

# MIT041 Levantamento de Processos

## Api Integração Labs x Protheus





## 3.1 Dados Gerais

Nome do Cliente: Live University	Código do Cliente:
Nome do Projeto: Customização Protheus	Data: 05/05/25
Objetivo: Levantamento de Processos	Produto: Protheus
Coordenador: Caio Moscatiello	Responsáveis Cliente: Rafael Bottura

Empresa/Razão Social	Cidade/UF	CNPJ
Live University	SP	50.278.461/0001-07

## 3.2 Integrantes da reunião

Participantes
Rafael Bottura
Gesivaldo Leite

## 3.3 Levantamento de Processos

### Unidade de Negócio

#### Live

Possui ERP legado denominado Labs, o processo comercial e financeiro é feito no sistema LABS. Os pedidos de vendas são gerados por meio de equipe interna, através de negociação com cliente (Email, Portal, Telefone, Eventos, etc...). Os pagamentos são gerados na forma online (Solução MaxPago), boleto, PIX etc...

Informações controladas ERP LABs:

- Pedido de Venda
- Títulos financeiros
- Pedido de compras
- Estoque de material didático
- Cadastros dos Clientes
- Cadastro de Produtos
- Emissão de Notas
- Controle Financeiro (MaxPago – Solução Pagamento online)



### Processo Pedido de Venda Situação Atual (AS IS)

#### Pedido de Venda

- Gerado por diversas formas, contratos corporativos, contratos individualizados, prospecção passiva (Portal da empresa, e-mail, telefone etc), prospecção ativa (Eventos).
- Os valores são gerados de acordo com o projeto do curso disponibilizado.
- base o registro das cotações e uso do Portal do Fornecedor para registro de preço do fornecedor

#### Pagamentos

- Online
- Crédito em Conta / Pix
- Boleto

#### Fluxo conforme levantamento e coleta de dados

##### Inclusão PV

- Adicionado pela equipe interna em sistema legado.
- Cadastro de cliente
- Cadastro Produto
- Pedidos com modalidade assinatura, venda direta e contratos de prestação de serviço.
- Vendas com negociações com superiores a doze meses, recebem a correção do índice de inflação a cada doze meses no saldo residual das parcelas.
- Vendas com material didático, são desmembrados em dois pedidos, um de serviço ficará na matriz e o PV do material didático, é gerado em filial específica para movimentação de estoque.

### Processo Pedido de Venda Situação Desejada (TO BE)

#### Pedido de venda

Após a inclusão do Pedido de Venda no ERP Labs, será acionada a API de inclusão no ERP Protheus para cadastro dos dados nas tabelas muro SZ0(Cabeçalho PV), SZ1(Itens PV), SZ2(Clientes), Produtos (SZ3). Os dados serão gravados separadamente para uma melhor administração e manutenção.

#### Estrutura padrão SZ0

Z0\_FILIAL – Campo texto - Filial de venda

Z0\_DATA – Campo data – Data registro

Z0\_TPREG – Campo data – Tipo registro – Inclusão/Alteração/Exclusão

Z0\_TPVEN – Campo texto – Tipo de Venda – Assinatura(A), Contrato de Serviço(C) , Venda direta(V) - Produto ou Serviço.



Z0\_TIPO – Campo texto -Conteúdo 1,2,3,4,5,6,7,8,9 - Tipo de faturamento, campo com 9 possibilidades de definidas.

Tipo 1 = Venda de Curso Apes – Pagamento Cartão/Boleto

Tipo 2 = Venda de Curso Apes – Pagamento Cartão recorrente, similar assinatura de serviço

Tipo 3 = Venda de Curso Apes – Pagamento Cartão 100%

Tipo 4 = Venda de assinatura ( supply go Apes)

Tipo 5 = Venda de assinatura (partner) E SPOTS

Tipo 6 = Locação de Salas

Tipo 7 = In Company – PELA LU

Tipo 8 = Livros – E-Books

Tipo 9 = Eventos (SUPLY CHAIN)

Z0\_NUMLAB – Campo texto – Código Pedido ERP Labs

Z0\_NUMPRT – Campo texto – Código pedido/Contrato Protheus

Z0\_CNPJ1 – Campo Texto – Cnpj/Cpf cliente 1

Z0\_CNPJ2 – Campo Texto – Cnpj/Cpf cliente 2

Z0\_CONPAG1 – Campo texto - Condição de pagamento 1

Z0\_INICIO – Campo data - Data de início

Z0\_UNVIG – Campo texto – Unidade Vigência (1=Dia;2=Meses;3=Anos;4=Indeterminado).

