

```
# 📱 BccSistem - Sistema Integrado de Gestão Comercial

<p align="center">
    
    
    
    
</p>
```

```
<p align="center">
    <strong>Sistema completo de gestão comercial com controle de clientes, vendas, funcionários e análise de desempenho</strong>
</p>
```

📄 Índice

- [Sobre o Projeto] (#-sobre-o-projeto)
- [Funcionalidades] (#-funcionalidades)
- [Tecnologias] (#-tecnologias)
- [Arquitetura] (#-arquitetura)
- [Instalação] (#-instalação)
- [Estrutura do Projeto] (#-estrutura-do-projeto)
- [Módulos do Sistema] (#-módulos-do-sistema)
- [Segurança] (#-segurança)
- [API] (#-api)
- [Logs e Auditoria] (#-logs-e-auditoria)
- [Capturas de Tela] (#-capturas-de-tela)
- [Contribuindo] (#-contribuindo)
- [Licença] (#-licença)

🎯 Sobre o Projeto

O **BccSistem** é uma plataforma web robusta e completa para gestão comercial, desenvolvida em **PHP puro** com foco em **segurança**, **performance** e **usabilidade**. O sistema oferece controle total sobre o ciclo de vendas, desde o cadastro de clientes até o fechamento de contratos, incluindo análises detalhadas de desempenho e auditoria completa de todas as operações.

💡 Diferenciais

- ✅ **Autenticação Segura** com hash bcrypt e proteção contra força bruta
- ✅ **Controle de Acesso Baseado em Perfis (RBAC)** com 3 níveis de permissão
- ✅ **Sistema de Notificações em Tempo Real** para ações importantes
- ✅ **Logs Detalhados e Auditoria Completa** de todas as operações
- ✅ **Dashboard Analítico** com gráficos e comparação de desempenho

- **Gestão Completa de Vendas** com upload de contratos
 - **Soft Delete** para proteção de dados
 - **Exportação para Excel** de relatórios
 - **Interface Responsiva** e moderna
-

🔥 Funcionalidades

🔒 Autenticação e Autorização

- **Login Seguro** com proteção contra brute force
- **Recuperação de Senha** via token seguro
- **Três Perfis de Acesso:**
 - **Administrador**: Acesso total ao sistema
 - **Padrão**: Gerencia clientes e vendas próprias
 - **Aprendiz**: Visualização limitada e cadastro básico
- **Sessões Seguras** com HttpOnly, SameSite e regeneração de ID

👤 Gestão de Funcionários (Admin)

- Cadastro completo de funcionários
- Gerenciamento de perfis e permissões
- Ativação/Inativação de usuários
- Histórico de acessos e ações
- Filtros avançados (ativos/inativos/todos)

📄 Gestão de Clientes

- Cadastro completo com validação de dados
- Busca e filtros avançados por:
 - Nome, telefone, cidade, estado
 - Interesse (segmento de mercado)
 - Funcionário responsável
- Edição de informações
- Soft delete (exclusão lógica)
- Exportação para Excel
- **Identificação por Telefone** nas vendas (mais confiável)
- Associação automática ao funcionário

🛒 Gestão de Vendas

- Cadastro completo de vendas com:
 - Seleção de cliente por telefone
 - Vendedor e virador
 - Endereço completo
 - Dados do contrato
 - Segmento e administradora
 - Valor do crédito
 - Upload de arquivo do contrato (PDF/DOC/IMG)
- Edição de vendas (apenas admin)
- Visualização detalhada de contratos
- Exclusão lógica
- Exportação para Excel
- Notificações automáticas para envolvidos

📊 Dashboard Analítico

- **Visões Temporais:**
 - Diária: análise por dia
 - Semanal: análise por semana
 - Mensal: análise por mês
- **Gráficos Interativos** (Chart.js)
- **Comparação entre Funcionários** (admin)
- **Cards de Estatísticas:**
 - Total de clientes
 - Total de vendas
 - Funcionários ativos
 - Vendas do período
- Dados em tempo real via API

📣 Sistema de Notificações

- Notificações em tempo real
- **Tipos de notificações:**
 - ✓ Novo cliente cadastrado (com nome do responsável)
 - ✓ Nova venda registrada (com nome do registrador)
 - ✓ Funcionário inativado (com nome do responsável)
 - ⚠ Tentativas de login suspeitas
- Badge com contador no header
- Marcação de lidas/não lidas
- Filtros por tipo
- Links diretos para ações relacionadas

