – Fatura Sem Papel

SCAP – Sistema de Certificação de Atributos Profissionais

FA – Fornecedor de Autenticação

AMA – Agência para a Modernização Administrativa

CC – Cartão de Cidadão

CMD – Chave Móvel Digital

2 Arquitetura da Solução

A Fatura Sem Papel (FSP) está inserida no ecossistema Autenticacao.Gov (ver Figura 1), tirando proveito das funcionalidades de sistemas já existentes. Nomeadamente:

1. Fornecedor de Autenticação (FA) – responsável pela autenticação de cidadãos, podendo os cidadãos utilizar a Chave Móvel Digital (CMD) ou o Cartão de Cidadão (CC) para proceder à sua autenticação. Após correta autenticação, o FA comunica com a FSP para criação de conta de assinatura de faturas eletrónicas;
2. Sistema de Certificação de Atributos Profissionais (SCAP) – responsável pela gestão e obtenção de atributos, em particular, os empresariais de cidadãos. A FSP comunica com o SCAP para verificar se um cidadão tem o atributo necessário para criar uma conta de envio de faturas eletrónicas.

A FSP integra com estes dois sistemas no fluxo de criação de conta (ver 4.1.1). Este fluxo é iniciado pelo Software de Faturação, comunicando com o FA.

No que diz respeito aos fluxos de assinatura, são também iniciados pelo Software de Faturação, comunicando diretamente com a FSP (ver 4.2).

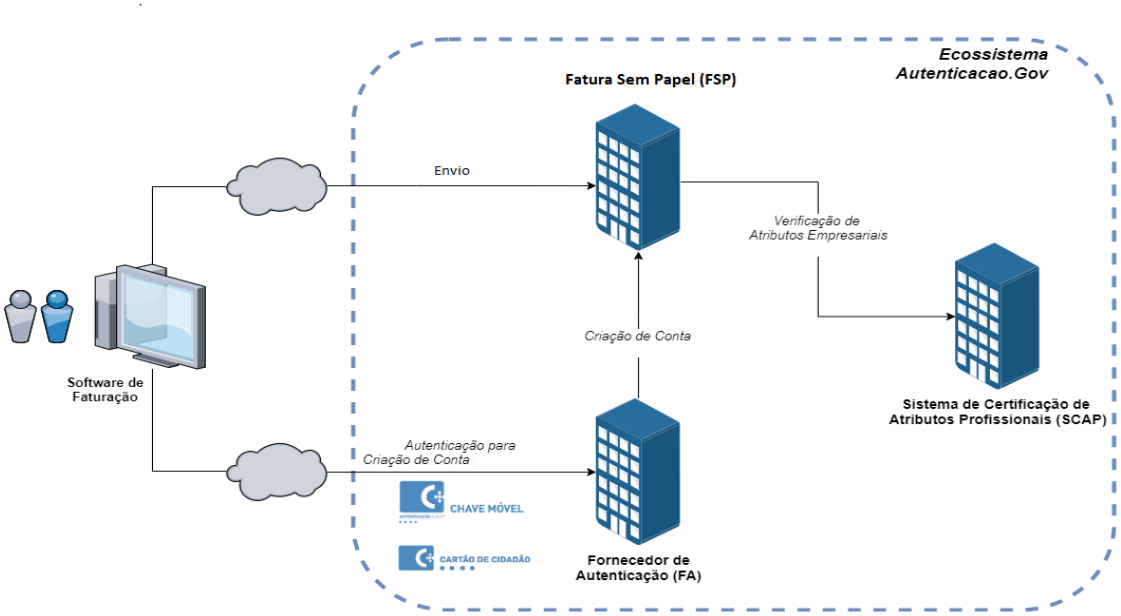


Figura 1. Ecossistema Autenticacao.Gov

3 Requisitos para utilização

3.1 Requisitos para a Empresa (ou Entidade)

- Acesso à Internet;
- Colaborador da empresa com poderes para emitir, que detenha:
    - Chave Móvel Digital + PIN de autenticação ou Cartão de Cidadão + PIN de autenticação + Leitor de Cartões;
    - Atributo “Emissão de faturas” ativo no SCAP na empresa para a qual pretende criar conta de envio de faturas.