Z0\_VIGE – Campo Numérico

Z0\_REAJ – Campo Texto (1=Sim;2=Não)

Z0\_INDICE – Campo texto, índice usado para correção contrato

Z0\_STATUS – Campo Texto – Status processamento

Z0\_MSGPROC– Campo Texto – Mensagem Processamento

### Estrutura padrão SZ1

Z1\_FILIAL – Campo texto - Filial de venda

Z1\_NUMPRT – Campo texto – Código pedido/Contrato Protheus

Z1\_PROD – Campo texto produto

Z1\_QUANT – Campo Numérico Quantidade

Z1\_VALOR – Campo Numérico Valor

Z1\_CC – Campo texto – Centro de custos

Z1\_ITCONT – Campo Texto – Item contábil

Z1\_CONTAB – Campo Texto – Conta Contábil

Z1\_CLAVLR – Campo Texto – Classe Valor

### Estrutura padrão SZ2

Z2\_FILIAL – Campo texto - Filial de venda

Z2\_DATA – Campo data – Data registro

Z2\_TPREG – Campo – Texto tipo registro – Inclusão/Alteração/Exclusão

Z2\_COD – Campo texto – Código

Z2\_LOJA– Campo texto – Loja

Z2\_CNPJ – Campo texto Quantidade

Z2\_IE – Campo texto Valor

Z2\_NOME – Campo texto – Nome Cliente

Z2\_END – Campo Texto – Endereço

Z2\_CEP – Campo Texto – CEP

Z2\_BAIRRO – Campo Texto – Bairro

Z2\_COMPLEM – Campo Texto – Complemento Endereço

Z2\_TIPO – Campo Texto – Consumidor Final/Revendedor/Rural

Z2\_EST – Campo Texto – Estado

Z2\_MUN – Campo Texto – Município

Z2\_COD\_MUN – Campo Texto – Código Município



## MIT041 – Levantamento de Processos



Z2\_DDD – Campo Texto – DDD  
Z2\_TEL – Campo Texto – Telefone  
Z2\_EMAIL – Campo Texto – Email  
Z2\_CONTAT – Campo Texto – Contato  
Z2\_NATUREZ – Campo Texto – Natureza financeira  
Z2\_PESSOA – Campo Texto – Pessoa Física/Jurídica  
Z2\_PAIS – Campo Texto – País Sefaz  
Z2\_CODPAIS – Campo Texto – País BACEN  
Z2\_STATUS – Campo Texto – Status processamento  
Z2\_MSGPROC – Campo Texto – Mensagem Processamento

### Estrutura padrão SZ3

Z3\_FILIAL – Campo texto - Filial de venda  
Z3\_DATA – Campo data – Data registro  
Z3\_TPREG – Campo data – Tipo registro – Inclusão/Alteração/Exclusão  
Z3\_COD – Campo texto – Código Produto  
Z3\_DESCR – Campo texto – Descrição Produto  
Z3\_ISS – Campo Numérico - Iss produto  
Z3\_TIPO – Campo texto – Tipo SV- Serviço / BN – Beneficiamento / EM – Embalagem  
Z3\_UNID – Campo Texto – Unidade de medida – HR – Horas / KG - Kilograma  
Z3\_GRUPO – Campo texto – Grupo de Produtos  
Z3\_NCM – Campo Texto – NCM  
Z3\_TRIBMUN – Campo Texto – Código tributação municipal  
Z3\_TS – Campo Texto – Tipo de saída – Configuração de faturamento  
Z3\_ORIGEM – Campo Texto – Origem – 0 - Nacional / 1 – Importado  
Z3\_STATUS – Campo Texto – Status processamento  
Z3\_MSGPROC – Campo Texto – Mensagem Processamento

No ambiente Protheus teremos rotinas do tipo schedule, elas serão responsáveis por manter os cadastros atualizados, incluir, consultar e atualizar pedidos de venda. Na query de leitura dos dados, primeiro serão processadas as tabelas de cadastros e na sequência as tabelas de pedido de venda. Os cadastros/PV processados com falha terão a letra “F” gravada no campo de Status, teremos um método Get na API para retornar o Status do processo e mensagem nas falhas. O processamento com sucesso terá gravado o valor “P – Processado”. Registros criados sem Status preenchidos, estão em processamento.