📄 Sistema de Logs Completo

- **Tipos de Logs:**
 - 🔴 Erros do sistema
 - ⚠ Avisos e alertas
 - 💡 Informações gerais
 - 🔒 Eventos de segurança
 - ⚡ Ações de usuários (CRUD)
- **Filtros Avançados:**
 - Por data (qualquer período)
 - Por nível de log
 - Por funcionário (mostra nome, não ID)
 - Por quantidade de registros
- Rotação automática de logs
- Limpeza de logs antigos
- Interface de visualização intuitiva
- Exportação de relatórios

📁 Exportação de Dados

- Exportação para Excel (.xlsx)
- Formatação automática
- Filtros aplicados mantidos
- Headers customizados
- Dados completos ou filtrados

Tecnologias

Backend

- **PHP 8.0+** - Linguagem principal
- **MySQL 8.0+** - Banco de dados
- **PDO** - Abstração de banco de dados com prepared statements
- **Composer** - Gerenciador de dependências

Frontend

- **HTML5** - Estrutura
- **CSS3** - Estilização (Design System próprio)
- **JavaScript (ES6+)** - Interatividade
- **Chart.js** - Gráficos interativos

Bibliotecas PHP

- **PhpSpreadsheet** - Geração de Excel
- **ZipStream** - Compactação de arquivos
- **voku/anti-xss** - Proteção contra XSS

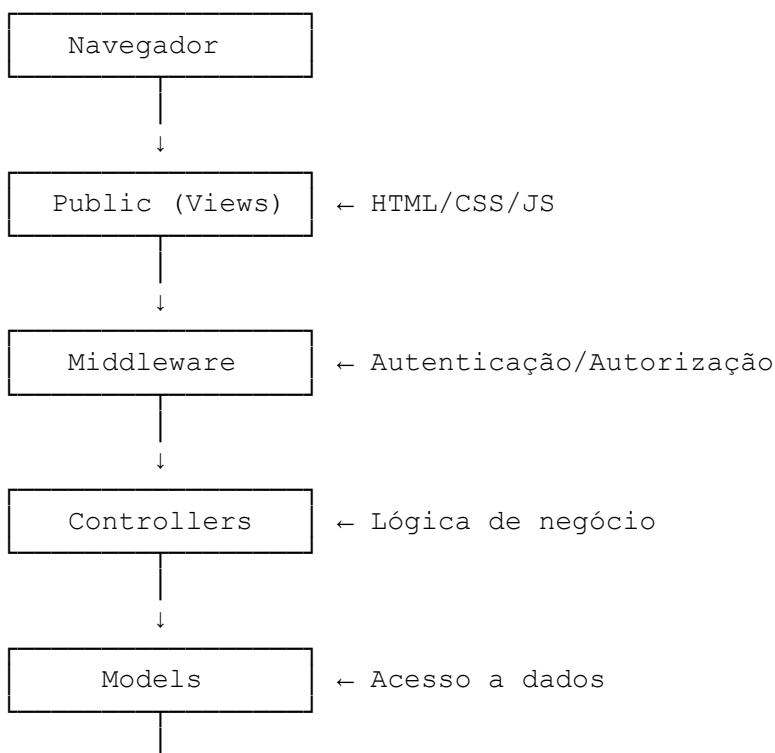
Segurança

- **CSRF Protection** - Tokens únicos por sessão
- **Password Hashing** - Bcrypt para senhas
- **SQL Injection Prevention** - Prepared statements
- **XSS Protection** - Sanitização de output
- **Session Security** - HttpOnly, SameSite, Secure

Arquitetura

O sistema segue uma arquitetura MVC adaptada com separação clara de responsabilidades:

```



↓

```
Database (PDO) ← MySQL
```

```

Camadas do Sistema

1. **Public** - Ponto de entrada (views e assets)
2. **Middleware** - Autenticação e autorização
3. **Models** - Lógica de dados e queries
4. **Lib** - Bibliotecas auxiliares (Auth, Logger, etc)
5. **Config** - Configurações do sistema

