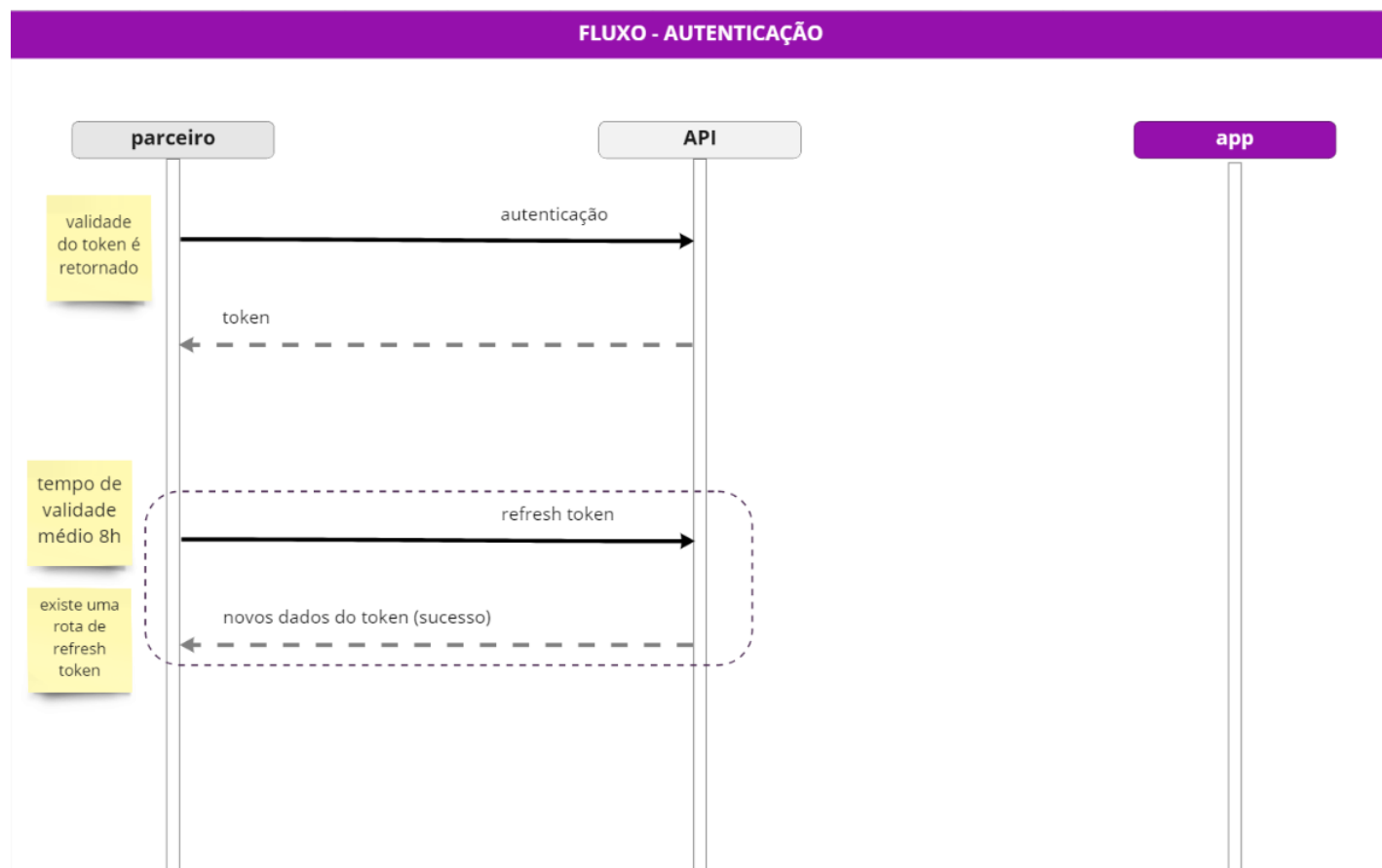


Link da Api no swagger: <https://app.swaggerhub.com/apis-docs/aiq6/api-aiqfome/1.0.0>