3.2 Requisitos para o Software de Faturação

- Efetuar integração com FSP e FA;
- Passar pelo Processo de Integração e credenciação (consultar secção 6 Processo de Certificação).

4 Fluxos

Esta secção descreve os fluxos necessários para que Softwares de Faturação possam integrar com o FSP.

4.1 Fluxos de gestão de conta

4.1.1 Criação de Conta

A criação de uma conta para envio de faturas através da FSP é feita via Fornecedor de Autenticação (FA) que, após a devida autenticação do cidadão (colaborador da empresa com poderes para enviar faturas), encaminha o pedido de criação de conta de envio para a FSP. Assim, neste fluxo, o Software de Faturação comunica apenas com o cidadão e com o FA.

4.1.2 Identificador de conta

As contas para envio de faturas pela FSP são identificadas univocamente pelas seguintes componentes:

- Identificador do cidadão;
- NIPC da empresa;
- Email associado à conta
- Nome
- Identificador da instância (UUID – consultar secção 5 Geração de UUID)

4.1.3 Autenticação do Cidadão

O cidadão autentica-se perante o FA, recorrendo a um dos seguintes meios:

- Chave Móvel Digital (CMD);
- Cartão de Cidadão (CC).

Mais informação sobre autenticações com CMD e CC pode ser encontrada em:

- https://www.autenticacao.gov.pt/chave-movel-digital/autenticacao
- https://www.autenticacao.gov.pt/cartao-cidadao/autenticacao

4.1.4 Elegibilidade do Cidadão (Comerciante)

A validação da elegibilidade de um cidadão para criação de uma conta como representante de uma empresa é feita pelo Sistema de Certificação de Atributos Profissionais (SCAP), através da existência do atributo ativo "Emissão de faturas".

4.1.5 Fluxo de Criação de conta

O diagrama da Figura 2 ilustra o processo de criação de conta de envio na FSP. São em seguida descritos os passos deste fluxo:

1. Cidadão pede adesão ao serviço de envio de faturas, introduzindo os seguintes dados:
   1.1. NIPC da empresa associada à conta – obrigatório (9 dígitos);
   1.2. Email associado à conta – obrigatório;
   1.3. Nome – obrigatório;
   1.4. Identificador da instância – obrigatório (consultar secção 5 Geração de UUID).
2. Software de Faturação invoca FA para o cidadão se poder autenticar. Esta autenticação será feita através do protocolo OAuth2 e devem ser pedidos os seguintes atributos:
   2.1. http://interop.gov.pt/MDC/Cidadao/NIC (se cidadão português)
   2.2. http://interop.gov.pt/MDC/Cidadao/DocType1 (se cidadão estrangeiro)
   2.3. http://interop.gov.pt/MDC/Cidadao/DocNationality1 (se cidadão estrangeiro)
   2.4. http://interop.gov.pt/MDC/Cidadao/DocNumber1 (se cidadão estrangeiro)
   2.5. http://interop.gov.pt/MDC/Cidadao/NomeProprio
   2.6. http://interop.gov.pt/MDC/Cidadao/NomeApelido
   2.7. http://interop.gov.pt/FSP/createFSPAccount?enterpriseNipc=<enterprise Nipc>$email=<email>$instanceId=<instanceId>$creationClientName=<*creationClientName (os valores entre <> devem ser substituídos pela informação introduzida no passo 1). No caso de alguma destas informações conter espaços em branco, os parâmetros do atributo (tudo o que vem depois do '?'), deve ser convertido numa string base64.*
3. FA mostra a página de autenticação no mecanismo utilizado pelo Software de Faturação (e.g. WebView ou Browser);
4. Cidadão efetua autenticação com CMD ou CC;
5. Página da autenticação envia dados para o FA;
6. FA valida a autenticação;
7. FA pede criação de conta de envio de fatura, enviando para a FSP os dados obtidos na autenticação;
8. FA devolve um token OAuth associado à autenticação efetuada;
9. Software de Faturação verifica token OAuth associado à autenticação efetuada;
10. Software de Faturação obtém token OAuth associado à autenticação efetuada;

