

# CREDSYSTEM

## (PDV+TOTEEM)

- [Integração Credsystem — Introdução](#)
- [Arquitetura — Como os módulos se conectam](#)
- [Instalação e Banco de Dados](#)
- [Configuração por Loja / Entidade](#)
- [Portal do Operador — Uso no Balcão](#)
- [Fluxo de Pagamento — Dinheiro e Cartão](#)
- [Fluxo de Pagamento Instantâneo e Totem de Autoatendimento](#)
- [Parcelamento e Consulta de Limite](#)
- [Estorno e Regras de Desfazimento](#)
- [Gestor Mobile — Caixa e Impressora](#)
- [Referência Técnica para equipe de Tecnologia da Informação](#)

# Integração Credsystem — Introdução

Este material explica como o sistema ILLI se conecta à Credsystem para consultar faturas, receber pagamentos e operar o Totem de Pagamento Instantâneo nas lojas.

## Índice dos capítulos

- 01 — Introdução** (esta página): o que é o sistema e palavras importantes
- 02 — Arquitetura**: como os módulos se conectam
- 03 — Instalação**: banco de dados e passos de instalação
- 04 — Configuração**: cadastro por loja/entidade
- 05 — Portal do operador**: tela de atendimento no balcão
- 06 — Fluxo de pagamento**: pagamento em dinheiro ou cartão
- 07 — Fluxo de Pagamento Instantâneo e Totem**: pagamento por Código de Resposta Rápida e autoatendimento
- 08 — Parcelamento e limite**: simulação e contratação de planos
- 09 — Estorno e regras**: desfazimento de pagamento
- 10 — Gestor no celular**: configuração de caixa e impressora
- 11 — Referência para Tecnologia da Informação**: endpoints, tabelas e auditoria

## Em linguagem simples

O sistema permite que a loja **consulte a fatura do cliente** pelo Cadastro de Pessoa Física, **receba o pagamento** (dinheiro, cartão ou Pagamento Instantâneo) e **registre tudo** no caixa e no histórico da loja.

Existem três formas principais de uso:

1. **Portal no computador** — o operador atende o cliente no balcão.
2. **Totem de Pagamento Instantâneo** — o cliente paga sozinho na tela de autoatendimento.
3. **Aplicativo Gestor no celular** — configura qual caixa e impressora cada funcionário usa.

Tudo isso conversa com os servidores da **Credsystem** (empresa do cartão/fatura) pela internet, de forma segura.

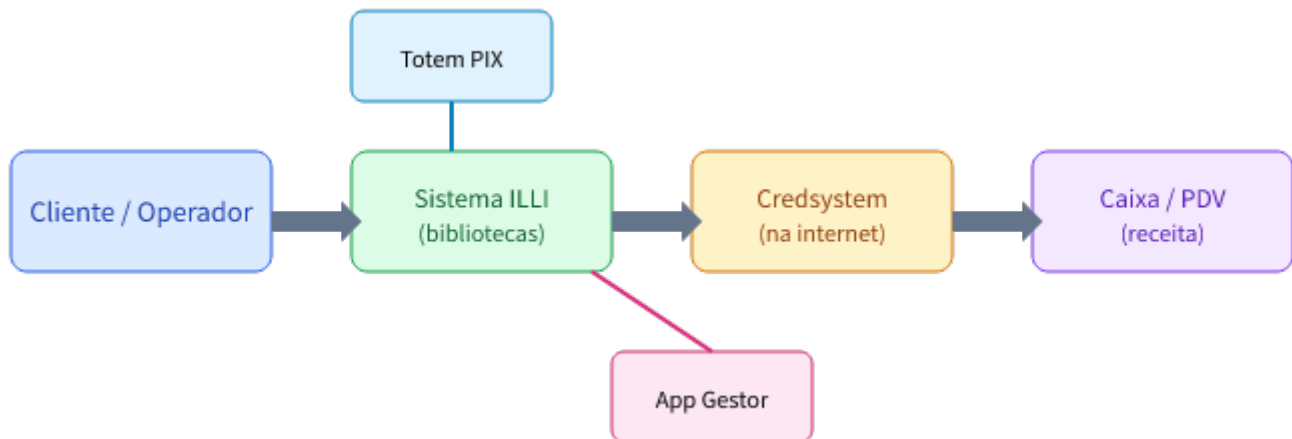


Figura 1 — O ILLI fica no meio: recebe o pedido da loja, fala com a Credsystem e registra no caixa.

# Glossário — palavras que você vai encontrar

## Cadastro de Pessoa Física

Número de identificação do cliente. É usado para buscar a fatura dele.

## Extrato / Fatura

Conta que o cliente precisa pagar. Pode ter valor total, valor mínimo ou parcelas.

## Entidade / Loja

Cada unidade de negócio cadastrada no ILLI. Cada loja tem sua própria configuração.

## Caixa

O ponto de venda onde o operador trabalha. O pagamento pode gerar uma entrada de dinheiro no caixa.

## Pré-autorização

Passo em que a Credsystem *reserva* o pagamento antes de confirmar de vez. É como "segurar" o valor.

## Efetivação

Confirmação final do pagamento na Credsystem.

## Estorno / Desfazimento

Cancelar um pagamento que já foi feito. Só é permitido em alguns casos e dentro de um prazo.

## Pagamento Instantâneo

Forma de pagamento instantânea pelo celular, usando Código de Resposta Rápida.

## Totem

Tela de autoatendimento na loja, onde o cliente paga sozinho pelo Pagamento Instantâneo.

## Integrador

Banco de dados auxiliar que guarda histórico de clientes, pagamentos e avaliações do totem.

## Ponto de Venda

Sistema de ponto de venda do ILLI. Registra entradas e saídas de dinheiro no caixa.

# Detalhes para Tecnologia da Informação

O diretório `bibliotecas/` contém 4 módulos independentes escritos em Pré-processador de Hipertexto, cada um identificado por Identificador Universal Único:

Identificador Universal Único	Classe	Função
<code>981bf94-...</code>	<code>api_integracao_credssystem</code>	Cliente de Transferência Representacional de Estado/Autorização Aberta Credssystem
<code>88e14c87-...</code>	<code>api_integracao_credssystem_pagamento_total</code>	Portal, totem, Ponto de Venda, banco integrador
<code>857e21c5-...</code>	<code>credssystem_gestor_configuracoes</code>	Config caixa/impressora por usuário
<code>f3d67c71-...</code>	<code>configCredssystem</code>	Tela legada (substituída pela Quasar em <code>981bf94</code> )

Dependência compartilhada: `util.php` ( `12262a22-30ab-11e9-bb4c-127101af6b0d` ).

Documentação focada nos módulos atuais; legado Protocolo Simples de Acesso a Objetos em `v1/credssystem` não é coberto neste livro.

# Arquitetura — Como os módulos se conectam

O sistema é dividido em peças que trabalham juntas: uma parte fala com a Credsystem, outra atende o operador, outra guarda o histórico e outra configura o celular do gestor.