📦 Instalação

Pré-requisitos

- ```
```bash
- PHP >= 8.0
- MySQL >= 8.0 ou MariaDB >= 10.3
- Composer
- Servidor Web (Apache/Nginx)
- Extensões PHP: PDO, pdo_mysql, mbstring, json, fileinfo, zip
```
```

#### ### Passo a Passo

##### #### 1 Clone o Repositório

```
```bash
git clone https://github.com/seu-usuario/BccSistem.git
cd BccSistem
```
```

##### #### 2 Instale as Dependências

```
```bash
composer install
```
```

##### #### 3 Configure o Banco de Dados

```
3.1. Crie o banco:
```sql
CREATE DATABASE bcc CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
```
```

```
3.2. Importe a estrutura:
```bash
mysql -u root -p bcc < database/schema.sql
```
```

```
3.3. Crie a tabela de comissões (se necessário):
```bash
```

```
mysql -u root -p bcc < scripts/create_comissoes_table.sql
```

Ou execute diretamente no MySQL:

```
```sql
CREATE TABLE `comissoes` (
 `id` bigint(20) UNSIGNED NOT NULL AUTO_INCREMENT,
 `venda_id` bigint(20) UNSIGNED NOT NULL,
 `funcionario_id` bigint(20) UNSIGNED NOT NULL,
 `tipo_comissao` enum('vendedor','virador') NOT NULL,
 `parcela` varchar(50) NOT NULL,
 `numero_parcela` int(11) NOT NULL,
 `porcentagem` decimal(5,2) NOT NULL,
 `valor_base` decimal(10,2) NOT NULL,
 `valor_comissao` decimal(10,2) NOT NULL,
 `created_by` bigint(20) UNSIGNED NOT NULL,
 `created_at` timestamp NOT NULL DEFAULT current_timestamp(),
 PRIMARY KEY (`id`),
 KEY `idx_venda_id` (`venda_id`),
 KEY `idx_funcionario_id` (`funcionario_id`),
 KEY `idx_tipo_comissao` (`tipo_comissao`),
 CONSTRAINT `fk_comissoes_venda` FOREIGN KEY (`venda_id`) REFERENCES `vendas` (`id`) ON DELETE CASCADE,
 CONSTRAINT `fk_comissoes_funcionario` FOREIGN KEY (`funcionario_id`) REFERENCES `funcionarios` (`id`) ON DELETE CASCADE,
 CONSTRAINT `fk_comissoes_created_by` FOREIGN KEY (`created_by`) REFERENCES `funcionarios` (`id`) ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

```

4 Configure a Aplicação

4.1. Copie o arquivo de configuração:

```
```bash
cp app/config/config.php.example app/config/config.php
```

```

4.2. Edite suas credenciais em `app/config/config.php`:

```
```php
return [
 'db' => [
 'host' => '127.0.0.1',
 'dbname' => 'bcc',
 'user' => 'seu_usuario',
 'pass' => 'sua_senha',
 'charset' => 'utf8mb4',
],
 'app' => [
 'base_url' => '/BccSistem/public',
 'timezone' => 'America/Sao_Paulo',
],
];
```

```

5 Configure Permissões (Linux/Mac)

```
```bash
chmod -R 755 public/uploads
```

```

```
chmod -R 755 logs
```

```
``` bash  
Crie o Usuário Administrador
```

```
php scripts/seed_admin.php
```

```
Credenciais padrão:
- **Login:** `admin`
- **Senha:** `admin123`
```

⚠ \*\*IMPORTANTE:\*\* Altere a senha imediatamente após o primeiro acesso!

```
Acesse o Sistema
```

Abra seu navegador em:

```
http://localhost/BccSistem/public
```

---

```
📁 Estrutura do Projeto
```

```
BccSistem/
|
| └── app/
| | config/
| | config.php
| | config.php.example
| | administradoras.php
| | interesses.php
|
| └── lib/
| Auth.php
| Database.php
| Logger.php
| Notification.php
| CSRF.php
| Validator.php
| FileUpload.php
| PasswordReset.php
| Helpers.php
| Request.php
| Cache.php
| Backup.php
| ErrorHandler.php
| ActivityLogger.php
|
| └── middleware/
| require_login.php
| require_admin.php

Núcleo da aplicação
Configurações
Config principal (não versionado)
Template de config
Lista de administradoras
Segmentos de mercado