11. FSP pede ao SCAP os atributos empresariais do cidadão na empresa para a qual pretende criar conta de envio;
12. SCAP devolve atributos empresariais do cidadão na empresa para a qual pretende criar conta de envio;
13. FSP valida se o cidadão tem o atributo “Emissão de Faturas” na empresa para a qual pretende criar conta de assinatura;
14. FSP cria conta de envio;
15. Software de Faturação invoca FA com o token OAuth obtido no passo 10, de forma a obter a informação sobre a conta de envio de fatura. Antes de fazer esta invocação, o Software de Faturação deve esperar **15 segundos**;
16. FA valida token OAuth recebido;
17. FA pede informação de conta de envio de fatura;
18. FSPdevolve informação de conta de assinatura para o FA.
19. FA devolve informação de conta de assinatura para o Software de Faturação
20. Software de Faturação guarda informação de conta de envio de fatura;
21. Software de Faturação mostra mensagem de sucesso ao cidadão.

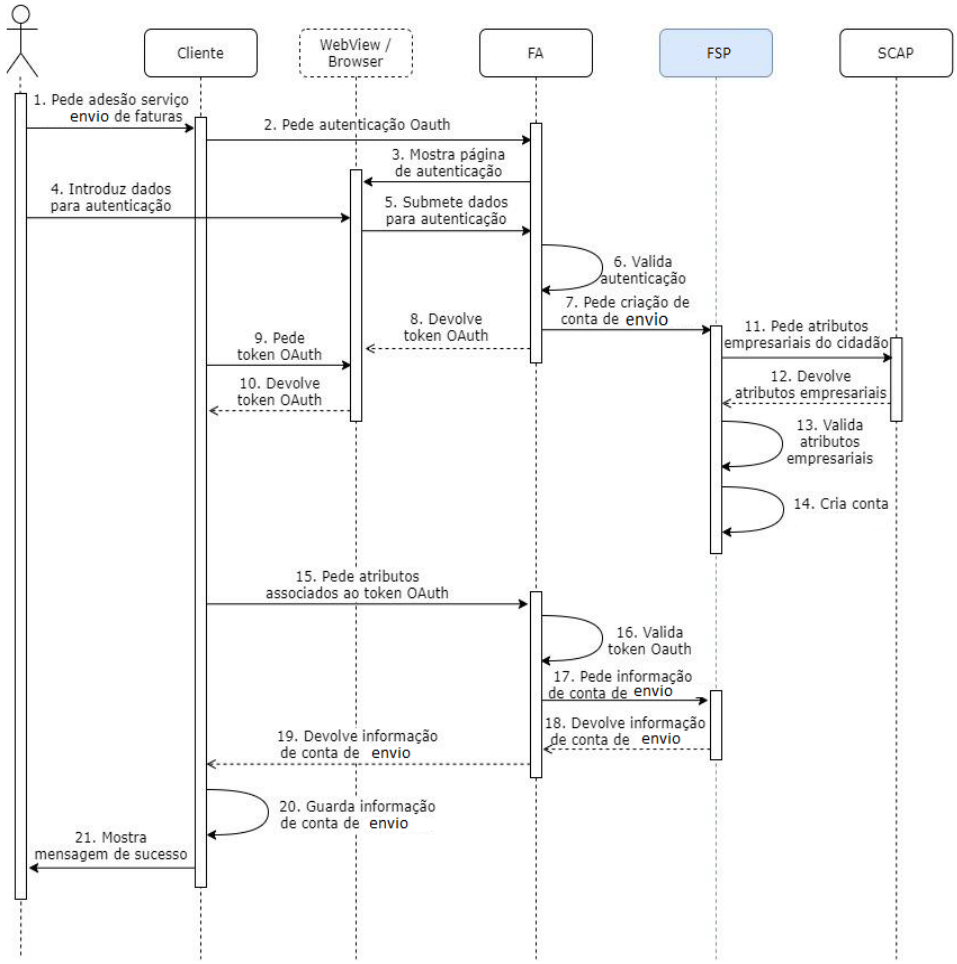


Figura 2. Fluxo de criação de conta

4.1.6 Estrutura do pedido de informação de conta

O pedido deverá ser efectuado utilizando a seguinte especificação:

Endpoint:<url base da api>/Authorize/createFSPAccount?enterpriseNipc=<enterpriseNipc>&email=<email>&instanceId&creationClientName<creationClientName>

Método HTTP: POST

Payload: <JSON contendo os parâmetros de envio>

Os parâmetros necessários deverão ser enviados no corpo do pedido HTTP, em formato JSON e são os seguintes:

```
{
  citizenDocId: string,
  citizenDocType: string,
  docNationality: string,
  citizenDocCountry: string,
  citizenGivenName: string
  citizenLastName: string
}
```

O FSP deverá receber também como query string os seguintes dados:

- enterpriseNipc: string
- email: string
- instanceId: string
- creationClientName: string

4.1.7 Estrutura da resposta de informação de conta

