

# RF001 - Login de Sistema

## 1. Identificação do Requisito

**Código:** RF001

**Nome:** Login de Sistema

**Módulo:** Autenticação

**Prioridade:** Alta

**Complexidade:** Média

**Arquivo Principal:** 02\_formularios\00\_login\FrmLogin.cs

## 2. Descrição Geral

O sistema de login é responsável por autenticar usuários no Add-in do SolidWorks, validando credenciais, gerenciando sessões e configurando permissões de acesso. O formulário implementa autenticação automática para usuários previamente alocados e login manual com validação de credenciais criptografadas.

## 3. Interface do Usuário

### 3.1 Elementos da Interface

Elemento	Tipo	Propriedades	Descrição
txtUsuario	LmTextBox	CampoObrigatorio=true, WaterMark="Usuário"	Campo para inserção do nome de usuário
txtSenha	LmTextBox	CampoObrigatorio=true, WaterMark="Senha", UseSystemPasswordChar=true	Campo para inserção da senha (mascarado)
btnLogin	LmButton	Text="Entrar"	Botão para executar o login
ptb	PictureBox	BackgroundImageLayout=Zoom	Exibe logo/imagem de carregamento
lblCarregando	LmLabel	Text="Carregando..."	Label de status durante inicialização

### 3.2 Layout e Dimensões

- **Tamanho do Formulário:** 281x361 pixels
- **Posicionamento:** Centralizado (Anchor = None)
- **Campos de entrada:** 220x30 pixels cada
- **Botão Login:** 220x30 pixels
- **Estilo:** Interface moderna com bordas arredondadas

### 3.3 Comportamento Visual

- **Inicialização:** Exibe indicador de carregamento
- **Auto-preenchimento:** Campos são preenchidos automaticamente se usuário já estiver alocado
- **Validação:** Campos obrigatórios são destacados se inválidos
- **Feedback:** Mensagens toast para sucesso/erro

## 4. Regras de Negócio (RN)

### RN001 - Validação de Conectividade

**Descrição:** O sistema deve verificar conectividade com internet antes de tentar autenticação

**Implementação:** `Web.IsConnected()`

**Ação:** Se sem internet, exibe aviso "Sem Internet"

### RN002 - Auto-Login por Máquina

**Descrição:** Usuários previamente autenticados na máquina devem ter login automático

**Implementação:** Consulta tabela `usuario_alocados` por `hostname` e `usuario_pc`

**Critério:** `hostname == Dns.GetHostName() && usuario_pc == Environment.UserName`

### RN003 - Criptografia de Senhas

**Descrição:** Senhas devem ser armazenadas e validadas com criptografia AES

**Implementação:** `Criptografar.EncryptAES()` e `DescriptografarAES()`

**Segurança:** Senhas nunca são armazenadas em texto plano

## RN004 - Validação de Usuário Ativo

**Descrição:** Apenas usuários com status ativo podem fazer login

**Implementação:** Verificação do campo `usuarios.ativo`

**Ação:** Se inativo, exibe "Usuário Inativo!"

## RN005 - Alocação de Usuário

**Descrição:** Após login bem-sucedido, usuário deve ser alocado à máquina

**Implementação:** Criação/atualização de registro em `usuario_alocados`

**Dados:** `usuario_pc, hostname, usuario_id`

## RN006 - Configuração de Ambiente

**Descrição:** Login bem-sucedido deve configurar variáveis de ambiente do sistema

**Implementação:**

- Nome do Sistema: "Axion Artama"
- Pasta Raiz: "LM Projetos Data"
- Cliente: "Artama"
- Email: "michalakleo@gmail.com"

## RN007 - Carregamento de Configurações

**Descrição:** Após autenticação, carregar configurações globais do sistema

**Implementação:**

- Configurações de API (`configuracao_api`)
- Processos (`Processo.Carregar()`)
- Processos não serializados (`ProcessoNaoSerializado.Carregar()`)
- Templates (`templates.Carregar()`)
- Configurações gerais (`InfoSetting.Carregar()`)

# 5. Fluxos de Processo

## 5.1 Fluxo Principal (FP001) - Login Manual

1. **Usuário acessa o sistema**
2. **Sistema verifica conectividade**
  - Se sem internet → FA001
3. **Sistema verifica auto-login**
  - Se usuário alocado → FP002
4. **Usuário preenche credenciais**
5. **Usuário clica em "Entrar"**
6. **Sistema valida campos obrigatórios**
  - Se inválidos → FA002
7. **Sistema busca usuário no banco**
  - Se não encontrado → FA003
8. **Sistema valida senha criptografada**
  - Se inválida → FA004
9. **Sistema verifica se usuário está ativo**
  - Se inativo → FA005
10. **Sistema aloca usuário à máquina**
11. **Sistema configura permissões**
12. **Sistema carrega configurações globais**
13. **Sistema oculta tela de login**
14. **Sistema exibe painel principal**