## Em linguagem simples

Imagine uma loja com três pontos de contato:

- **Computador do balcão** — onde o operador consulta e recebe pagamentos.
- **Totem na entrada** — onde o cliente paga sozinho pelo Pagamento Instantâneo.
- **Celular do gestor** — onde se escolhe caixa e impressora.

Todos esses pontos passam pelo **mesmo sistema central** (ILLI), que por sua vez conversa com a Credsystem na internet e grava tudo no banco de dados da loja.

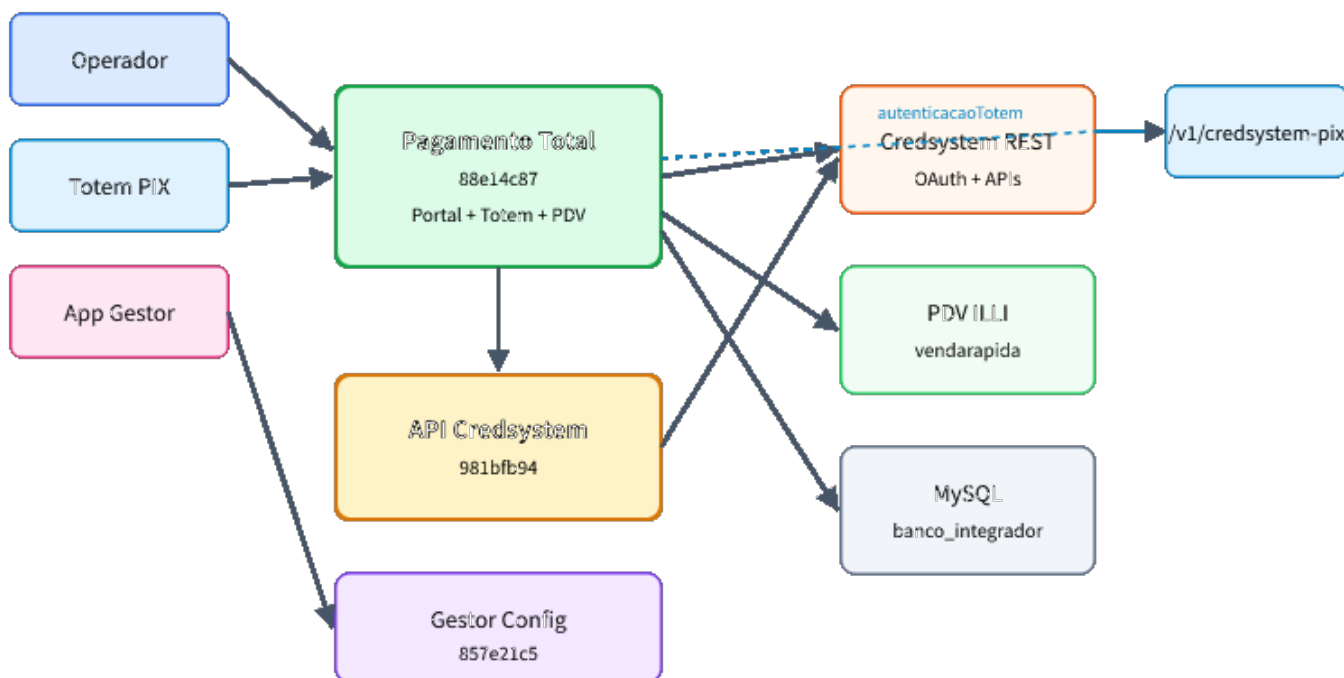


Figura 2 — Arquitetura: o módulo Pagamento Total (88e14c87) orchestra tudo; a Interface de Programação de Aplicações Credsystem (981bfb94) faz as chamadas externas.

# Detalhes para Tecnologia da Informação

## Camadas

Camada	Módulo	Responsabilidade
Apresentação	88e14c87/visual.html	Vue 3 — portal operador
Apresentação	/v1/credssystem-pix	Interface do Usuário do Totem de Pagamento Instantâneo (fora de bibliotecas/)
Apresentação	981bf94/visual.html	Quasar — configuração por entidade
Negócio	88e14c87/controle.php	Orquestração, Ponto de Venda, integrador, auditoria
Integração	981bf94/controle.php	Autorização Aberta versão 2 + Guzzle → Interface de Programação de Aplicações Credssystem
Configuração mobile	857e21c5/controle.php	configuracao_usuario

## Dependências entre módulos

- 88e14c87 instancia new api\_integracao\_credssystem() para todas as operações Credssystem.
- 88e14c87 chama vendarapida::sangriasuprimento para receita/despesa no Ponto de Venda.
- 857e21c5 e 88e14c87 usam util.php (12262a22).

## Rotas principais (@Link)

- /bibliotecas/981bf94-.../api\_integracao\_credssystem
- /bibliotecas/88e14c87-.../api\_integracao\_credssystem\_pagamento\_total
- /bibliotecas/857e21c5-.../credssystem\_gestor\_configuracoes

# Instalação e Banco de Dados

Antes de usar o sistema, é preciso instalar os módulos e criar as tabelas que guardam clientes, pagamentos e avaliações do totem.

## Em linguagem simples

A equipe de Tecnologia da Informação precisa rodar a **instalação** dos módulos no servidor. Isso cria um "caderno de anotações" digital (banco de dados) onde ficam guardados:

- Os **dados dos clientes** que já consultaram a fatura.
- O **histórico de pagamentos** feitos na loja.
- As **notas de satisfação** que o cliente dá no totem (de 1 a 5).

Sem essa instalação, o portal e o totem não conseguem salvar informações.

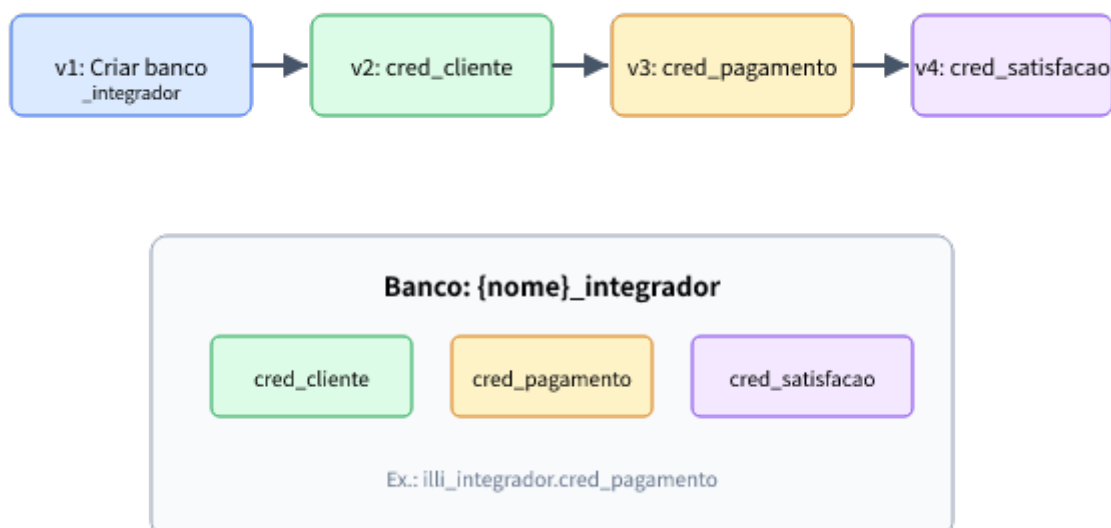


Figura 3 — Ordem das migrações do instalador (88e14c87): v1 cria o banco; v2, v3 e v4 criam as tabelas.