O FA devolve a informação de conta para o Software de Faturação. Esta informação é enviada em formato JSON como valor do atributo http://interop.gov.pt/FSP/createFSPAccount. Em caso de sucesso na criação de conta, é enviada uma string base64, cuja estrutura é um JSON com as seguintes propriedades

- Token de acesso às operações de assinatura (accessToken);
- Token para atualização de tokens (refreshToken);
- Data de expiração do refresh token (expirationDate).

{

accessToken: string,

refreshToken: string,

expirationDate: DateTime (yyyy-MM-ddTHH:mm:ss.fffffffZ)

}

Em caso de erro na criação de conta, são enviados os atributos:

- Erro (error);
- Descrição do erro (error_description);

As causas possíveis para se obter um erro, são (error – error_description):

- Bad Request - Invalid parameter citizenDocId
- Bad Request - Missing parameter citizenDocId
- Bad Request - Invalid parameter citizenDocType
- Bad Request - Missing parameter citizenDocType
- Bad Request - Invalid parameter citizenDocCountry
- Bad Request - Missing parameter citizenDocCountry
- Bad Request - Invalid parameter enterpriseNipc

- Bad Request - Missing parameter enterpriseNipc
- Bad Request - Missing parameter citizenGivenName
- Bad Request - Missing parameter citizenLastName
- Bad Request - Invalid parameter email
- Bad Request - Missing parameter creationClientName
- Bad Request - Client is not active
- Missing required enterprise attributes - The citizen attributes obtained are not valid
- Internal Server Error - error_description: Unexpected error while processing client request

4.1.8 Cancelamento

O pedido de cancelamento de conta deverá ser efectuado utilizando a seguinte especificação:

Endpoint: <url base da API>/Account

Método HTTP: DELETE

Headers: Authorization : <access token>

A Figura 3 ilustra o processo cancelamento de uma conta de assinatura:

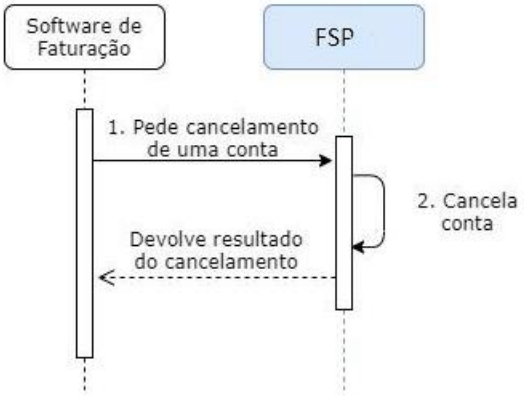


Figura 3 – Fluxo de Cancelamento de Conta.