Qualquer alteração nos cadastros de cliente, produto ou pedido de venda no lado do ERP Labs, deverá ter o acesso na API do Protheus para geração das tabelas muro.

Qualquer manutenção em pedido de venda (alteração ou exclusão) no Labs, obrigatoriamente devem ser inseridos nas tabelas muro via API para futuro processamento, porém os pedidos/contratos não podem ter nenhum tipo de faturamento para atualização no Protheus, caso já tenha algum faturamento, o registro será revogado da atualização. Salvo o caso do cancelamento no Labs da venda, este evento subirá via API para o Protheus e terá o contrato de venda cancelado ou pedido de venda excluído por resíduo (histórico mantido).

As exclusões de produto ou cliente no Labs, subirão via API para o Protheus e terão a validação para execução similar ao pedido de vendas, o produto/cliente não pode ter movimento em nenhuma tabela do Protheus, caso tenha, a atualização é revogada.

Os pedidos criados no labs serão divididos em grupos no Protheus, teremos pedidos do tipo contrato ou pedido de venda. Teremos contratos de vendas para negociações que precisam de emissão de nota fiscal mensal, reajuste de preços etc... Para os casos de venda direta ou pagamentos no ato da negociação, seja a vista boleto ou cartão de crédito, teremos a criação de pedido de venda.



## MIT041 – Levantamento de Processos



Após a inclusão do contrato de venda ou pedido de venda no ERP Protheus via API, todos os processos no ERP serão com base nos processos padrões estabelecidos na ferramenta Protheus Totvs.

O Json com dados de cadastros recebidos na Api protheus, serão recebidos com as informações mínimas para geração de um contrato de venda, pedido de venda e nota fiscal. O Protheus Totvs será a base oficial dos cadastros, qualquer alteração deve ser replicada no ERP Labs manualmente.

Todos os processamentos da rotina de schedule, terão os resultados reportados via Email ou webhook definido pelo cliente. Qualquer situação será reportada, resultado de sucesso ou erro.

Após o faturamento, será disparado email ou webook definido pelo cliente para atualização do Status dos PVs no Labs.

### Padrão Json cadastro de Cliente:

```
{
  "companyCODE": "000001",
  "companyBRANCH": "01",
  "companyName": "João Silva ME",
  "Contribuinte": "Sim",
  "municipalInscription": "123456789",
  "EstadualInscription": "123456789",
  "document":
  [
    {
      "number": "12345678901",
      "type": "CNPJ"
    }
  ],
  "communication":
  [
    {
      "contactName": "João Silva",
      "email": "joao.silva@email.com",
      "phone": "1122334455",
    }
  ],
  "address": [
    {
      "Endereco": "Rua das Flores, 123",
      "Complemento": "Apto 101",
      "Bairro": "Centro",
      "Estado": "SAO PAULO",
      "UF": "SP",
      "CEP": "01001000",
      "Pais": "Brasil"
    }
  ],
}
```

### Padrão Json cadastro de produtos

```
{
  "produtos":
```



```
[
  {
    "Sku": "123",
    "nome": "Calça Jeans",
    "descricao": "Calça jeans de cintura alta, modelo skinny.",
    "preco": 89.99,
    "Armazen": "01",
    "Grupo": "0001",
    "IPI": 5,
    "Icms": 18,
    "ISS": 0,
    "Ncm": "09578996"
  },
  {
    "Sku": "456",
    "nome": "Calça Jeans",
    "descricao": "Calça jeans de cintura alta, modelo skinny.",
    "preco": 89.99,
    "Armazen": "01",
    "Grupo": "0001",
    "IPI": 5,
    "Icms": 18,
    "ISS": 0,
    "Ncm": "09578996"
  }
]
```

## Padrão Json pedido de venda

```
[
  {
    "Pedido": "000001",
    "PaymentTerms": "001", /* 30 DDL */
    "MessageForNote": "Mensagem para nota",
    "Nature": "0000000001", /* VENDA DE PRODUTO */
    "Cliente1":
    [
      {
        "companyCODE": "000001",
        "companyBRANCH": "01",
        "companyName": "João Silva ME",
        "Contribuinte": "Sim",
        "municipalInscription": "123456789",
        "estadualInscription": "123456789",
        "document":
        [
          {
            "number": "12345678901",
            "type": "CNPJ"
          }
        ],
        "communication":
        [
          {
            "contactName": "João Silva",
```