Bibliotecas do sistema
Autenticação
Conexão PDO
Sistema de logs
Notificações
Proteção CSRF
Validações
Upload de arquivos
Recuperação de senha
Funções auxiliares
Manipulação de requests
Sistema de cache
Backup do sistema
Tratamento de erros
Log de atividades

Middleware de autorização
Requer login
Requer admin
```

```
└── models/
 ├── Cliente.php
 ├── Funcionario.php
 ├── Venda.php
 ├── Dashboard.php
 └── Comissao.php

 # Models (camada de dados)
 # Model de clientes
 # Model de funcionários
 # Model de vendas
 # Model do dashboard
 # Model de comissões

 └── views/
 └── partials/
 ├── header.php
 └── footer.php

 # Views compartilhadas
 # Header padrão
 # Footer padrão

 └── public/
 ├── index.php
 ├── login.php
 ├── logout.php
 ├── dashboard.php
 ├── forgot_password.php
 ├── reset_password.php
 ├── notifications.php
 └── logs.php

 # Pasta pública (document root)
 # Página inicial
 # Login
 # Logout
 # Dashboard
 # Recuperar senha
 # Resetar senha
 # Notificações
 # Visualizar logs

 └── assets/
 └── css/
 ├── main.css
 ├── dashboard.css
 ├── clients.css
 ├── employees.css
 ├── forms.css
 ├── design-system.css
 └── bcc-brand.css

 # Assets estáticos
 # Estilos

 └── js/
 └── dashboard.js

 # Scripts

 └── clientes/
 ├── index.php
 ├── create.php
 ├── edit.php
 ├── delete.php
 └── export_excel.php

 # CRUD de Clientes
 # Listar
 # Criar
 # Editar
 # Excluir
 # Exportar

 └── vendas/
 ├── index.php
 ├── create.php
 ├── edit.php
 ├── delete.php
 ├── view.php
 └── export_excel.php

 # CRUD de Vendas
 # Listar
 # Criar
 # Editar (admin)
 # Excluir (admin)
 # Visualizar
 # Exportar

 └── funcionarios/
 ├── index.php
 ├── create.php
 └── edit.php

 # CRUD de Funcionários
 # Listar
 # Criar (admin)
 # Editar (admin)

 └── comissoes/
 # Sistema de Comissões (admin)
```

```

 └── index.php # Tela principal
 └── create.php # Gerar comissão

 └── api/
 ├── clients.php # APIs REST
 ├── dashboard_counts.php # API de clientes
 ├── cliente_info.php # API do dashboard
 └── cliente_info.php # Info de cliente

 └── uploads/ # Arquivos enviados (não versionado)
 └── contratos/ # Contratos de vendas

 └── logs/ # Logs do sistema (não versionado)
 ├── errors_YYYY-MM-DD.log # Logs de erros
 ├── actions_YYYY-MM-DD.log # Logs de ações
 ├── security_YYYY-MM-DD.log # Logs de segurança
 └── system_YYYY-MM-DD.log # Logs gerais

 └── scripts/ # Scripts auxiliares
 ├── seed_admin.php # Criar admin
 ├── cleanup_logs.php # Limpar logs antigos
 ├── health_check.php # Verificar saúde do sistema
 └── create_comissoes_table.sql # Script SQL da tabela comissões

 └── vendor/ # Dependências Composer (não
versionado)

 └── docs/ # Documentação
 └── screenshots/ # Capturas de tela

 .gitignore # Arquivos ignorados pelo Git
 composer.json # Dependências PHP
 composer.lock # Lock de dependências
 README.md # Este arquivo
 README_INSTALACAO.md # Guia de instalação
 CONFIGURACAO_GIT.md # Configuração do Git
 LICENSE # Licença MIT
```

```

```

## ## 🔮 Módulos do Sistema

### ### 1. 🧑 Autenticação e Autorização

#### \*\*Funcionalidades:\*\*

- Login seguro com proteção contra brute force
- Recuperação de senha via e-mail/token
- Três níveis de acesso (Admin, Padrão, Aprendiz)
- Sessões seguras com timeout automático
- Logs de tentativas de acesso

#### \*\*Rotas:\*\*

- `GET/POST /login.php` - Tela de login
- `GET /logout.php` - Encerrar sessão
- `GET/POST /forgot\_password.php` - Recuperar senha
- `GET/POST /reset\_password.php` - Resetar senha

---

### ### 2. Dashboard

#### \*\*Funcionalidades:\*\*

- Visão geral do sistema
- Gráficos de vendas por período
- Comparação entre funcionários
- Cards com estatísticas em tempo real
- Filtros por data e usuário

#### \*\*Rotas:\*\*

- `GET /dashboard.php` - Dashboard principal
- `GET /api/dashboard\_counts.php` - API de estatísticas

#### \*\*API Endpoint:\*\*