Em caso de sucesso no pedido de cancelamento da conta, é devolvido um JSON com as seguintes propriedades:

- Resultado da operação(result);

Exemplo:

```
{
  result: string
}
```

Em caso de erro na criação de conta, são enviados os atributos:

- Erro (error);
- Descrição do erro (error_description).

As causas possíveis de erro são (error – error_description):

- Unauthorized - Invalid token
- Bad Request - Invalid user
- Internal Server Error - error_description: Unexpected error while processing client request

4.2 Tokens

4.2.1 Access Tokens

Token necessário para efetuar operações de envio no FSP. Este token é do tipo Bearer e deve ser passado no header Authorization dos pedidos.

Por uma questão de segurança, o AccessToken tem uma validade reduzida (24 horas), definida pela FSP. Sempre que for invocado um método do FSP com um AccessToken expirado, a FSP retorna um erro HTTP 400 Bad Request, com a mensagem de erro "The access or refresh token is expired or has been revoked". Nestes casos, o Software de Faturação deve invocar o método /Token, de modo a ser gerado um novo accessToken. Estes novos tokens devem ser utilizados nas invocações futuras ao serviço.

4.2.2 Refresh Tokens

Token necessário para invocar o método /Token. Este token é do tipo Bearer e deve ser passado no header Authorization dos pedidos. O resultado da invocação do método /Token é a geração de um novo accessToken. Estes novos tokens devem ser utilizados nas invocações futuras ao serviço.

Sempre que for invocado um método do FSP com um RefreshToken expirado, a FSP retorna um erro HTTP 400 Bad Request, com a mensagem de erro "The access or refresh token is expired or has been revoked". Nestes casos, o Software de Faturação deve voltar a obter novos tokens através do fluxo de criação de conta.

4.2.3 Atualizar Token

Método que retorna um novo *AccessToken* e um novo *RefreshToken* para uma conta de envio. Estes novos tokens devem ser utilizados nas invocações futuras aos serviços.

Este método deve ser invocado sempre que o sistema retorne o erro HTTP **400 Bad Request**, com a mensagem de erro "*The access or refresh token is expired or has been revoked*". A Figura 4 ilustra o processo de atualização de tokens.

O pedido deverá ser efectuado utilizando a seguinte especificação:

Endpoint: <*url base da API*>/Token?accesstoken={access_token}&refreshtoken={refresh_token}

Método HTTP: *PUT*

Headers: Authorization : <access token>

Os parâmetros necessários são:

- accesstoken - token de acesso actual do comerciante *(obrigatório)*
- refreshtoken- refresh token do comerciante *(obrigatório)*