## MIT041 – Levantamento de Processos



```
"email": "joao.silva@email.com",  
"phone": "1122334455",  
"cellPhone": "11987654321",  
"webSite": "www.joaosilva.com"
```

```
],  
"address": [  
  {  
    "line1": "Rua das Flores",  
    "line2": "Apto 101",  
    "number": 123,  
    "district": "Centro",  
    "city": "São Paulo",  
    "state": "SP",  
    "zipCode": "01001000",  
    "country": "Brasil"  
  }  
],  
"Itens": [  
  {  
    "Sku": "123",  
    "nome": "Calça Jeans 1",  
    "descricao": "Calça jeans de cintura alta, modelo skinny.",  
    "preco": 89.99,  
    "Armazem": "01",  
    "Grupo": "0001",  
    "IPI": 5,  
    "Icms": 18,  
    "ISS": 0,  
    "Ncm": "09578996",  
    "QtdPV": 5,  
    "PrcPV": 3800,  
  },  
  {  
    "Sku": "456",  
    "nome": "Calça Jeans 2",  
    "descricao": "Calça jeans de cintura alta, modelo skinny.",  
    "preco": 89.99,  
    "Armazem": "01",  
    "Grupo": "0001",  
    "IPI": 5,  
    "Icms": 18,  
    "ISS": 0,  
    "Ncm": "09578996",  
    "QtdPV": 5,  
    "PrcPV": 3800,  
  }  
]  
},  
"Cliente2": [  
  {
```





## MIT041 – Levantamento de Processos



```
"companyCODE": "000001",
"companyBRANCH": "01",
"companyName": "João Silva ME",
  "Contribuinte": "Sim",
"municipalInscription": "123456789",
"estadualInscription": "123456789",
"document":
[
{
  "number": "12345678901",
  "type": "CNPJ"
}
],
"communication":
[
{
  "contactName": "João Silva",
  "email": "joao.silva@email.com",
  "phone": "1122334455",
  "cellPhone": "11987654321",
  "webSite": "www.joaosilva.com"
}
],
"address": [
{
  "line1": "Rua das Flores",
  "line2": "Apto 101",
  "number": 123,
  "district": "Centro",
  "city": "São Paulo",
  "state": "SP",
  "zipCode": "01001000",
  "country": "Brasil"
}
],
"Itens":
[
{
  "Sku": "123",
  "nome": "Calça Jeans 1",
  "descricao": "Calça jeans de cintura alta, modelo skinny.",
  "preco": 89.99,
  "Armazem": "01",
  "Grupo": "0001",
  "IPI": 5,
  "Icms": 18,
  "ISS": 0,
  "Ncm": "09578996",
  "QtdPV": 5,
  "PrcPV": 3800,
},
{
  "Sku": "456",
  "nome": "Calça Jeans 2",
```



```
"descricao": "Calça jeans de cintura alta, modelo skinny.",
"preco": 89.99,
"Armazem": "01",
"Grupo": "0001",
"IPI": 5,
"Icms": 18,
"ISS": 0,
"Ncm": "09578996",
"QtdPV": 5,
"PrcPV": 3800,
}
]
}
],
}
]
```

### Rotina automatizadas

As rotinas automatizadas serão acionadas por schedule nas leituras das tabelas customizadas SZ0, SZ1, SZ2 e SZ3. As rotinas serão processadas na sequência de produto, cliente e pedidos/contrato de venda.

Os executos de geração são construídos de acordo com a versão atual do Protheus, levando em consideração a configuração de tabelas e campos atuais do banco de dados, em futura atualização do ERP, é necessário validação dos processos de integração.

Cadastro de Produtos MATA010( MVC FWLoadModel("MATA010"))

O execututo MATA010 será acionado via schedule com os dados da seleção da tabela SZ3.