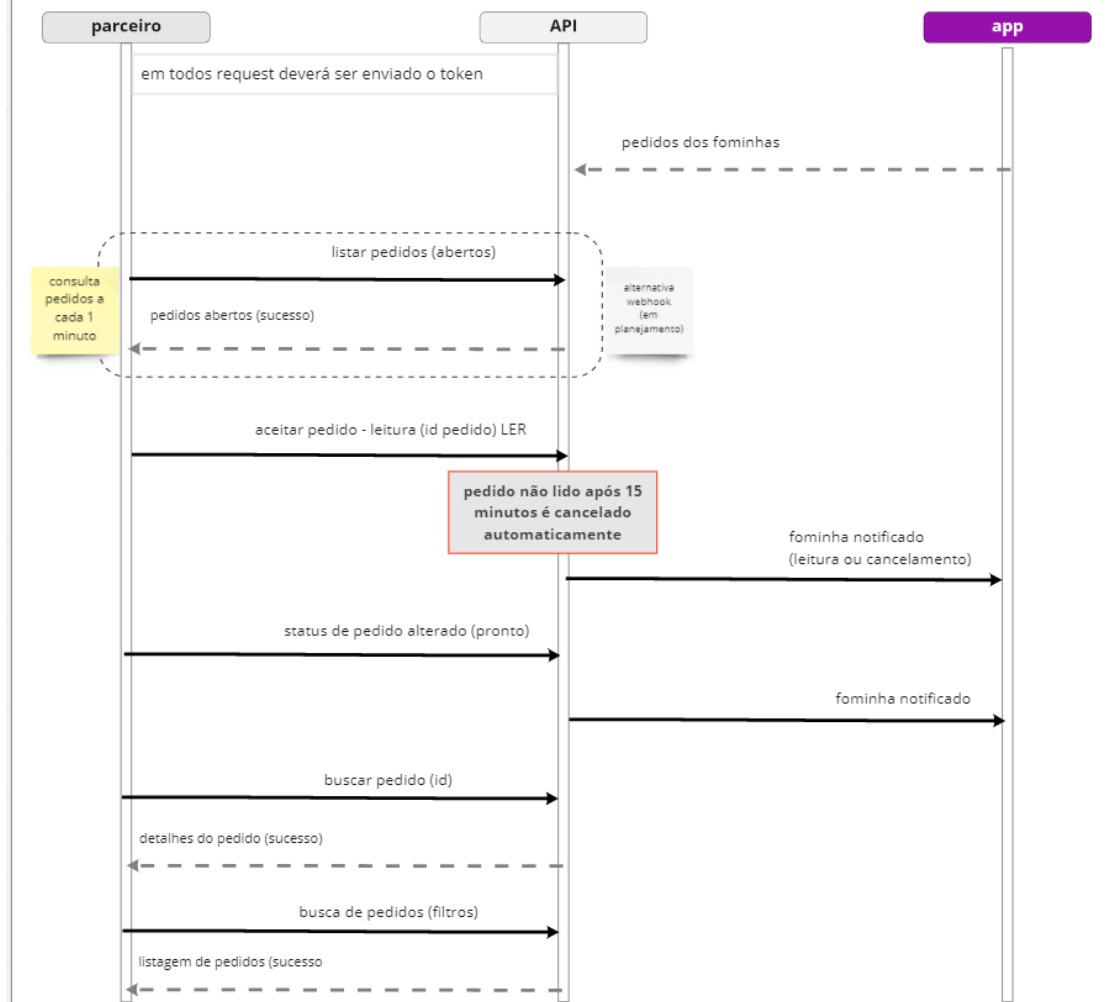
Os fluxos abaixo representa de forma visual os métodos que estão na documentação dentro do link acima do swagger.
Fluxo de funcionamento da API:



FLUXO - STORE



FLUXO - PEDIDOS



Autenticação

auth

POST

(F1_1) Autenticação

host: /alfredo/auth/token
Host para chamadas de auth.

Método HTTP.