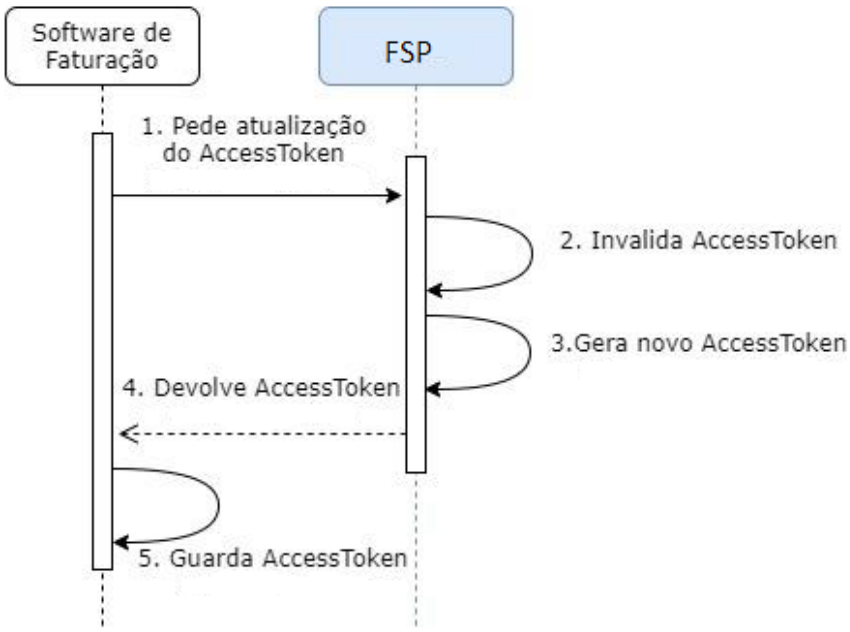


Figura 4 - Fluxo de Atualização de Token

Em caso de sucesso no pedido de atualização , é devolvido um JSON com as seguintes propriedades:

- Novo token de acesso(access_token);
- Refresh(refresh_token);

Exemplo:

```
{
  access_token: string,
  refresh_token: string
}
```

Em caso de erro são enviados os atributos:

- Erro (error);

- Descrição do erro (error_description).

As causas possíveis de erro são (error – error_description):

- Unauthorized - Invalid token
- Bad Request - Invalid parameter accessToken
- Bad Request - Invalid parameter refreshToken
- Bad Request - Missing parameter accesToken
- Bad Request - Missing parameter refreshToken
- Bad Request - Invalid user
- Internal Server Error - error_description: Unexpected error while processing client request

4.3 Fluxos de Envio

4.3.1 Obter Cifra

O pedido para obter a cifra de um cidadão , que será utilizada para proteger os pdf que serão enviados pelo software de faturação, deverá ser efetuado utilizando a seguinte especificação:

Endpoint: <*url base da API*>/Cypher?id={citizen_doc_id}&type={citizen_doc_type}

Método HTTP: *GET*

Headers: Authorization : <access token>

Os parâmetros necessários são:

- id - id do documento de identificação do cidadão *(obrigatório)*
- type - tipo de documento de identificação *(obrigatório)*

A Figura 5 ilustra o pedido para obter a cifra de um cidadão.

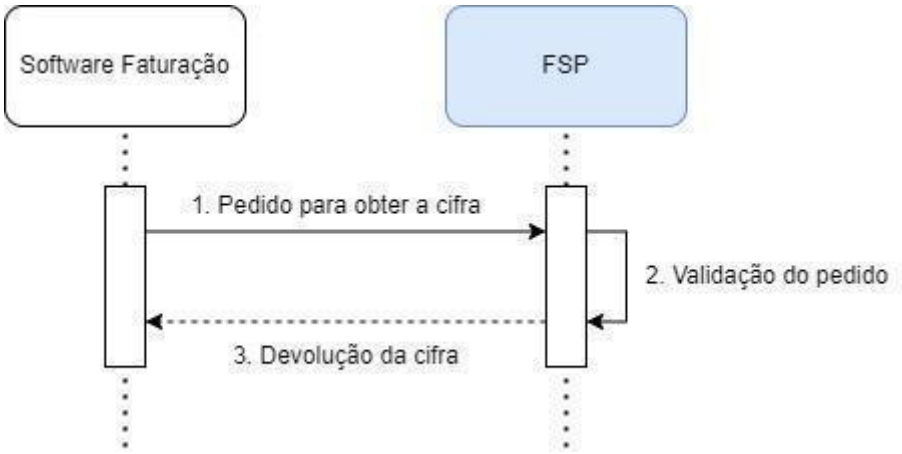


Figura 5 - Fluxo de pedido de cifra.

Em caso de sucesso no pedido de obtenção da cifra, é devolvido um JSON com as seguintes propriedades:

- Identificador da instância (instanceId);
- Cifra (cypher);

No caso em que o cidadão/empresa não definiu nenhuma cifra no FSP este campo deverá vir a null.