Os campos principais para execução da rotina são os exemplificados abaixo:

```
//-----
//  Dados do Produto
//-----

oSB1Mod:SetValue("B1_COD" , Alltrim(_cDados[Nx][1])) // Codificação Produto, conforme regra da empresa
oSB1Mod:SetValue("B1_DESC" , Alltrim(UPPER(_cDados[Nx][2]))) //Descrição Produto
oSB1Mod:SetValue("B1_UM" , "HR") //Hora
oSB1Mod:SetValue("B1_TIPO" , "SV") //Serviço
oSB1Mod:SetValue("B1_LOCPAD" , "01") //Armazém padrão
oSB1Mod:SetValue("B1_GRUPO" , "CS01") //Exemplo Curso 01
oSB1Mod:SetValue("B1_ORIGEM" , "0") //Origem Nacional
oSB1Mod:SetValue("B1_POSIPI" , cNCMSB1) // Ncm
oSB1Mod:SetValue("B1_CODISS" , cISSFED) // Codigo serviço federal
oSB1Mod:SetValue("B1_TRIBMUN" , cISSMUN) // Codigo serviço Municipal
oSB1Mod:SetValue("B1_ISS" , cISSMUN) // Codigo serviço Municipal
oSB1Mod:SetValue("B1_ALIQISS" , cISSMUN) // Aliq ISS
oSB1Mod:SetValue("B1_IPI" , cISSMUN) // Aliq IPI
oSB1Mod:SetValue("B1_PICM" , cISSMUN) // Aliq ICM
```

Cadastro de Cliente CRMA980 ( MExecAuto({[a,b,c] CRMA980(a,b,c)},aSA1Auto,3,aAI0Auto) )

O execututo CRMA980 será acionado via schedule com os dados da seleção da tabela SZ2.



O conceito de loja no ERP Protheus é para criar clientes com mesmo código para empresas com mesma raiz de Cnpj.

Exemplo : Raiz de Cnpj 61100244

No sistema teria o código de cliente "000001"

A empresa 61100244/0001-AA

Seria Código "000001" Loja "01".

A empresa 61100244/0002-BB

Seria Código "000001" loja "02".

E assim por diante, para empresas com mais 99 filiais, teríamos a combinação de letras e números a partir da filial 100, no Protheus ficaria "A0".

Os campos principais para execução da rotina são os exemplificados abaixo:

```
//-----  
// DADOS DO CLIENTE  
//-----  
aAdd(aSA1Auto,{"A1_COD" ,GETSXENUM("SA1","A1_COD") ,Nil}) // Codigo  
aAdd(aSA1Auto,{"A1_LOJA" , "01" ,Nil}) // Loja  
aAdd(aSA1Auto,{"A1_NOME" ,UPPER(oCliente[1]:GetJsonObject("name")) ,Nil}) // Nome  
aAdd(aSA1Auto,{"A1_END" ,UPPER(oCliente[1]:GetJsonObject("endereco")),Nil}) // Endereco  
aAdd(aSA1Auto,{"A1_CEP" ,UPPER(oCliente[1]:GetJsonObject("cep")),Nil}) // Endereco  
aAdd(aSA1Auto,{"A1_BAIRRO" ,UPPER(oCliente[1]:GetJsonObject("bairro")),Nil}) // Endereco  
aAdd(aSA1Auto,{"A1_COMPLEM" ,UPPER(oCliente[1]:GetJsonObject("complemento")),Nil}) // Endereco  
aAdd(aSA1Auto,{"A1_NREDUZ" ,Substr(oCliente[1]:GetJsonObject("name"),1,30),Nil}) // Nome Fantasia  
aAdd(aSA1Auto,{"A1_TIPO" , "F" ,Nil}) // Tipo  
aAdd(aSA1Auto,{"A1_EST" ,UPPER(oCliente[1]:GetJsonObject("uf")) ,Nil}) // Estado  
aAdd(aSA1Auto,{"A1_CONTATO",Upper(AllTrim(oContato[1]:GetJsonObject("firstname")) ,Nil}) // Contato  
aAdd(aSA1Auto,{"A1_DDD" ,Substr(oContato[1]:GetJsonObject("phone"),4,2) ,Nil}) // DDD  
aAdd(aSA1Auto,{"A1_TEL" ,Substr(oContato[1]:GetJsonObject("phone"),6,9) ,Nil}) // Tel  
aAdd(aSA1Auto,{"A1_EMAIL" ,AllTrim(oContato[1]:GetJsonObject("email")) ,Nil}) // Email  
aAdd(aSA1Auto,{"A1_NATUREZ",cNaturez ,Nil}) // Natureza de venda  
aAdd(aSA1Auto,{"A1_CONTA",cConta ,Nil}) // Conta Contábil  
  
If Len(AllTrim(oCliente[1]:GetJsonObject("cnpj"))) == 14  
  aAdd(aSA1Auto,{"A1_PESSOA" , "J" ,Nil}) // Pessoa  
Else  
  aAdd(aSA1Auto,{"A1_PESSOA" , "F" ,Nil}) // Pessoa  
Endif  
If !Empty(AllTrim(oCliente[1]:GetJsonObject("cnpj")))    
  aAdd(aSA1Auto,{"A1_CGC" ,AllTrim(oCliente[1]:GetJsonObject("cnpj")) ,Nil}) // cnpj  
Else  
  If Len(AllTrim(oCliente[1]:GetJsonObject("cnpj"))) == 14  
    aAdd(aSA1Auto,{"A1_CGC" ,Replace("0",14) ,Nil}) // cnpj  
  Else  
    aAdd(aSA1Auto,{"A1_CGC" ,Replace("0",11) ,Nil}) // cnpj  
  Endif  
Endif  
  
If !Empty(AllTrim(oCliente[1]:GetJsonObject("inscricao_estadual")))    
  aAdd(aSA1Auto,{"A1_INSCR" ,AllTrim(oCliente[1]:GetJsonObject("inscricao_estadual")) ,Nil}) // Inscrição  
Estadual  
  aAdd(aSA1Auto,{"A1_CONTRIB" , "1" ,Nil}) // Contribuinte(Sim/Não)  
Else  
  aAdd(aSA1Auto,{"A1_INSCR" , "ISENTO" ,Nil}) // Inscrição Estadual  
  aAdd(aSA1Auto,{"A1_CONTRIB" , "2" ,Nil}) // Contribuinte(Sim/Não)  
Endif
```