# Detalhes para Tecnologia da Informação

## Instalador

Arquivo: `bibliotecas/88e14c87-.../instalador.php`

Rota: `/bibliotecas/88e14c87-.../api_integracao_credsystem_pagamento_total/install`

Dependência: `util.php` ( `12262a22-30ab-11e9-bb4c-127101af6b0d` )

## Migrações

Versão	Ação
v1	<code>CREATE DATABASE IF NOT EXISTS {banco}_integrador</code>
v2	Cria tabela <code>cred_cliente</code> (Notação de Objetos JavaScript: <code>cpf</code> , <code>nome</code> , <code>codigoCliente</code> , <code>diaMesNascimento</code> )
v3	Cria tabela <code>cred_pagamento</code> ( <code>cpf</code> , entidade, usuário, caixa, valor Notação de Objetos JavaScript, tipo, situação)
v4	Cria tabela <code>cred_satisfacao</code> ( <code>cpf</code> , nota 1-5)

## Schema cred\_pagamento

Coluna	Tipo	Descrição
<code>cpf_cliente</code>	<code>varchar(14)</code>	Cadastro de Pessoa Física sem formatação
<code>id_entidade</code>	<code>int</code>	Loja/entidade
<code>id_usuario</code>	<code>int</code>	Operador
<code>id_caixa</code>	<code>int</code>	Caixa do Ponto de Venda
<code>valor</code>	Notação de Objetos JavaScript	Detalhes do pagamento
<code>pagamento</code>	<code>varchar</code>	<code>DINHEIRO</code> ou <code>PIX</code>
<code>situacao</code>	<code>varchar</code>	<code>EFETIVADO</code> ou <code>ESTORNADO</code>

# Outros módulos

- `981bfb94` — instalador vazio (sem migrações).
- `857e21c5` — usa tabela existente `configuracao_usuario`.

## Ordem recomendada de instalação

1. Instalar `981bfb94` (Interface de Programação de Aplicações Credsystem)
2. Instalar `88e14c87` (executar v1 → v4)
3. Instalar `857e21c5` (gestor)
4. Configurar entidade (capítulo 04)

# Configuração por Loja / Entidade

Cada loja precisa ter seus dados de acesso à Credsystem cadastrados no sistema, incluindo usuário, senha, Endereços Uniformes de Recurso e vínculos com o caixa.

## Em linguagem simples

Antes de consultar faturas ou receber pagamentos, a equipe de Tecnologia da Informação precisa **configurar cada loja** no sistema. É como cadastrar o "login e senha" que a loja usa para falar com a Credsystem.

O que precisa ser preenchido:

1. **Selecionar a loja** (entidade) no menu.
2. **Usuário e senha** fornecidos pela Credsystem.
3. **Endereços das Interfaces de Programação de Aplicações** (links de conexão — a equipe de Tecnologia da Informação recebe da Credsystem).
4. **Forma de pagamento e natureza** — para o caixa registrar corretamente a entrada de dinheiro.
5. **Imagem de fundo do totem** (opcional) — personaliza a tela do autoatendimento por Pagamento Instantâneo.

A tela de configuração atual usa o módulo **Interface de Programação de Aplicações Credsystem** (interface Quasar). Existe uma tela antiga que foi substituída por esta.

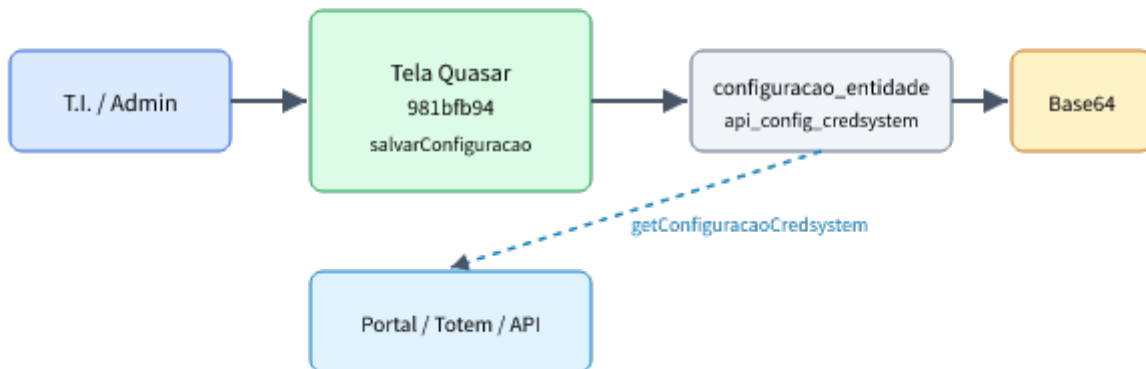


Figura 4 — A configuração é salva por entidade e lida por todos os módulos que operam pagamentos.

# Detalhes para Tecnologia da Informação

## Armazenamento

- Tabela: `configuracao_entidade`
- Chave (`descricao`): `api_config_credsystem`
- Valor: Notação de Objetos JavaScript codificado em **Codificação Base 64**
- Métodos: `getConfiguracaoCredsystem()` / `salvarConfiguracaoCredsystem()` em `981bfb94/controle.php`

## Campos do Notação de Objetos JavaScript de configuração

Campo	Descrição
<code>idEntidade</code>	ID da entidade/loja
<code>usuario</code>	Usuário Credsystem

Campo	Descrição
senha	Senha Credsystem (não expor em logs)
urlArrecadacao	Endereço Uniforme de Recurso base Interface de Programação de Aplicações arrecadação (legado Linguagem de Descrição de Serviços Web; Transferência Representacional de Estado usa Endereços Uniformes de Recurso fixos no código)
urlLimite	Endereço Uniforme de Recurso Interface de Programação de Aplicações limite/venda
bandeira	Bandeira do cartão (ex.: OBJETIVA)
idFormaPagamento	ID forma de pagamento no Ponto de Venda para receita
idNaturezaLancamento	Natureza contábil da receita
idNaturezaDespesa	Natureza contábil da despesa (estorno)

## Configuração do Totem

- Chave global: `credsystem_pix_fundo_totem_atracao` (tabela `configuracao`)
- Endpoints: `obterFundoTotemAtracao`, `salvarFundoTotemAtracao`, `limparFundoTotemAtracao`, `exibirFundoTotemAtracao`
- Armazena imagem via `arquivo_imagem_model`

## Tela de configuração

- Canônica: `981bfb94/visual.html` (Quasar/Vue)
- Legada (não usar): `f3d67c71/visual.html` (AngularJS) — mesma chave `api_config_credsystem`

## Autenticação Credsystem (Transferência Representacional de Estado)

O módulo `981bfb94` obtém token Autorização Aberta versão 2 via Oracle IDCS (`client_credentials`). Endereços Uniformes de Recurso de homologação estão definidas no construtor da classe. Para produção, parametrizar Endereços Uniformes de Recurso e credenciais Autorização Aberta fora do código ou via variáveis de ambiente.