- um usuário com várias lojas;
- um usuário por loja;

chave por aplicação (ERP)

```
request body:
{
  "username": "user@store.com",
  "password": "12341234"
}
```

Parâmetros da request.

header: Authorization: Basic
cmFpbXVudG86NTY2ODgzZGMtZDMwYy00OGM0LWE1MzgtZjc1NDZlMwY1NjA4

O header "Authorization" é formado por um **base64** do **CLIENT_ID:CLIENT_SECRET** que passaremos para cada integrador.

```
{
  "data": {
    "access_token": "jwt_access_token",
    "expires_in": 3600,
    "refresh_expires_in": 3600,
    "refresh_token": "jwt_refresh_token",
    "token_type": "bearer",
    "user_token": "jwt_user_token"
  }
}
```

- tempo em segundos

Refresh Token

Rota de renovação do token usado na API.

host: /alfredo/auth/token/refresh
Host para chamadas de auth.method:
POST Método HTTP.

header:

Authorization: Basic
cmFpbXVudG86NTY2ODgzZGMtZDMwYy00OGM0LWE1MzgtZjc1NDZlMwY1NjA4
RefreshToken: jwt_refresh_token

O header "Authorization" é formado por um **base64** do **CLIENT_ID:CLIENT_SECRET** que passaremos para cada integrador.

"RefreshToken" é o token "refresh_token" adquirido ao fazer login.

```
{
  "data": {
    "access_token": "jwt_access_token",
    "expires_in": 3600,
    "refresh_expires_in": 3600,
    "refresh_token": "jwt_refresh_token",
    "token_type": "bearer",
    "user_token": "jwt_user_token"
  }
}
```

Header

Header

Todo endpoint tem um Header onde o campo Aiq-User-Agent deve ser preenchido:

Aiq-User-Agent
string
(header)

Used to identify where the request comes from. Standard value is "company name (contact e-mail)"

Example : Company (user@company.com)

Company (user@company.com)

No Header das requisições, utiliza-se authorization Basic somente nos endpoints "alfredo/auth", para os demais endpoints utiliza-se authorization Bearer, utilizando o access_token proveniente das rotas de autenticação.

Authorization
string
(header)

Example : Bearer
M2tkNlpiVmVQNHZhYmg1MThmRURnYTY=

Bearer M2tkNlpiVmVQNHZhYmg1MThmRUf

store management

GET

(F2_1) Consultar lojas

-> listagem

/alfredo/store

headers:

Authorization: Bearer

cmFpbXVuZG86NTY2ODgzZGMtZDMwYy00OGM0LWE1MzgtZjc1NDZIMWY1NjA4

Aiq-User-Agent: User (user@company.com)

request

- page
- per_page
- filter[id]
- filter[name]
- filter[city_ids]

response

```
{
  "data": [
    {
      "id": 1,
      "name": "nesciunt",
      "status": "STANDBY",
      "city_name": "cedias",
      "uf": "AA"
    },
    {
      "id": 2,
      "name": "laquis",
      "status": "OPEN",
      "city_name": "lamia",
      "uf": "AO"
    }
  ],
  "links": {
    "first": "/?page=1",
    "last": "/?page=1",
    "prev": null,
    "next": null
  },
  "meta": {
    "current_page": 1,
    "from": 1,
    "last_page": 1,
    "path": "/",
    "per_page": "15",
    "to": 2,
    "total": 2
  }
}
```

POST

(F2_2) Abrir loja

/alfredo/store/open

headers:

Authorization: Bearer

cmFpbXVuZG86NTY2ODgzZGMtZDMwYy00OGM0LWE1MzgtZjc1NDZIMWY1NjA4

Aiq-User-Agent: User (user@company.com)

request

```
{
  "store_id": 35,
}
```

response

```
{
  "store_id": 15
  "status": "OPEN"
}
```

POST

(F2_4) Stand by

/alfredo/store/stand-by

headers:

Authorization: Bearer

cmFpbXVuzG86NTY2ODgzZGMtZDMwYy00OGM0LWE1MzgtZjc1NDZIMWY1NjA4

Aiq-User-Agent: User (user@company.com)

request

```
{
  "store_id": 35,
}
```

response

```
{
  "store_id" : 15
  "status": "STANDBY"
}
```

POST

(F2_3) Fechar loja

/alfredo/store/close

headers:

Authorization: Bearer

cmFpbXVuzG86NTY2ODgzZGMtZDMwYy00OGM0LWE1MzgtZjc1NDZIMWY1NjA4

Aiq-User-Agent: User (user@company.com)

request

```
{
  "store_id": 35,
}
```

response

```
{
  "store_id" : 15
  "status": "CLOSED"
}
```

Gestão dos itens

items management

GET

(F3_1) Consultar categorias

/alfredo/menu/{store}/categor

headers:

Authorization: Bearer

cmFpbXVuzG86NTY2ODgzZGMtZDMwYy00OGM0LWE1MzgtZjc1NDZIMWY1NjA4

Aiq-User-Agent: User (user@company.com)

request

- filter[status]
- id

response

The status (available/unavailable) of the menu category.

The id of the store to list menu categories.

```
{
  "data": [
    {
      "id": 1,
      "name": null,
      "description": null,
      "hour_order": null,
      "status": 1
    },
    {
      "id": 2,
      "name": null,
      "description": null,
      "hour_order": null,
      "status": 1
    }
  ]
}
```

PUT

(F3_2) Bloquear itens

/alfredo/menu/{store}/items/{item}/toggle-status

headers:

Authorization: Bearer

cmFpbXVuZG86NTY2ODgzZGMtZDMwYy00OGM0LWE1MzgtZjc1NDZlMwY1NjA4

Aiq-User-Agent: User (user@company.com)

request

- store
- item

response

Item is available if status = 1 and unavailable if status = 0.

```
{
  "data": {
    "id": 1,
    "name": "id",
    "status": 307,
    "internal_code": "sunt",
    "description": "Aliquid magni quae aut quia. Numquam itaque totam numquam. Itaque magnam expedita deserunt quae molestias voluptatem fugit. Sint eveniet asperiores vel ut.",
    "image": "incididunt",
    "highlight": 772110,
    "observations": "velit",
    "menu_category_id": 1,
    "order": 851329065,
    "excluded": 9243674,
    "approve_image": "id",
    "reason_image": "aut",
    "weekdays": 3798021,
    "group": 53353,
    "reason_status": "vero"
  }
}
```

GET

(F3_3) Listar itens

/alfredo/menu/{store}/categories/{category}/items

headers:

Authorization: Bearer

cmFpbXVudG86NTY2ODgzZGMtZDMwYy00OGM0LWE1MgtZjc1NDZlMwY1NjA4

Alq-User-Agent: User (user@company.com)

request

filter[name]

Filter by the complete or substring name of the item.

Example : Burger

filter[status]

Filter by status

(available/unavailable) of the item.

Example : 1

response

```
{
  "data": [
    {
      "id": 399832,
      "name": "Abobrinha",
      "status": 1,
      "internal_code": "",
      "description": "mussarela, abobrinha refogada ",
      "image": "399832.jpg",
      "highlight": 1,
      "observations": "|1|",
      "menu_category_id": 42418,
      "order": 1,
      "excluded": 0,
      "approve_image": "",
      "reason_image": "",
      "weekdays": 0,
      "group": null,
      "reason_status": ""
    },
    {
      "id": 399834,
      "name": "Aliche",
      "status": 1,
      "internal_code": "",
      "description": "Mussarela, rodelas de tomate, aliche, cebola e parmesão",
      "image": "399834.jpg",
      "highlight": 0,
      "observations": "",
      "menu_category_id": 42418,
      "order": 3,
      "excluded": 0,
      "approve_image": "",
      "reason_image": "",
      "weekdays": 0,
      "group": 0,
      "reason_status": ""
    }
  ]
}
```

Gestão de pedidos

GET

(F4_1) Listar pedidos NÃO LIDOS

/alfredo/orders

request

response

sem parâmetros

OBS: se o pedido for cancelado automaticamente (caso não lido em 15 minutos) a loja será fechada. Para voltar a receber pedidos, a loja deverá ser aberta manualmente.

headers:

Authorization: Bearer

cmFpbXVuzG86NTY2ODgzZGMiZDMwYy00OGM0LWE1MzgtZjc1NDZlMwY1NjA4

Aiq-User-Agent: User (user@company.com)

```
{
  "data": [
    {
      "order_id": 6853865,
      "created_at": "2020-07-20 09:11:56",
      "is_ready": false,
      "is_read": false,
      "is_cancelled": false,
      "user_name": "paula",
      "order_delivery_time": "20 - 40",
      "order_is_pickup": false,
      "store_name": "Death Pizza"
    },
    {
      "order_id": 6853869,
      "created_at": "2020-07-21 14:52:25",
      "is_ready": true,
      "is_read": false,
      "is_cancelled": true,
      "user_name": "paula",
      "order_delivery_time": "20 - 40",
      "order_is_pickup": false,
      "store_name": "Death Pizza"
    },
    {
      "order_id": 6853874,
      "created_at": "2020-07-21 14:52:36",
      "is_ready": false,
      "is_read": false,
      "is_cancelled": false,
      "user_name": "paula",
      "order_delivery_time": "20 - 40",
      "order_is_pickup": false,
      "store_name": "Death Pizza"
    }
  ],
  "links": {
    "first": "http://localhost:8889/external/orders?page=1",
    "last": null,
    "prev": null,
    "next": "http://localhost:8889/external/orders?page=2"
  },
  "meta": {
    "current_page": 1,
    "from": 1,
    "path": "http://localhost:8889/external/orders",
    "per_page": 15,
    "to": 15
  }
}
```


GET

(F4_2) Consultar pedido

/alfredo/orders/{order}

headers:

Authorization: Bearer

cmFpbXVuZG86NTY2ODgzZGMtZDMwYy00OGM0LWE1MzgtZjc1NDZlMwY1NjA4

Aiq-User-Agent: User (user@company.com)

OBS.: Caso hajam subitens no pedido (e.g. pizzas de vários sabores) os campos "id" e "sku" do item pai serão nulos, fazendo valer os campos "id" e "sku" de cada subitem.

request

response

id

```
{
  "data": {
    "id": 6853865,
    "delivery_time": "20 - 40",
    "created_at": "2020-07-20 09:11:56",
    "is_ready": false,
    "is_read": false,
    "is_cancelled": false,
    "order_observations": "",
    "is_pickup": false,
    "pickup_at": null,
    "coupon_hash": null,
    "items": [
      {
        "id": null,
        "name": "pizza grande",
        "sku": null,
        "category_name": "Pizza",
        "size": "G",
        "unit_value": 20,
        "promotional_value": null,
        "quantity": 1,
        "discount_tax": null,
        "description": "Pizza grande",
        "observations": "",
        "order_mandatory_items": [
          {
            "group": "molho",
            "name": "molho 1",
            "value": 0.03,
            "sku": "79"
          },
          {
            "group": "Bebida",
            "name": "Coca-Cola",
            "value": 5,
            "sku": "coca350ml"
          }
        ],
        "order_additional_items": [
          {
            "name": "garfo",
            "value": 10,
            "sku": "1"
          }
        ],
        "order_item_subitems": [
          {
            "id": "83312969-7a3c-3118-8aa2-063278647d69",
            "name": "quatro queijos",
            "nome": "quatro queijos",
            "description": "pizza quatro queijos",
            "sku": "4"
          },
          {
            "id": "2e6c008b-b80f-11eb-ad49-06a16d64951c",
            "name": "Calabresa ",
            "nome": "Calabresa ",
            "description": "Molho, mussarela, calabresa, cebola e azeitona.",
            "sku": "5"
          }
        ]
      }
    ]
  }
}
```

```
  ],
  "user": {
    "name": "paula",
    "surname": "daianee",
    "mobile_phone": "(45) 55587-4587",
    "phone_number": "(45) 55587-4587",
    "email": "paula@aiqfome.com",
    "order_count": 305,
    "document_receipt": null,
    "address": {
      "street_name": "raasd",
      "number": "24",
      "complement": "radsgfadf",
      "reference": "fadsgadf",
      "phone": "(24) 2-4242-4242",
      "mobile_phone": "",
      "neighborhood_name": "Centro",
      "city_name": "Maringa",
      "state_uf": "PR"
    }
  },
  "payment_method": {
    "name": "Pagamento Online",
    "subtotal": "20.15",
    "delivery_tax": "1.00",
    "total": "21.15",
    "change": 0,
    "coupon_value": 0,
    "pre_paid": true
  },
  "store": {
    "name": "Death Pizza",
    "phones": "(44) 44444-4444,(35) 33535-3535,(35) 35353-1111,(54) 5-4545-4545",
    "preparation_time": 30
  },
  "timeline": {
    "created_at": "2020-07-20 09:11:56",
    "read_at": null,
    "cancelled_at": null,
    "ready_at": null,
    "timezone": "America/Fortaleza"
  }
}
```