```

GET /api/dashboard_counts.php?mode=week&start=2025-01-01&end=2025-01-07

```

#### \*\*Resposta:\*\*

```json

```
{  
    "ok": true,  
    "mode": "week",  
    "labels": ["Semana 1", "Semana 2"],  
    "series": {  
        "1": {  
            "name": "João Silva",  
            "data": [5, 8],  
            "total": 13  
        }  
    }  
}
```

```

---

### ### 3. Gestão de Clientes

#### \*\*Funcionalidades:\*\*

- CRUD completo de clientes
- Busca avançada com múltiplos filtros
- Validação de dados (nome, telefone, etc)
- Soft delete (exclusão lógica)
- Exportação para Excel
- Associação automática ao funcionário

#### \*\*Rotas:\*\*

- `GET /clientes/index.php` - Listar clientes
- `GET/POST /clientes/create.php` - Criar cliente
- `GET/POST /clientes/edit.php?id={id}` - Editar cliente
- `POST /clientes/delete.php?id={id}` - Excluir cliente
- `GET /clientes/export\_excel.php` - Exportar Excel

#### \*\*Validações:\*\*

- Nome completo (mínimo 3 caracteres)

- Telefone no formato brasileiro
- Estado com 2 letras
- Interesse válido (conforme configuração)

---

### ### 4. Gestão de Vendas

#### \*\*Funcionalidades:\*\*

- CRUD completo de vendas
- Seleção de cliente por telefone (mais confiável)
- Cadastro de vendedor e virador
- Endereço completo do contrato
- Upload de arquivo do contrato
- Notificações automáticas
- Visualização detalhada
- Exportação para Excel
- Edição e exclusão (apenas admin)

#### \*\*Rotas:\*\*

- `GET /vendas/index.php` - Listar vendas
- `GET/POST /vendas/create.php` - Criar venda
- `GET/POST /vendas/edit.php?id={id}` - Editar venda (admin)
- `GET /vendas/view.php?id={id}` - Visualizar venda
- `POST /vendas/delete.php?id={id}` - Excluir venda (admin)
- `GET /vendas/export\_excel.php` - Exportar Excel

#### \*\*Dados da Venda:\*\*

- Cliente (busca por telefone)
- Vendedor e Virador
- Número do contrato (único)
- Endereço completo
- CPF do cliente
- Segmento de mercado
- Tipo (Normal/Meia)
- Administradora
- Valor do crédito
- Arquivo do contrato (opcional)

---

### ### 5. Gestão de Funcionários

#### \*\*Funcionalidades:\*\*

- CRUD completo (apenas admin)
- Gerenciamento de perfis
- Ativação/Inativação
- Senha opcional na edição
- Validação de login único
- Logs de todas as alterações

#### \*\*Rotas:\*\*

- `GET /funcionarios/index.php` - Listar funcionários (admin)
- `GET/POST /funcionarios/create.php` - Criar funcionário (admin)
- `GET/POST /funcionarios/edit.php?id={id}` - Editar funcionário (admin)

#### \*\*Perfis Disponíveis:\*\*

1. \*\*Administrador (role\_id = 1)\*\*

- Acesso total ao sistema
  - Gerencia funcionários
  - Edita e exclui vendas
  - Visualiza todos os dados
2. \*\*Padrão (role\_id = 2)\*\*
- Cadastra clientes e vendas
  - Visualiza próprios clientes
  - Visualiza próprias vendas
  - Dashboard com dados próprios
3. \*\*Aprendiz (role\_id = 3)\*\*
- Cadastra clientes
  - Visualiza próprios clientes
  - Acesso limitado ao dashboard

---

### ### 6. 💰 Sistema de Comissões

#### \*\*Funcionalidades:\*\*

- Gestão completa de comissões de vendedores e viradores
- Comissões separadas por tipo (vendedor/virador)
- Controle de parcelas até parcela final
- Regra especial para Gazin + Meia Parcela (50% do valor base)
- Visualização de comissões por funcionário
- Filtro por mês
- Estatísticas de comissões geradas
- Logs e notificações automáticas

#### \*\*Acesso:\*\*

- \*\*Apenas Administradores\*\* podem acessar e gerenciar comissões

#### \*\*Rotas:\*\*

- `GET /comissoes/index.php` - Tela principal de comissões
- `GET /comissoes/index.php?tipo=vendedor&funcionario\_id={id}` - Listar vendas do vendedor
- `GET /comissoes/index.php?tipo=virador&funcionario\_id={id}` - Listar vendas do virador
- `GET /comissoes/index.php?visualizar=comissoes&funcionario\_comissao\_id={id}` - Visualizar comissões geradas
- `GET/POST /comissoes/create.php` - Gerar nova comissão

#### \*\*Regras Especiais:\*\*

- \*\*Gazin + Meia Parcela:\*\* Se administradora = "Gazin" E tipo de venda = "Meia", o valor base para cálculo de comissão será 50% do valor do crédito
- \*\*Controle de Parcelas:\*\* Vendedores e viradores têm parcelas independentes. Quando "Parcela Final" é gerada para um tipo, a venda não aparece mais para aquele tipo (mas pode aparecer para o outro tipo se ainda houver parcelas pendentes)

#### \*\*Exemplo de Cálculo:\*\*