# Portal do Operador — Uso no Balcão

O portal é a tela que o operador usa no computador para consultar a fatura do cliente, receber pagamentos e imprimir comprovantes.

## Em linguagem simples

O operador abre o portal no navegador e segue estes passos:

1. **Digitar o Cadastro de Pessoa Física** do cliente e clicar em consultar.
2. **Ver os extratos** (faturas) que aparecem na tela — valor total, mínimo, vencimento.
3. **Escolher o que pagar** — pode ser o valor total, o mínimo ou outro valor permitido.
4. **Confirmar o pagamento** — o sistema pede confirmação antes de cobrar.
5. **Imprimir o comprovante** — se a impressora estiver configurada.
6. **Desfazer** (se necessário) — só funciona em até 24 horas e não vale para Pagamento Instantâneo.

Se o cliente quiser parcelar, o operador pode consultar o limite e simular planos (ver capítulo 08).



Interface: visual.html (Vue 3) — Módulo 88e14c87

Figura 5 — Passo a passo do operador no portal de pagamento.

# Detalhes para Tecnologia da Informação

## Interface

- Arquivo: `88e14c87/visual.html` (Vue 3)
- Rota: `/bibliotecas/88e14c87-.../api_integracao_credssystem_pagamento_total`
- Chamadas JavaScript Assíncrono e XML para `.../api/{metodo}`

## Endpoints usados pelo portal

Endpoint	Função
<code>consultarTitulos</code>	Busca faturas na Credsystem; salva cliente em <code>cred_cliente</code>
<code>consultarLimiteSituacao</code>	Simula limite e planos de parcelamento
<code>contratarPlanoLimite</code>	Contrata plano na Interface de Programação de Aplicações venda-loja
<code>pagamentoTotal</code>	Pagamento convencional (dinheiro/cartão)
<code>preAutorizacaoPix</code>	Gera Código de Resposta Rápida Pagamento Instantâneo
<code>consultarStatusPixPagamento</code>	Polling status de Pagamento Instantâneo
<code>confirmarRecebimentoPix</code>	Confirma Pagamento Instantâneo concluído
<code>listarPagamentosCredsystemIntegrador</code>	Lista pagamentos estornáveis
<code>desfazerPagamento</code>	Estorno na Credsystem + despesa no Ponto de Venda
<code>registrarControleImpressao</code>	Auditoria — impressão correta
<code>registrarControleImpressaoIncorreta</code>	Auditoria — impressão com erro

## Sessão do Ponto de Venda

Para lançar receita/despesa no caixa, o payload deve incluir dados de sessão: `token`, `id_conta`, `id_operador`. Sem isso, o pagamento na Credsystem pode ocorrer mas o lançamento no Ponto de Venda falha.

# Payload típico — consultarTitulos

```
{  
  "codigoEstabelecimento": "...",  
  "cpfCliente": "12345678901"  
}
```

# Fluxo de Pagamento — Dinheiro e Cartão

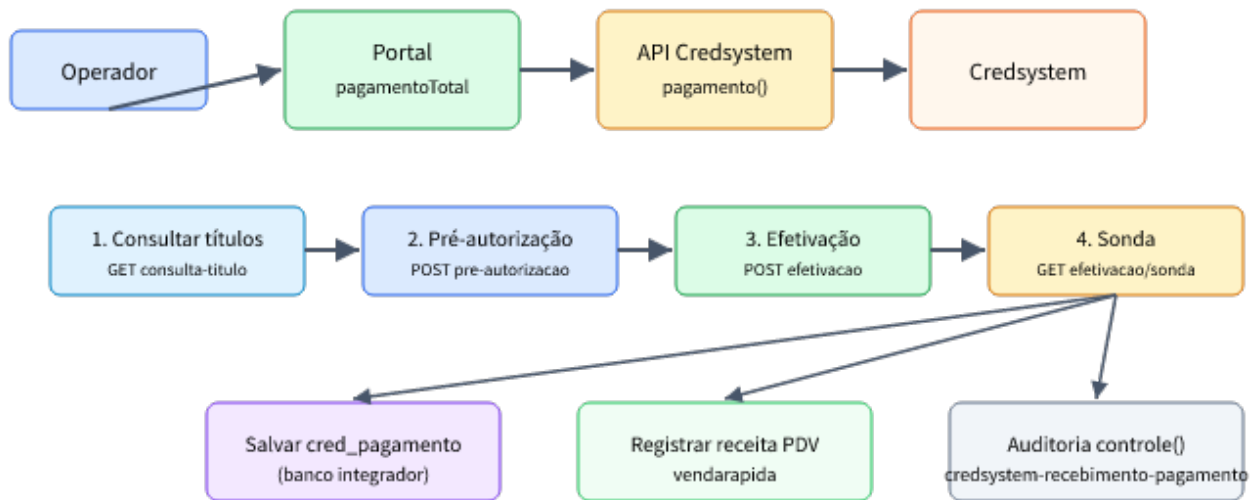
Quando o cliente paga a fatura em dinheiro ou cartão no balcão, o sistema passa por quatro etapas na Credsystem antes de registrar o valor no caixa.

## Em linguagem simples

O pagamento no balcão funciona assim:

1. **Buscar a fatura** — o sistema consulta na Credsystem o que o cliente deve.
2. **Reservar o valor** (pré-autorização) — a Credsystem "segura" o valor antes de confirmar.
3. **Confirmar o pagamento** (efetivação) — a Credsystem registra que o dinheiro foi recebido.
4. **Verificar se deu certo** (sonda) — o sistema confere se a Credsystem confirmou tudo.
5. **Registrar no caixa** — o valor entra como receita no Ponto de Venda da loja.
6. **Guardar no histórico** — fica salvo no banco integrador para consultas futuras.

Se algo der errado no meio do caminho, o sistema pode tentar **estornar automaticamente** a reserva (quando configurado).



Pagamento tipo DINHEIRO — gera receita no caixa

Figura 6 — Fluxo completo do pagamento convencional: quatro etapas na Credsystem, depois gravação local e lançamento no Ponto de Venda.