## MIT041 – Levantamento de Processos



```
aAdd(aSA1Auto,{"A1_INSCRM" ,AllTrim(oCliente[1]:GetJsonObject("codigo_municipal")) ,Nil}} // Inscrição
municipal
CC2->(dbSetOrder(4))
If CC2->(dbSeek(xFilial("CC2")+AllTrim(oCliente[1]:GetJsonObject("uf")))+
    Upper(oCliente[1]:GetJsonObject("mun"))))
    aAdd(aSA1Auto,{"A1_MUN" ,Upper(CC2->CC2_MUN) ,Nil}} // Municipio
    aAdd(aSA1Auto,{"A1_COD_MUN",Upper(CC2->CC2_CODMUN) ,Nil}} //Cod lbge
Endif
aAdd(aSA1Auto,{"A1_PAIS","105" ,Nil}} //Cod lbge
aAdd(aSA1Auto,{"A1_CODPAIS","01058" ,Nil}} //Cod bacen
```

Cadastro de Pedido/Contrato de vendas ( MsExecAuto({x, y, z| MATA410(x, y, z)},aCabec,altens,3) /MVC -> FWLoadModel('CNTA301'))

Os execautos MATA410(Pedido de vendas) / CNTA301(Contrato de vendas) são acionados via schedule com os dados da seleção das tabelas SZ0 / SZ1. Todos os processos após a criação do contrato de vendas ou pedido de vendas serão base no padrão do sistema.

A definição de qual execauto executar vem do campo ZO\_TPVEN, o conteúdo A ou C, executa a rotina de contrato de vendas (CNTA301). O conteúdo igual V, executa a rotina MATA410.

ZO\_TPVEN- Campo texto - Tipo de Venda - Assinatura(A), Contrato de Serviço(C) , Venda direta(V) - Produto ou Serviço.

A rotina CNTA301 irá gerar contrato de vendas com títulos provisórios em todos os cenários para controle de fluxo financeiro. Teremos a definição de mais de uma planilha por contrato, caso tenha os campos de Cnpj preenchidos na tabela SZ0. Com esta condição, o contrato de vendas estará associado a mais de um cliente.

ZO\_CNPJ1 - Campo Texto - Cnpj/Cpf cliente 1  
ZO\_CNPJ2 - Campo Texto - Cnpj/Cpf cliente 2