## 5.2 Fluxo Principal (FP002) - Auto-Login

1. Sistema verifica conectividade
2. Sistema consulta usuários alocados
3. Sistema encontra usuário para hostname/usuario\_pc
4. Sistema preenche campos automaticamente
5. Sistema executa login automaticamente
6. Continua no passo 10 do FP001

## 6. Fluxos Alternativos (FA)

### FA001 - Sem Conectividade

**Condição:** `!Web.IsConnected()`

**Ação:** Exibe toast "Sem Internet" e oculta indicador de carregamento

**Retorno:** Usuário deve tentar novamente quando houver conexão

### FA002 - Campos Inválidos

**Condição:** `Controles.PossuiCamposInvalidos(this)`

**Ação:** Destaca campos obrigatórios não preenchidos

**Retorno:** Usuário deve preencher campos obrigatórios

### FA003 - Usuário Não Encontrado

**Condição:** `db.usuarios.FirstOrDefault(x => x.login == login) == null`

**Ação:** Exibe "Retomou Usuário Inválido. Login Cancelado."

**Retorno:** Usuário deve verificar credenciais

### FA004 - Senha Inválida

**Condição:** `senha != Criptografar.EncryptAES(txtSenha.Text)`

**Ação:** Exibe "Usuário ou Senha Inválido."

**Retorno:** Usuário deve verificar credenciais

### FA005 - Usuário Inativo

**Condição:** `!usu.ativo`

**Ação:** Exibe toast "Usuário Inativo!"

**Retorno:** Usuário deve contatar administrador

### FA006 - Erro de Sistema

**Condição:** Exception durante processo de login

**Ação:** Exibe `LmException.ShowException(ex, "Erro ao Logar no Sistema")`

**Retorno:** Foco retorna para campo usuário

## 7. Validações e Restrições

### 7.1 Validações de Entrada

- **Campo Usuário:** Obrigatório, máximo 30 caracteres
- **Campo Senha:** Obrigatório, máximo 100 caracteres (criptografado)
- **Conectividade:** Obrigatória para funcionamento

### 7.2 Validações de Negócio

- **Usuário deve existir** na tabela `usuarios`
- **Usuário deve estar ativo** (`ativo = true`)
- **Senha deve corresponder** ao hash AES armazenado
- **Máquina deve ser identificável** (hostname válido)

### 7.3 Restrições Técnicas

- **Singleton Pattern:** Apenas uma instância do formulário
- **Thread Safety:** Uso de `Invoke` para operações de UI
- **Gerenciamento de Recursos:** `using` statements para contexto de dados

## 8. Integração com Outros Módulos

## 8.1 Dependências

- **ContextoDados:** Acesso ao banco de dados
- **UcPainelTarefas:** Painel principal do sistema
- **Web:** Verificação de conectividade
- **Criptografar:** Criptografia de senhas
- **Toast/MsgBox:** Exibição de mensagens

## 8.2 Dados Utilizados

- **Tabela usuarios:** Credenciais e informações do usuário
- **Tabela usuario\_alocados:** Controle de sessões por máquina
- **Tabela configuracao\_api:** Configurações globais do sistema
- **Tabela perfis:** Permissões do usuário

## 8.3 Configurações Carregadas

- **API Token:** Para integração com serviços externos
- **Chave eDrawings:** Para visualização de desenhos
- **URL da API:** Endpoint dos serviços
- **Código da Empresa:** Identificação organizacional

## 9. Tratamento de Erros

### 9.1 Erros de Conectividade

- **Sem Internet:** Toast informativo
- **Erro de Banco:** Exception tratada com mensagem amigável

### 9.2 Erros de Autenticação

- **Credenciais Inválidas:** Mensagem padronizada
- **Usuário Inativo:** Toast específico
- **Usuário Não Encontrado:** Mensagem de erro

### 9.3 Erros de Sistema

- **Exception Geral:** `LmException.ShowException()`
- **Erro de Inicialização:** "Erro ao inicializar login"
- **Erro de Login:** "Erro ao Logar no Sistema"

## 10. Considerações de Segurança

### 10.1 Proteção de Dados

- **Senhas criptografadas** com AES
- **Não exposição** de senhas em logs
- **Validação server-side** de credenciais

### 10.2 Controle de Sessão

- **Alocação por máquina** impede múltiplos logins
- **Identificação única** por hostname + usuário PC
- **Limpeza de sessão** ao deslogar

### 10.3 Auditoria

- **Rastreamento de tentativas** de login
- **Identificação de máquina** para auditoria
- **Controle de usuários ativos**

## 11. Performance e Otimização

### 11.1 Otimizações Implementadas

- **Singleton Pattern** para instância única
- **Lazy Loading** de configurações

- **Consultas otimizadas** ao banco
- **Cache de permissões** após login

## 11.2 Considerações de Performance

- **Timeout de conectividade** configurável
- **Consultas assíncronas** quando possível
- **Liberação de recursos** com `using`
- **Minimização de round-trips** ao banco

---

**Documento gerado em:** \$(Get-Date)

**Versão:** 1.0

**Responsável:** Analista de Sistema

**Status:** Aprovado para Implementação