# Detalhes para Tecnologia da Informação

## Orquestração

- A interface front-end chama `pagamentoTotal()` em `88e14c87/controlle.php`
- Que delega para `api_integracao_credsystem::pagamento()` em `981bfb94/controlle.php`

## Etapas na Interface de Programação de Aplicações Credsystem (método pagamento)

#	Etapa	Endpoint de Transferência Representacional de Estado	Campo etapa em erro
1	Consultar títulos	GET consulta-titulo	consultarTitulos
2	Pré-autorização	POST pre-autorizacao	preAutorizar
3	Efetivação	POST efetivacao	efetivar
4	Sonda efetivação	GET efetivacao/sonda	sondaEfetivacao

## Pós-sucesso em 88e14c87

1. `salvarCredsystemPagamento()` → INSERT em `cred_pagamento` (tipo `DINHEIRO`, situação `EFETIVADO`)
2. `registrarControleAcaoPagamentoCredsystem()` → log tipo `credsystem-recebimento-pagamento`
3. `registraPagamento()` → `registrarReceita()` → `vendarapida::sangriasuprimento` (RECEITA)

## Parâmetros relevantes

- `valorPagamento` — valor efetivo (mínimo, parcela ou total)
- `estornarEmFalha` — default `true`; chama `desfazer` se efetivação falhar após pré-autorização
- `idFormaPagamento` e `idNaturezaLancamento` — da configuração da entidade

# Fluxo de Pagamento Instantâneo e Totem de Autoatendimento

O cliente pode pagar a fatura pelo Pagamento Instantâneo usando o Código de Resposta Rápida — seja no balcão com o operador ou sozinho no totem de autoatendimento.

## Em linguagem simples

### No balcão (com operador)

1. O operador consulta o Cadastro de Pessoa Física e escolhe pagar com **Pagamento Instantâneo**.
2. Aparece um **Código de Resposta Rápida** na tela.
3. O cliente escaneia com o aplicativo do banco e paga.
4. O sistema fica **verificando** se o pagamento chegou (a cada poucos segundos).
5. Quando confirmado, aparece a mensagem de sucesso.

### No totem (autoatendimento)

1. O cliente toca na tela do totem na loja.
2. O totem abre a página de pagamento instantâneo (`/v1/credsystem-pix`).
3. O cliente digita o Cadastro de Pessoa Física, vê o Código de Resposta Rápida e paga pelo celular.
4. No final, pode dar uma **nota de 1 a 5** sobre o atendimento.

**Importante:** pagamento por Pagamento Instantâneo *não pode ser desfeito* pelo operador depois (diferente do dinheiro/cartão).

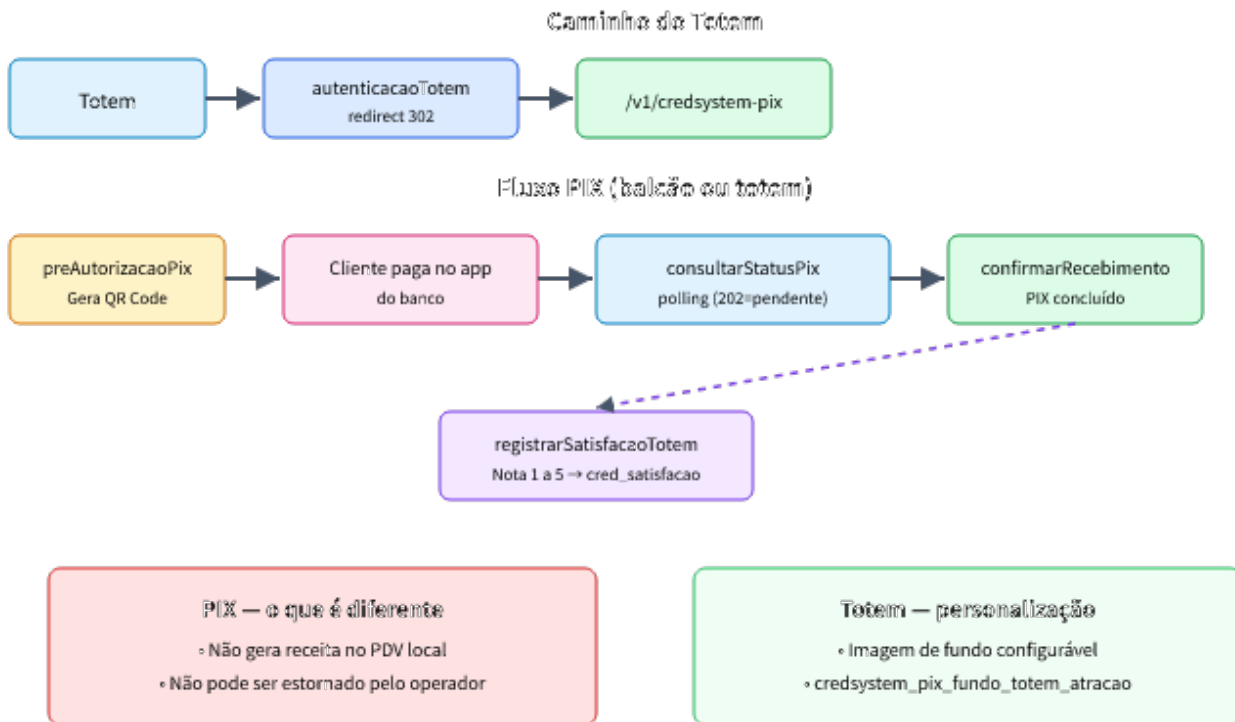


Figura 7 — Totem redireciona para `/v1/credsystem-pix`; fluxo Pagamento Instantâneo usa Código de Resposta Rápida, verificação periódica de status e avaliação opcional.

# Detalhes para Tecnologia da Informação

## Totem — entrada

- `autenticacaoTotem()` em `88e14c87/controle.php`
- Protocolo de Transferência de Hipertexto código 302 → `/v1/credsystem-pix` (Interface do Usuário fora de `bibliotecas/`)

## Endpoints de Pagamento Instantâneo

Método 88e14c87	Interface de Programação de Aplicações 981bfb94	Endpoint Credsystem
<code>preAutorizacaoPix</code>	<code>preAutorizacaoPix()</code>	POST <code>pre-autorizacao/pix</code>
<code>consultarStatusPixPagamento</code>	<code>consultarStatusPreAutorizacaoPix()</code>	GET <code>pre-autorizacao/sonda/status-pix</code>

Método 88e14c87	Interface de Programação de Aplicações 981bfb94	Endpoint Credsystem
confirmarRecebimentoPix	—	Salva em cred_pagamento (tipo Pagamento Instantâneo); sem registrarReceita

## Consulta repetida de Pagamento Instantâneo

- Protocolo de Transferência de Hipertexto **202** = pagamento ainda pendente; front deve continuar consultando.
- Protocolo de Transferência de Hipertexto **200** com sucesso = Pagamento Instantâneo confirmado.
- Auditoria: credsystem-pix-pre-autorizacao, credsystem-recebimento-pix

## Satisfação do totem

- registrarSatisfacaoTotem() → INSERT em cred\_satisfacao
- Campos: cpf\_cliente, nota (1-5)

## Imagem de fundo

- Config global: credsystem\_pix\_fundo\_totem\_atracao
- Criar, Ler, Atualizar e Excluir via obter/salvar/limpar/exibirFundoTotemAtracao

# Parcelamento e Consulta de Limite

Além de pagar a fatura, o sistema permite consultar o limite de crédito do cliente e contratar um plano de parcelamento na Credsystem.