POST

(F4_3) Aceitar pedido (ler)

/alfredo/orders/mark-as-read

headers:

Authorization: Bearer

cmFpbXVuzG86NTY2ODgzZGMtZDMwYy00OGM0LWE1MzgtZjc1NDZIMWY1NjA4

Aiq-User-Agent: User (user@company.com)

- Pedido não lido por 15 minutos é cancelado!

request

```
{  
  "order_id": 35  
}
```

response

```
{  
  "data": {  
    "order_id": 1,  
    "read_at": "2011-02-07T01:17:50+00:00",  
    "read_by": false  
  }  
}
```

POST

(F4_4) Atualizar status pedido (pronto para pickup ou entrega)

/alfredo/orders/mark-as-ready

headers:

Authorization: Bearer

cmFpbXVuzG86NTY2ODgzZGMtZDMwYy00OGM0LWE1MzgtZjc1NDZIMWY1NjA4

Aiq-User-Agent: User (user@company.com)

request

```
{  
  "order_id": 3151  
}
```

response

```
{  
  "data": {  
    "order_id": 1,  
    "read_at": "2011-02-07T01:17:50+00:00",  
    "read_by": false  
  }  
}
```

GET

(F4_5) Listar pedidos (Busca)

/alfredo/orders/search

headers:

Authorization: Bearer

cmFpbXVuZG86NTY2ODgzZGMtZDMwYy00OGM0LWE1MzgtZjc1NDZlMwY1NjA4

Aiq-User-Agent: User (user@company.com)

request

parâmetros

```
page
string

per_page
string

filter[order_id]
string

filter[delivery_id]
string

filter[user_email]
string

filter[store_id]
string
```

response

```
{
  "data": [
    {
      "order_id": null,
      "created_at": null,
      "is_ready": false,
      "is_read": false,
      "is_cancelled": false,
      "user_name": null,
      "order_delivery_time": null,
      "order_is_pickup": false,
      "store_name": null
    },
    {
      "order_id": null,
      "created_at": null,
      "is_ready": false,
      "is_read": false,
      "is_cancelled": false,
      "user_name": null,
      "order_delivery_time": null,
      "order_is_pickup": false,
      "store_name": null
    }
  ],
  "links": {
    "first": "/?page=1",
    "last": "/?page=1",
    "prev": null,
    "next": null
  },
  "meta": {
    "current_page": 1,
    "from": 1,
    "last_page": 1,
    "path": "/",
    "per_page": "15",
    "to": 2,
    "total": 2
  }
}
```

GET

(F4_6) Listar pedidos cancelados

/alfredo/orders/cancelled

headers:

Authorization: Bearer

cmFpbXVuzG86NTY2ODgzZGMtZDMwYy00OGM0LWE1MzgtZjc1NDZIMWY1NjA4

Aiq-User-Agent: User (user@company.com)

request

response

sem parâmetros

```
{
  "data": [
    {
      "order_id": null,
      "cancelled_at": null,
      "reason": null,
      "cancelled_by": null
    },
    {
      "order_id": null,
      "cancelled_at": null,
      "reason": null,
      "cancelled_by": null
    }
  ],
  "links": {
    "first": "V?page=1",
    "last": "V?page=1",
    "prev": null,
    "next": null
  },
  "meta": {
    "current_page": 1,
    "from": 1,
    "last_page": 1,
    "links": [
      {
        "url": null,
        "label": "&laquo;
Previous",
        "active": false
      },
      {
        "url": "V?page=1",
        "label": "1",
        "active": true
      },
      {
        "url": null,
        "label": "Next
&raquo;",
        "active": false
      }
    ],
    "path": "V",
    "per_page": "15",
    "to": 2,
    "total": 2
  }
}
```