```

Venda: R\$ 1.000,00

Administradora: Gazin

Tipo: Meia Parcela
Porcentagem: 5,50%

Valor Base = 1.000,00 ÷ 2 = R\$ 500,00
Valor Comissão = 500,00 × 5,50% = R\$ 27,50
` `` `

7. Sistema de Notificações

Funcionalidades:

- Notificações em tempo real
- Badge com contador no header
- Filtros e busca
- Marcação como lida
- Links diretos para ações
- Tipos diferenciados por cor

Rotas:

- `GET /notifications.php` - Listar notificações
- `GET /notifications.php?action=mark_read&id={id}` - Marcar como lida
- `GET /notifications.php?action=mark_all_read` - Marcar todas

Tipos de Notificações:

-  **Success** - Ações bem-sucedidas
-  **Info** - Informações gerais
-  **Warning** - Avisos importantes
-  **Error** - Erros críticos

Eventos que Geram Notificações:

- Novo cliente cadastrado (notifica admins)
- Nova venda registrada (notifica vendedor e virador)
- Funcionário inativado (notifica admins)
- Tentativas de login suspeitas (notifica admins)
- Comissão gerada (notifica admins)

8. Sistema de Logs

Funcionalidades:

- Logs detalhados de todas as operações
- Filtros avançados (data, nível, usuário)
- Exibição de nomes de funcionários
- Rotação automática de arquivos
- Limpeza de logs antigos
- Exportação de relatórios

Rotas:

- `GET /logs.php` - Visualizar logs (admin)
- `GET /logs.php?date=YYYY-MM-DD&level=ERROR&user_id=1` - Filtrar logs

Tipos de Logs:

-  **ERROR** - Erros do sistema
-  **WARNING** - Avisos e alertas

- **INFO** - Informações gerais
- **SECURITY** - Eventos de segurança
- **ACTION** - Ações de usuários (CRUD)

Estrutura do Log:

```
```
[2025-10-17 14:30:45] [ACTION] [João Silva] CRUD: CREATE em vendas |
{"operation":"CREATE","table":"vendas","record_id":123}
```
```

Arquivos de Log:

- `errors_YYYY-MM-DD.log` - Erros
- `warnings_YYYY-MM-DD.log` - Avisos
- `security_YYYY-MM-DD.log` - Segurança
- `actions_YYYY-MM-DD.log` - Ações
- `system_YYYY-MM-DD.log` - Sistema geral

Segurança

Proteções Implementadas

1. Autenticação

- Senhas com **bcrypt** (hash seguro)
- Proteção contra **brute force** (limite de tentativas)
- Regeneração de **session ID** após login
- Tokens seguros para recuperação de senha
- Timeout de sessão configurável

2. Autorização

- **RBAC** (Role-Based Access Control)
- Middleware de autenticação em todas as rotas
- Verificação de permissões por perfil
- Logs de acessos negados