## Em linguagem simples

Às vezes o cliente não quer pagar tudo de uma vez. O sistema permite:

1. **Consultar o limite** — ver quanto o cliente pode parcelar no cartão.
2. **Simular planos** — ver opções de parcelas (quantas vezes, valor de cada parcela).
3. **Contratar o plano** — o cliente escolhe e o sistema confirma na Credsystem.

O operador faz isso pelo mesmo portal do balcão, depois de consultar o Cadastro de Pessoa Física do cliente.

É como ir à loja e perguntar: "Posso parcelar essa compra em 3 vezes?" — o sistema responde com as opções disponíveis.

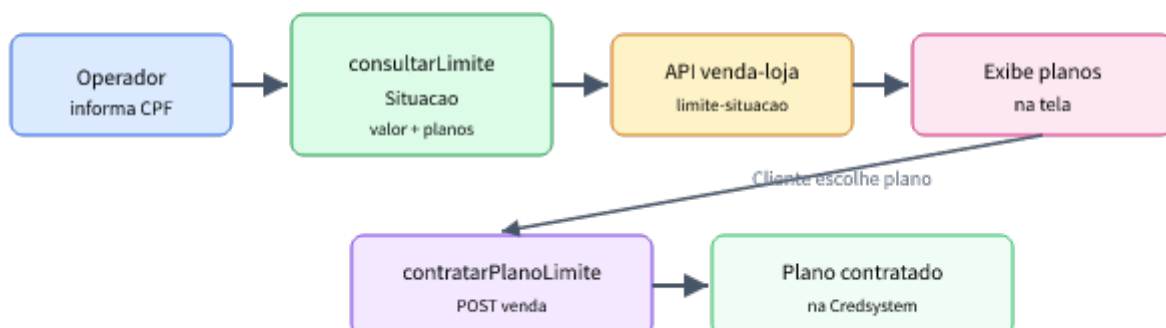


Figura 8 — Consulta de limite retorna planos; contratação confirma o parcelamento na Interface de Programação de Aplicações venda-loja.

# Detalhes para Tecnologia da Informação

## Endpoints

Método 88e14c87	Método 981bf94	Interface de Programação de Aplicações Credsystem
<code>consultarLimiteSituacao</code>	<code>consultarLimiteSituacao()</code>	GET venda-loja/.../limite-situacao (header <code>cpf</code> )
<code>contratarPlanoLimite</code>	<code>contratarPlanoVenda()</code>	POST venda-loja/.../venda

## Payload — consultarLimiteSituacao

```
{
  "codigoEstabelecimento": "...",
  "cpfCliente": "12345678901",
  "valorCompra": 150.00,
  "isInfoPlanos": true
}
```

## Payload — contratarPlanoLimite (obrigatórios)

- `codigoEstabelecimento`, `cpfCliente`
- `codigoPlano`, `parcelas` ( $\geq 1$ )
- `valorCompra`, `valorParcela`
- Opcionais: `numeroCartao`, `tipoParcelamento`, `isComJuros`

## Endereço Uniforme de Recurso base

Interface de Programação de Aplicações de venda-loja: `https://apihtml.credsystem.com.br/venda-loja/api/v1/` (homologação, definida em `981bf94`)

# Estorno e Regras de Desfazimento

Em alguns casos, o operador pode cancelar um pagamento já feito — mas existem regras de prazo e tipo de pagamento que precisam ser respeitadas.

## Em linguagem simples

Se o operador cometeu um erro ou o cliente desistiu, é possível **desfazer o pagamento** — mas com estas regras:

Regra	Explicação
<b>Prazo de 24 horas</b>	Só dá para desfazer pagamentos feitos nas últimas 24 horas.
<b>Pagamento Instantâneo não pode</b>	Pagamentos feitos por Pagamento Instantâneo não aparecem na lista de desfazimento.
<b>Só pagamentos confirmados</b>	Apenas pagamentos com situação "efetivado" (confirmado).
<b>Registra saída no caixa</b>	O estorno gera uma despesa no Ponto de Venda (dinheiro "saí" do caixa).

Passos para o operador:

1. Consultar o Cadastro de Pessoa Física do cliente.
2. Abrir a lista de pagamentos que podem ser desfeitos.
3. Escolher o pagamento e confirmar o desfazimento.
4. O sistema cancela na Credsystem e registra a despesa no caixa.

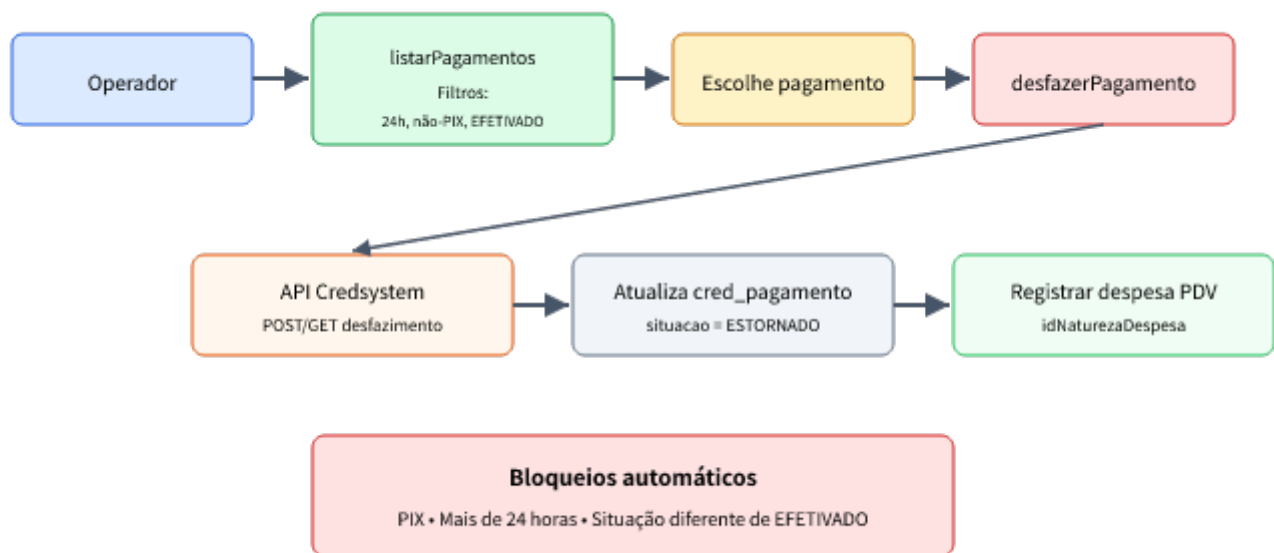


Figura 9 — Estorno: lista filtrada → desfazimento na Credsystem → atualização local → despesa no Ponto de Venda.