Exemplo:

```
{
  instanceId: uuid,
  cypher: string
}
```

Em caso de erro na criação de conta, são enviados os atributos:

- Erro (error);
- Descrição do erro (error_description)

As causas possíveis de erro são (error – error_description):

- Unauthorized - Invalid token
- Bad Request - Invalid parameter citizenDocId
- Bad Request - Invalid parameter citizenDocType
- Bad Request - Invalid user
- Internal Server Error - error_description: Unexpected error while processing client request

4.3.2 Envio

O pedido para envio de faturas deverá ser efectuado utilizando a seguinte especificação:

Endpoint: <*url base da API*>/Invoice

Método HTTP: *POST*

Headers: Authorization : <access token>

Payload: <JSON contendo os parâmetros de envio>

Os parâmetros necessários deverão ser enviado no corpo do pedido HTTP, em formato JSON e são os seguintes:

- clientId - id do documento de identificação fiscal do cidadão *(obrigatório)*
- enterpriseNipc - id do documento de identificação do comerciante *(obrigatório)*

- invoice - PDF da fatura (no formato Base64) *(obrigatório)*
- fileName - descrição da fatura *(obrigatório)*
- localId - Identificador da fatura por parte do Software de Faturação

Exemplo:

```
{
  clientId: string,
  enterpriseNipc: string,
  invoice: base64,
 description: string,
 localId: string
}
```

A Figura 6 ilustra o pedido de envio.

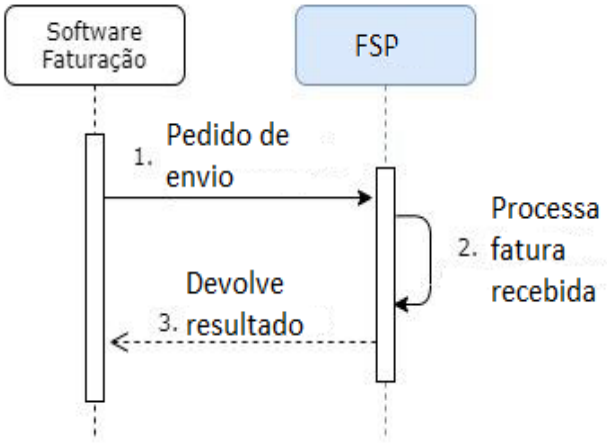


Figura 6 - Fluxo de Pedido de Envio

Em caso de sucesso no pedido de envio da fatura, é devolvido um JSON com as seguintes propriedades:

- Identificador da fatura no sistema FSP (id);
- Resultado da operação(result);

Exemplo:

```
{
```

id: uuid,

result: string

}

Em caso de erro na criação de conta, são enviados os atributos:

- Erro (error);
- Descrição do erro (error_description)

As causas possíveis de erro são (*error – error_description*):