3. Proteção de Dados

- **Prepared Statements** em todas as queries SQL
- Sanitização de **input** com validadores
- Escape de **output** (função `e()`)
- Proteção contra **SQL Injection**
- Proteção contra **XSS** (voku/anti-xss)

4. Sessões

- Cookies **HttpOnly** (não acessíveis via JavaScript)
- Cookies **SameSite=Lax** (proteção CSRF)
- Cookies **Secure** em HTTPS
- Regeneração de ID em ações críticas

5. CSRF Protection

- Tokens únicos por sessão
- Validação em todos os formulários POST
- Expiração automática de tokens

```
#### 6. Upload de Arquivos
- ✓ Validação de tipo (whitelist)
- ✓ Validação de tamanho (máx 5MB)
- ✓ Renomeação automática de arquivos
- ✓ Armazenamento fora do document root (uploads/)
- ✓ Proteção via `htaccess`
```

```
#### 7. Logs e Auditoria
- ✓ Log de todas as operações críticas
- ✓ Log de tentativas de acesso
- ✓ Log de alterações de dados
- ✓ Rastreamento de IP e User-Agent
```

🛡 API

O sistema possui APIs REST para integração e consumo de dados.

Endpoints Disponíveis

1. Dashboard API

Endpoint: `GET /api/dashboard_counts.php`

Parâmetros:

- `mode` - Modo de visualização: `day`, `week`, `month`
- `start` - Data inicial (YYYY-MM-DD) - para modo week
- `end` - Data final (YYYY-MM-DD) - para modo week
- `month` - Mês (YYYY-MM) - para modo month
- `day` - Dia (YYYY-MM-DD) - para modo day
- `users[]` - Array de IDs de usuários (admin) - opcional

Exemplo:

```
```bash
GET /api/dashboard_counts.php?mode=week&start=2025-01-01&end=2025-01-07&users[]=1&users[]=2
````
```

Resposta:

```
```json
{
 "ok": true,
 "mode": "week",
 "start": "2025-01-01",
 "end": "2025-01-07",
 "labels": ["Semana 1 (01-07 Jan)", "Semana 2 (08-14 Jan)"],
 "series": {
 "1": {
 "name": "João Silva",
 "data": [5, 8],
 "total": 13
 },
 "2": {
 "name": "Maria Santos",
 }
 }
}
```

```
 "data": [3, 6],
 "total": 9
 }
}
```
``
```

2. Clientes API

Endpoint: `GET /api/clients.php`

Parâmetros:

- `page` - Página (default: 1)
- `limit` - Limite por página (default: 25, máx: 100)
- `search` - Busca por nome, telefone ou cidade
- `interesse` - Filtro por interesse
- `estado` - Filtro por estado
- `criado_por` - Filtro por funcionário (admin)

Exemplo:

```
```bash
GET /api/clients.php?search=João&interesse=Crédito&page=1&limit=25
```
``
```

Resposta:

```
```json
{
 "success": true,
 "data": [
 {
 "id": 1,
 "nome": "João Silva",
 "telefone": "11999999999",
 "telefone_formatado": "(11) 99999-9999",
 "cidade": "São Paulo",
 "estado": "SP",
 "interesse": "Crédito",
 "created_at": "2025-01-15 10:30:00",
 "criado_por_nome": "Admin"
 }
],
 "pagination": {
 "page": 1,
 "limit": 25,
 "total": 150,
 "pages": 6
 }
}
```
``
```