# Detalhes para Tecnologia da Informação

## Constante de prazo

`PRAZO_DESFAZIMENTO_HORAS = 24` em `88e14c87/controle.php`

## Query de listagem

### (listarPagamentosCredsystemIntegrador)

- `situacao = 'EFETIVADO'`
- `pagamento <> 'PIX'`
- `data_criacao >= NOW() - 24 HOURS`
- Limite: 2000 registros por Cadastro de Pessoa Física

## desfazerPagamento()

1. Valida prazo e existência do registro em `cred_pagamento`
2. Chama `api_integracao_credssystem::desfazer()` → `desfazimento`
3. Atualiza `cred_pagamento.situacao` para `ESTORNADO`
4. `registrarDespesa()` → `vendarapida::sangriasuprimimento` (DESPESA, usa `idNaturezaDespesa`)
5. Auditoria: `credssystem-estorno-pagamento`

## Estorno automático em falha

No fluxo `pagamento()` (981bfb94), se `estornarEmFalha=true` (default) e a efetivação falhar após pré-autorização, o sistema chama `desfazer` automaticamente.

# Gestor Mobile — Caixa e Impressora

O aplicativo gestor no celular permite que cada funcionário configure qual caixa e qual impressora vai usar no dia a dia.

## Em linguagem simples

Cada funcionário da loja pode ter um **caixa diferente** e uma **impressora diferente**. O aplicativo gestor no celular guarda essas preferências para não precisar configurar toda vez.

O que o gestor configura:

- **Caixa** — qual ponto de venda o funcionário vai usar.
- **Impressora** — qual impressora vai imprimir os comprovantes (pode ser Bluetooth, USB, rede, ou nenhuma).

Quando o funcionário abre o aplicativo, o sistema **carrega automaticamente** as configurações salvas da última vez.

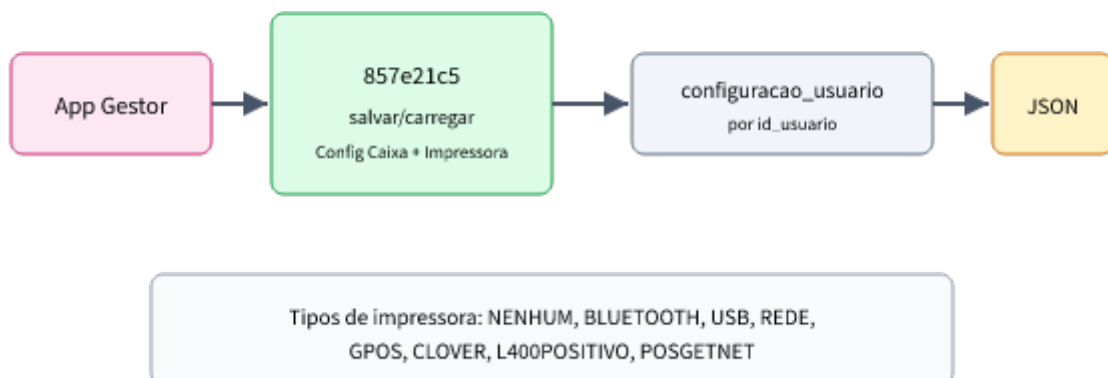


Figura 10 — O aplicativo gestor salva e carrega configurações por usuário na tabela `configuracao_usuario`.

# Detalhes para Tecnologia da Informação

## Módulo

- Classe: `credssystem_gestor_configuracoes`
- Identificador Universal Único: `857e21c5-203f-4943-b60d-9258bf0ebd7e`
- Rota: `/bibliotecas/857e21c5-.../credssystem_gestor_configuracoes`

## Endpoints

Método	Descrição
<code>salvarConfiguracaoCaixa</code>	Persiste configuração de caixa
<code>carregarConfiguracaoCaixa</code>	Carrega caixa + mescla impressora
<code>salvarConfiguracaoImpressora</code>	Persiste configuração de impressora
<code>carregarConfiguracaoImpressora</code>	Carrega só impressora

## Armazenamento

descricao	Conteúdo
<code>gestor_configuracao_caixa</code>	Notação de Objetos JavaScript legível (não Codificação Base 64)
<code>gestor_configuracao_impresora</code>	Notação de Objetos JavaScript com tipo e parâmetros da impressora

Chave: `id_usuario` (resolvido via formulário/dispositivo). Usa `ON DUPLICATE KEY UPDATE`.

Cache de sessão: `setGlobalSession("USUARIO-{id}", descricao, base64(json))` quando disponível.

## Tipos de impressora suportados

`NENHUM`, `BLUETOOTH`, `USB`, `REDE`, `GPOS`, `CLOVER`, `L400POSITIVO`, `POSGETNET`

# Referência Técnica para equipe de Tecnologia da Informação

Tabelas de endpoints, rotas, auditoria, integrações externas e dependências — material de consulta rápida para equipes de tecnologia.

## Em linguagem simples

Esta página é um **dicionário técnico** para a equipe de Tecnologia da Informação. Ela lista todas as "portas" do sistema (endpoints), onde os dados ficam guardados e como auditar o que aconteceu.

Se você é operador ou gestor, os capítulos anteriores já têm o que precisa. Esta página é para quem instala, configura e dá suporte ao sistema.

## Mapa de módulos e rotas

Identificador Universal Único	Classe	Rota base
981bfb94-afe7-4956-8ecd-4ecbc3b45246	api_integracao_credssystem	/bibliotecas/981bfb94- .../api_integracao_credssystem
88e14c87-331c-485b-95f8-8f63d76d446d	api_integracao_credssystem_pagamento_to tal	/bibliotecas/88e14c87- .../api_integracao_credssystem_pagamento_ total
857e21c5-203f-4943-b60d-9258bf0ebd7e	credssystem_gestor_configuracoes	/bibliotecas/857e21c5- .../credssystem_gestor_configuracoes

## Integrações externas

Sistema	Protocolo	Endereço Uniforme de Recurso (homologação)
Autorização Aberta Oracle IDCS	POST client_credentials	<code>https://idcs-...identity.oraclecloud.com/oauth2/v1/token</code>
Credsystem Arrecadação	Transferência Representacional de Estado versão 2	<code>https://apihtml.credsystem.com.br/arrecadacao-lojista/api/v2/</code>
Credsystem Venda-loja	Transferência Representacional de Estado versão 1	<code>https://apihtml.credsystem.com.br/venda-loja/api/v1/</code>
ILLI Ponto de Venda	Pré-processador de Hipertexto interno	<code>vendarapida::sangriasuprimento</code>
Totem de Pagamento Instantâneo Interface do Usuário	redirecionamento do Protocolo de Transferência de Hipertexto	<code>/v1/credsystem-pix</code>

Autorização Aberta `client_id` e `client_secret` estão no código de 981bfb94 — parametrizar para produção; não expor em documentação.