Os campos principais para execução da rotina são os exemplificados abaixo:

```
//Cabeçalho do Contrato
oModel:SetValue('CN9MASTER','CN9_NUMERO', cContrat)
oModel:SetValue('CN9MASTER','CN9_DTINIC', dDataBase)
oModel:SetValue('CN9MASTER','CN9_UNVIGE', '2')
oModel:SetValue('CN9MASTER','CN9_VIGE', 6 )
oModel:SetValue('CN9MASTER','CN9_MOEDA', 1)
oModel:SetValue('CN9MASTER','CN9_TPCTO', "031")
oModel:SetValue('CN9MASTER','CN9_CONDPG', "000")
oModel:SetValue('CN9MASTER','CN9_AUTO', '1')

//Cliente/Fornecedor do Contrato
oModel:SetValue('CNCDETAIL','CNC_CLIENT' , "CN300R")
oModel:SetValue('CNCDETAIL','CNC_LOJACL' , "01")

//Planilhas do Contrato
oModel:SetValue('CNADETAIL','CNA_NUMERO', PadL("1", Len(CNA->CNA_NUMERO),"0"))
oModel:SetValue('CNADETAIL','CNA_CLIENT', "CN300R")
oModel:SetValue('CNADETAIL','CNA_LOJACL', "01")
oModel:SetValue('CNADETAIL','CNA_TIPPLA', "ZOF")

//Itens da Planilha do Contrato
oModel:SetValue('CNBDETAIL','CNB_ITEM', PadL("1", Len(CNB->CNB_ITEM),"0"))
```

```
oModel:SetValue('CNBDETAIL','CNB_PRODUT' ,GCT00000000000000000000000001')
oModel:SetValue('CNBDETAIL','CNB_QUANT',6)
oModel:SetValue('CNBDETAIL','CNB_TS'      ,SB1->B1_TS)
```

A rotina Mata410 irá gerar pedidos de venda liberados e aptos para faturamento.

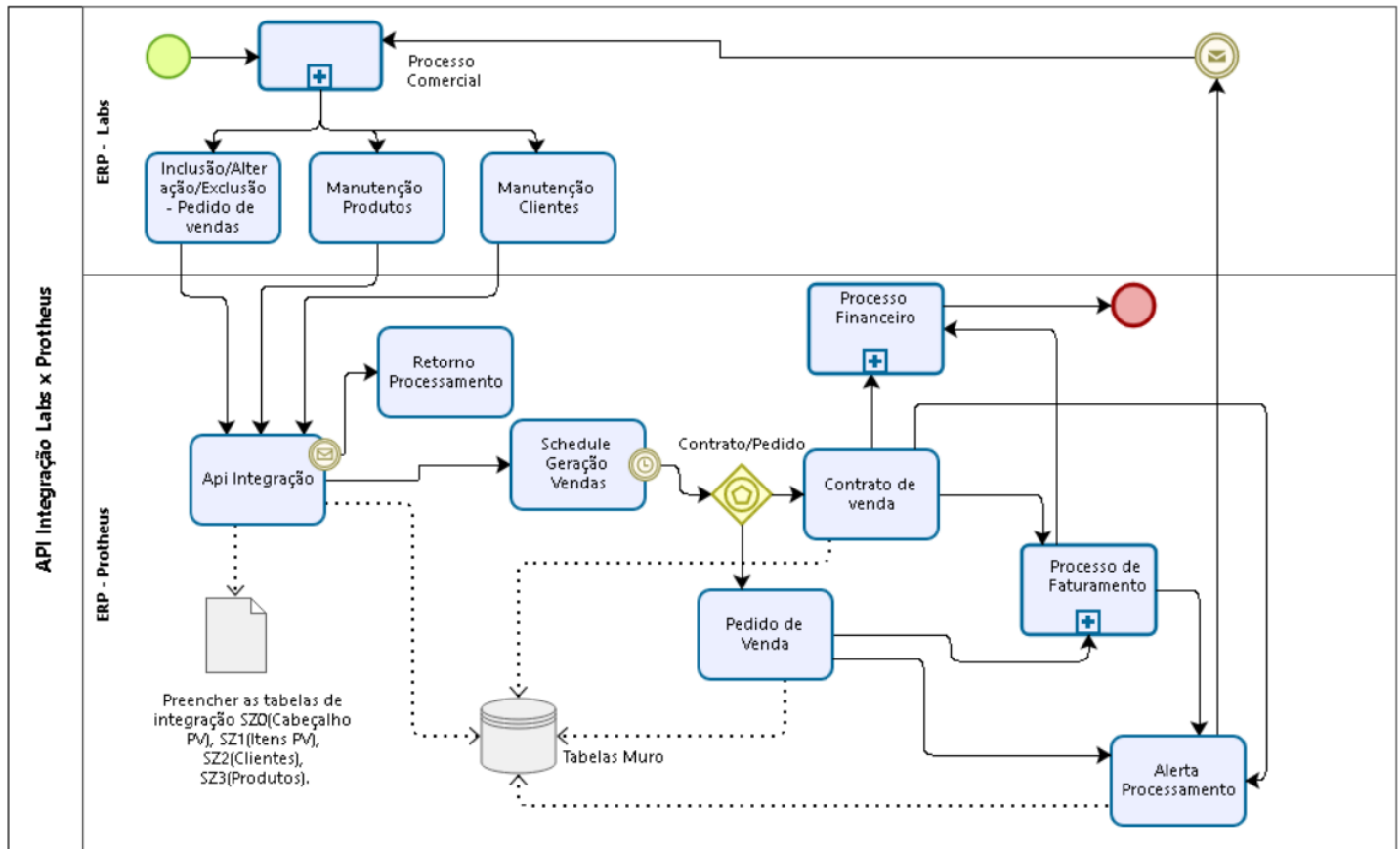
Os campos principais para execução da rotina são os exemplificados abaixo:

```
//Cabeçalho do Pedido
aAdd(aCabec,{"C5_NUM" , cPedido, NIL})
aAdd(aCabec,{"C5_TIPO" , "N", NIL})
aAdd(aCabec,{"C5_CLIENTE", SA1->A1_COD, NIL})
aAdd(aCabec,{"C5_LOJACLI", SA1->A1_LOJA, NIL})
aAdd(aCabec,{"C5_CLIENT", SA1->A1_COD, NIL})
aAdd(aCabec,{"C5_LOJAENT", SA1->A1_LOJA, NIL})
aAdd(aCabec,{"C5_CONDPAG", "001", NIL})

//Itens dp pedido
aAdd(aLinha,{"C6_ITEM", StrZero(nX,2), NIL})
aAdd(aLinha,{"C6_PRODUTO", SB1->B1_COD , NIL})
aAdd(aLinha,{"C6_QTDVEN", Val(oltems[nX]:GetJsonObject('quantity')), NIL})
aAdd(aLinha,{"C6_PRCVEN",Round(Val(oltems[nX]:GetJsonObject('amount'))/Val(oltems[nX]:GetJsonObject('quantity')),6), NIL})
aAdd(aLinha,{"C6_VALOR", Val(oltems[nX]:GetJsonObject('amount')), NIL})
aAdd(aLinha,{"C6_TES", "531", NIL})
aAdd(aLinha,{"C6_CCUSTO", cCusto NIL})
aAdd(aLinha,{"C6_CLVL", cCLVL , NIL})
aAdd(aLinha,{"C6_ITEMCTA", cltemCta, NIL})
aAdd(aLinha,{"C6_CC ", cConta, NIL})
```