a) Unauthorized - Invalid token

b) Bad Request - Invalid parameter clientId

c) Bad Request - Missing parameter clientId

d) Bad Request - Invalid parameter enterpriseNipc

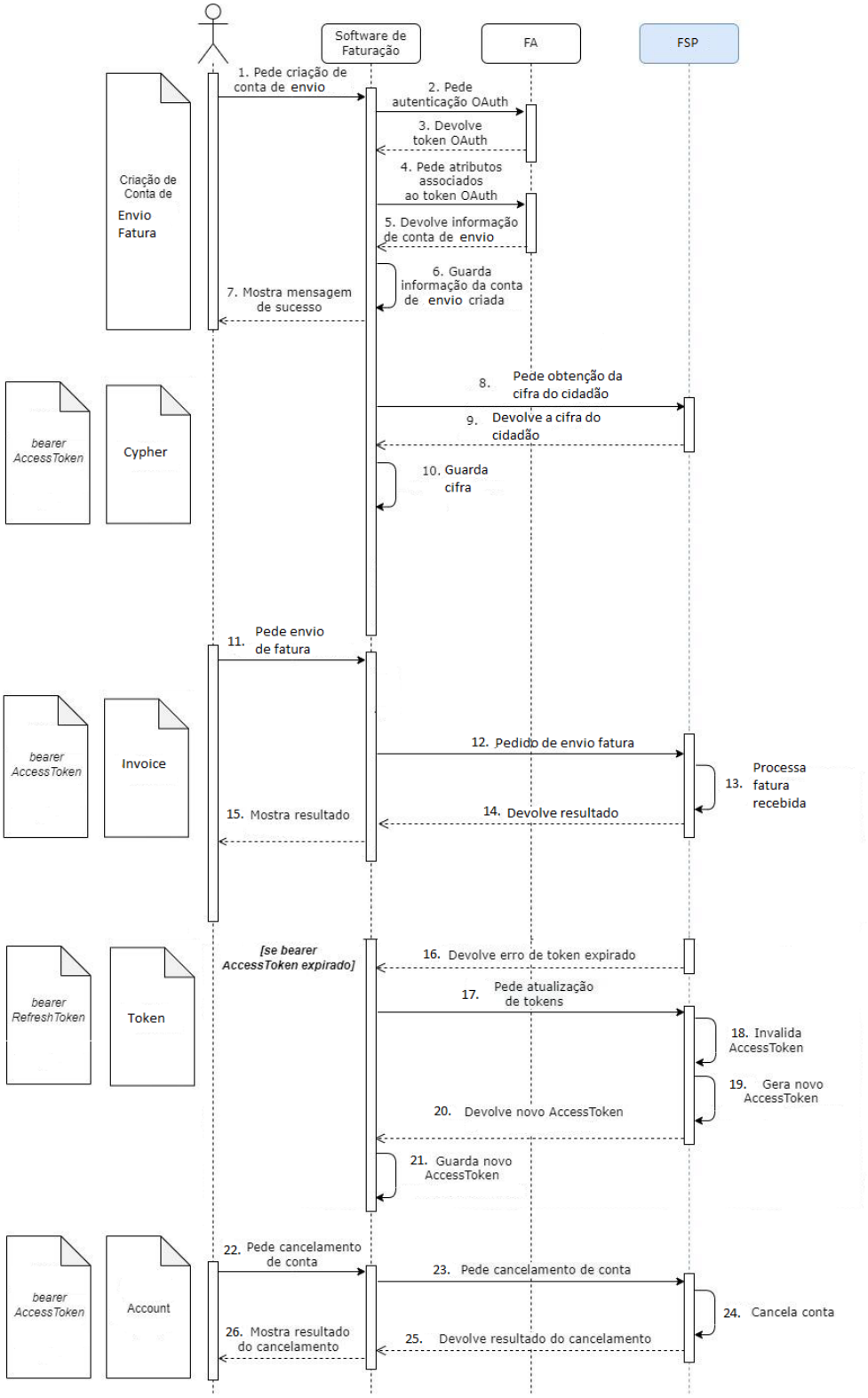
e) Bad Request - Missing parameter enterpriseNipc

f) Bad Request - Invalid parameter invoice

g) Bad Request - Missing parameter invoice

h) Internal Server Error - error_description: Unexpected error while processing client request

4.4 Fluxo Típico



5 Geração de UUID

Um Universally Unique IDentifier (UUID) é um número de 16 octetos (128 bits).

Na sua forma canónica, um UUID é representado por 32 dígitos em formato hexadecimal, exibidos em cinco grupos separados por hífens, na forma 8-4-4-4-12 para um total de 36 caracteres (32 caracteres alfanuméricos e 4 hífens, utilizando exclusivamente letras minúsculas).

Por exemplo:

- 123e4567-e89b-12d3-a456-426655440000 – Corresponde a um UUID.
- 123E4567-E89B-12D3-A456-426655440000 – Não corresponde a um UUID.

A geração deve seguir a norma da especificação (disponível em https://www.ietf.org/rfc/rfc4122.txt).

6 Processo de Certificação

Em atualização.

7 Guidelines de Integração

As guidelines de integração encontram-se disponíveis no repositório da Fatura Sem Papel em https://github.com/amagovpt/doc-FSP