# Endpoints — Interface de Programação de Aplicações Credsystem (981bfb94)

Método Pré-processador de Hipertexto	Protocolo de Transferência de Hipertexto	Endpoint Credsystem
<code>consultarTitulos</code>	GET	<code>consulta-titulo</code>
<code>preAutorizar</code>	POST	<code>pre-autorizacao</code>
<code>preAutorizarPix</code>	POST	<code>pre-autorizacao/pix</code>
<code>consultarStatusPreAutorizacao</code>	GET	<code>pre-autorizacao/sonda/status</code>
<code>consultarStatusPreAutorizacaoPix</code>	GET	<code>pre-autorizacao/sonda/status-pix</code>
<code>efetivar</code>	POST	<code>efetivacao</code>
<code>consultarSondaEfetivacao</code>	GET	<code>efetivacao/sonda</code>
<code>desfazer</code>	POST/GET	<code>desfazimento</code>
<code>consultarFaturas</code>	GET	<code>faturas</code>
<code>consultarExtrato</code>	GET	<code>faturas/extrato/{cod}/vencimento/{data}</code>
<code>consultarTransacoes</code>	GET	<code>faturas/extrato/.../transacoes</code>

Método Pré-processador de Hipertexto	Protocolo de Transferência de Hipertexto	Endpoint Credsystem
consultarLimiteSituacao	GET	limite-situacao (cabeçalho cpf)
contratarPlanoVenda	POST	venda
pagamento	—	Orquestra: títulos → pré-autorização → efetivação → sonda
pagamentoPix	—	Orquestra fluxo Pagamento Instantâneo com verificação periódica de status
getConfiguracaoCredsystem	—	Lê configuracao_entidade
salvarConfiguracaoCredsystem	—	Grava api_config_credsystem

## Endpoints — Pagamento Total (88e14c87)

Método Pré-processador de Hipertexto	Efeito colateral
consultarTitulos	Interface de Programação de Aplicações + INSERT cred_cliente
consultarLimiteSituacao	Delega 981bfb94
contratarPlanoLimite	Delega 981bfb94
pagamentoTotal	Interface de Programação de Aplicações + cred_pagamento + receita no Ponto de Venda + auditoria
preAutorizacaoPix	Interface de Programação de Aplicações + auditoria de Pagamento Instantâneo
consultarStatusPixPagamento	Delega 981bfb94
confirmarRecebimentoPix	cred_pagamento (Pagamento Instantâneo) — sem receita no Ponto de Venda
desfazerPagamento	Interface de Programação de Aplicações desfazer + ESTORNADO + despesa no Ponto de Venda
listarPagamentosCredsystemIntegrador	SELECT cred_pagamento (filtros 24h, não-Pagamento Instantâneo)
registrarControleImpressao	Auditoria impressão correta
registrarControleImpressaoIncorreta	Auditoria impressão incorreta
registrarSatisfacaoTotem	INSERT cred_satisfacao

Método Pré-processador de Hipertexto	Efeito colateral
autenticacaoTotem	Protocolo de Transferência de Hipertexto código 302 → /v1/credssystem-pix
obter/salvar/limpar/exibirFundoTotemAtracao	Criar, Ler, Atualizar e Excluir imagem totem

## Banco de dados

Tabela / Chave	Módulo	Uso
configuracao_entidade.api_config_credssystem	981bfb94	Configuração por loja (Codificação Base 64)
configuracao_usuario.gestor_configuracao_caixa	857e21c5	Caixa por usuário
configuracao_usuario.gestor_configuracao_impresora	857e21c5	Impressora por usuário
configuracao_credssystem_pix_fundo_totem_atracao	88e14c87	Imagem fundo totem
{banco}_integrador.cred_cliente	88e14c87	Dados cliente (Notação de Objetos JavaScript)
{banco}_integrador.cred_pagamento	88e14c87	Histórico pagamentos
{banco}_integrador.cred_satisfacao	88e14c87	índice de satisfação do totem (1-5)

## Auditoria (controle)

Método `controle()` em 88e14c87 registra eventos com campo `tipo`:

tipo	Quando
credssystem-recebimento-pagamento	Pagamento convencional efetivado
credssystem-pix-pre-autorizacao	Código de Resposta Rápida Pagamento Instantâneo gerado
credssystem-recebimento-pix	Pagamento Instantâneo confirmado
credssystem-estorno-pagamento	Desfazimento realizado
credssystem-comprovante-impressao	Comprovante impresso corretamente
credssystem-comprovante-impressao-incorreta	Erro na impressão

# Regras de negócio (constantes)

Regra	Valor	Arquivo
Prazo estorno	24 horas	<code>88e14c87/controlle.php</code> — <code>PRAZO_DESFAZIMENTO_HORAS</code>
Pagamento Instantâneo estornável	Não	Filtro Linguagem de Consulta Estruturada <code>pagamento &lt;&gt; 'PIX'</code>
Pagamento Instantâneo gera receita no Ponto de Venda	Não	<code>confirmarRecebimentoPix</code> sem <code>registrarReceita</code>
Estorno automático em falha	Default true	<code>estornarEmFalha</code> em <code>pagamento()</code>
Consulta repetida de Pagamento Instantâneo pendente	Protocolo de Transferência de Hipertexto código 202	<code>consultarStatusPreAutorizacaoPix</code>

## Dependências compartilhadas

- `APP_PATH/controllers/bibliotecas/12262a22-30ab-11e9-bb4c-127101af6b0d/util.php` — utilitários de banco de dados, entidades e consultas
- `APP_PATH/controllers/pdv/vendarapida.php` — sangria/suprimento (receita/despesa)
- `APP_PATH/controllers/biblioteca.php` — classe base biblioteca
- Guzzle para Protocolo de Transferência de Hipertexto — usado em `981bfb94` para chamadas de Transferência Representacional de Estado

## Arquivos-fonte principais

Arquivo	Linhas aprox.	Conteúdo
<code>bibliotecas/981bfb94-.../controlle.php</code>	~1700	Cliente de Transferência Representacional de Estado Credsistema completo
<code>bibliotecas/88e14c87-.../controlle.php</code>	~1710	Orquestração pagamento, Ponto de Venda, totem
<code>bibliotecas/88e14c87-.../visual.html</code>	~2000	Portal Vue 3
<code>bibliotecas/88e14c87-.../instalador.php</code>	~110	Migrações v1-v4
<code>bibliotecas/981bfb94-.../visual.html</code>	—	Config Quasar
<code>bibliotecas/857e21c5-.../controlle.php</code>	~460	Configuração do gestor



Figura 11 — Mapa resumido de módulos e dependências externas ao diretório bibliotecas/.