### 3.4 Fluxograma





## 3.5 Premissas

- Licenciamento específico aplicação WEB
- Obrigatoriamente os cadastros de controle gerencial, devem ser replicados entre os dois ambientes (Labs e Protheus). No caso de centro de custos, contas contábeis, itens contábeis e classe de valores. Na ausência, serão aplicados valores padrão para dar seguimento ao fluxo de informações.
- No cadastro de produto, deve-se preencher o tipo de saída padrão para o faturamento correto dos produtos. É preciso definir um TES para venda de serviço e outro para venda de produto, assim como o vínculo correto em cada tipo de produto.
- Os campos principais para geração dos Speds devem ser preenchidos nos cadastros de cliente e produto.
- Configuração das series de notas fiscais do cliente no sistema e configuração do certificado das duas filiais. Inicialmente, em ambiente teste, poderemos assumir as mesmas séries de produção em ambiente Sefaz homologação.
- Parametrizar/configurar parâmetros internos do sistema. Os parâmetros envolvidos na prestação de serviço, venda de produtos, assim como na apuração dos impostos. Devem ser configurados de acordo com a necessidade da empresa.

## 3.6 Integrações do ERP

Módulos
Faturamento
Gestão de Contratos

## 3.7 Customizações

ID	Customizações
01	Api Integração ERP Labs x ERP Protheus

## 3.8 Roteiro de testes

ID	Descrição do caso de teste	Resultado esperado
01	Inclusão/ Manutenção Cadastros (Produtos, Clientes) via API	
02	Inclusão/Manutenção pedido de vendas via API	
03	Inclusão/Manutenção Contrato de vendas via API	



### 3.9 Processos que não se aplicam

ID	Descrição do processo que não se aplica
01	Faturamento
02	Baixas financeiras
03	Apuração de impostos

### 3.10 Aprovação

Aprovador por	Assinatura	Data
Rafael Botura		
Cleusa Franca		
Caio Moscatiello		
Gesi Souza		