3. Cliente Info API

Endpoint: `GET /api/cliente_info.php?id={id}`

Resposta:

```
```json
{
 "success": true,
```

```
"cliente": {
 "id": 1,
 "nome": "João Silva",
 "telefone": "11999999999",
 "cidade": "São Paulo",
 "estado": "SP",
 "interesse": "Crédito"
}
```
```

```

---

## ## 📊 Logs e Auditoria

### ### Sistema de Logs

O BccSistem possui um sistema completo de logs que registra todas as operações importantes do sistema.

#### #### Tipos de Logs

1. \*\*Erros (ERROR)\*\*
  - Erros de sistema
  - Exceções não tratadas
  - Falhas de conexão
  - Erros de validação críticos
2. \*\*Avisos (WARNING)\*\*
  - Tentativas de login falhadas
  - Operações inválidas
  - Dados inconsistentes
3. \*\*Informações (INFO)\*\*
  - Logins bem-sucedidos
  - Operações bem-sucedidas
  - Eventos importantes
4. \*\*Segurança (SECURITY)\*\*
  - Acessos negados
  - Tentativas de brute force
  - Ações suspeitas
5. \*\*Ações (ACTION)\*\*
  - CREATE - Criações de registro
  - UPDATE - Atualizações de registro
  - DELETE - Exclusões de registro
  - READ - Consultas importantes
6. \*\*Comissões (COMISSAO\_CREATED)\*\*
  - Geração de comissões de vendedores e viradores
  - Registra todos os detalhes: parcela, porcentagem, valores

#### #### Visualização de Logs

Acesse `/logs.php` (apenas admin) para:

- Filtrar por data específica
- Filtrar por nível de log

- Filtrar por funcionário
- Ver contexto completo de cada log
- Exportar relatórios

#### #### Rotação de Logs

- Arquivos são rotacionados quando atingem 10MB
- Mantém até 5 versões de cada arquivo
- Limpeza automática de logs com mais de 30 dias

#### #### Estrutura do Log

```

```
[TIMESTAMP] [NÍVEL] [USUÁRIO] MENSAGEM | {"contexto": "em JSON"}  
```
```

\*\*Exemplo:\*\*

```

```
[2025-10-17 14:30:45] [ACTION] [1] CRUD: CREATE em vendas |  
{"operation": "CREATE", "table": "vendas", "record_id": 123, "data": {"numero_contrato": "12345", "valor_credito": "5000.00"} }  
```
```

---

#### ## 📸 Capturas de Tela

##### ### Tela de Login

! [Login] (docs/screenshots/login.png)

##### ### Dashboard

! [Dashboard] (docs/screenshots/dashboard.png)

##### ### Gestão de Clientes

! [Clientes] (docs/screenshots/clientes.png)

##### ### Gestão de Vendas

! [Vendas] (docs/screenshots/vendas.png)

##### ### Sistema de Notificações

! [Notificações] (docs/screenshots/notificacoes.png)

##### ### Logs do Sistema

! [Logs] (docs/screenshots/logs.png)

---

#### ## 💪 Contribuindo

Contribuições são bem-vindas! Siga os passos abaixo:

##### ### 1. Fork o Projeto

```bash

```
git clone https://github.com/seu-usuario/BccSistem.git  
cd BccSistem  
```
```

```
2. Crie uma Branch
```

```
```bash
git checkout -b feature/minha-funcionalidade
```
```

```
3. Faça suas Alterações
```

- Escreva código limpo e documentado
- Siga os padrões do projeto
- Adicione testes se possível

```
4. Commit suas Mudanças
```

```
```bash
git add .
git commit -m "feat: Adiciona nova funcionalidade X"
```
```

\*\*Padrão de commits:\*\*

- `feat:` - Nova funcionalidade
- `fix:` - Correção de bug
- `docs:` - Documentação
- `style:` - Formatação
- `refactor:` - Refatoração de código
- `test:` - Testes
- `chore:` - Tarefas de manutenção

```
5. Push para o GitHub
```

```
```bash
git push origin feature/minha-funcionalidade
```
```

```
6. Abra um Pull Request
```

- Descreva suas alterações
- Referencie issues relacionadas
- Aguarde revisão

---

## ## 📄 Licença

Este projeto está sob a licença \*\*MIT\*\*. Veja o arquivo [LICENSE] (LICENSE) para mais detalhes.

```

MIT License

Copyright (c) 2025 BccSistem

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to
deal
in the Software without restriction, including without limitation the
rights

to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

```

---

## ## Suporte e Contato

Para dúvidas, sugestões ou problemas:

-  \*\*E-mail:\*\* suporte@bccsistem.com
-  \*\*Issues:\*\* [GitHub Issues] (<https://github.com/seu-usuario/BccSistem/issues>)
-  \*\*Documentação:\*\* [Wiki do Projeto] (<https://github.com/seu-usuario/BccSistem/wiki>)

---

## ## Agradecimentos

Desenvolvido com  pela equipe BccSistem

---

```
<p align="center">
 ★ Se este projeto foi útil para você, considere dar uma estrela no GitHub! ★
</p>
```

```
<p align="center">
 ⬆ Voltar ao topo
</p>
```