

Sistemas de Gestão de Certificados Eletrônicos (SGCE)

**Última revisão:
21/11/2016**

2016

UNIVERSIDADE FEDERAL DO PAMPA

REITOR

Marco Antonio Fontoura Hansen

VICE-REITOR

Maurício Aires Vieira

DIRETORIA DE TECNOLOGIA DA INFORMAÇÃO E COMUNICAÇÃO

DIRETOR

Pierre Correa Martin

COORDENADOR DE DESENVOLVIMENTO

Sérgio Bortolin Júnior

Documentação

André Luiz de Matos Camargo

Pedro Conrad Junior

Sergio Antonio Martini Bortolin Junior

Sumário

1. Informações Técnicas.....	5
1.1 Informações sobre Arquitetura e Tecnologias.....	5
1.1.1 Framework de Desenvolvimento Codeigniter.....	5
1.1.2 Banco de Dados PostgreSQL.....	6
1.2 Requisitos do Sistema.....	6
2. Instalação do SGCE	7
2.1. Instalando PHP, Apache, PostgreSQL e extensions.....	7
2.2. Instalando a aplicação.....	7
2.3. Configurando o Apache e PHP.....	8
2.3.1. Habilitando a reescrita de URLs.....	8
2.3.2. Configuração das diretivas de permissão de acesso do Apache.....	8
2.3.3. Configuração das diretivas do php.ini.....	9
2.4. Configurando phpPgAdmin.....	9
2.5. Criando um usuário no PostgreSQL para o sistema.....	10
2.6. Criando o banco de dados.....	10
2.7. Configurando a aplicação para conectar ao banco de dados.....	11
2.8. Localização do Sistema na Web.....	12
2.9. Criptografia e Chaves.....	13
2.10. Permissões de Arquivo.....	13
2.11 Primeiro uso.....	14

1. Informações Técnicas

1.1 Informações sobre Arquitetura e Tecnologias

Este sistema foi desenvolvido utilizando a linguagem PHP de forma orientada a objetos. Para agilizar os processos de desenvolvimento foi utilizado o framework de desenvolvimento Codeigniter versão 1.7.3.

Já para o armazenamento de dados foi utilizado o banco de dados PostgreSQL, versão 8.4.

Além disto, o desenvolvimento segue um padrão de interface e estrutura definido pela equipe do DTIC, já utilizado no desenvolvimento de outros sistemas web da Instituição.

1.1.1 Framework de Desenvolvimento Codeigniter

O objetivo do framework Codeigniter é fornecer um conjunto de bibliotecas para as tarefas mais comuns que necessitem ser realizadas dentro da estrutura de programação, além de fornecer uma interface de acesso a estas bibliotecas, possibilitando um processo de desenvolvimento mais rápido, mantendo o foco dos desenvolvedores em seu projeto, minimizando a quantidade de código que é necessária para que as tarefas sejam realizadas e os problemas do projeto resolvidos.

O Codeigniter é um framework leve, com ampla compatibilidade entre diversas versões e configurações típicas do PHP, que necessita de pouca ou nenhuma configuração para começar o desenvolvimento, sem necessidade de se utilizar linhas de comando. Mais informações, bem como a documentação completa do framework estão disponíveis em <http://www.codeigniter.com/>.

1.1.2 Banco de Dados PostgreSQL

O PostgreSQL é um sistema gerenciador de banco de dados objeto-relacional de código aberto largamente utilizado em todo o mundo. Este sistema roda em todos os grandes sistemas operacionais, incluindo GNU/Linux, Unix, e MS Windows. É totalmente compatível com ACID, tendo suporte completo a chaves estrangeiras, junções, visões, gatilhos e procedimentos armazenados. Inclui a maior parte dos tipos de dados do ISO SQL:1999, suportando também o armazenamento de objetos binários, incluindo figuras, sons ou vídeos.

Foi escolhido devido às suas características funcionais, sua estabilidade, confiabilidade e desenvolvimento ativo. Para mais informações, consulte o site: <http://www.postgresql.org/>.

1.2 Requisitos do Sistema

Para utilizar o Sistema Gerenciador de Certificados Eletrônicos, será necessário instalar em seu ambiente os seguintes pacotes:

- Apache2 com mod_rewrite habilitado.
- PHP5, com suporte a GD2, PostgreSQL e GetText
- Servidor de Banco de Dados PostgreSQL
- Servidor de Diretório LDAP (Opcional)

2. Instalação do SGCE

Nesse tutorial de instalação utilizamos as seguintes versões:

- *Ubuntu 14.04 ou 14.10*
- *Apache 2.4.10*
- *PHP 5.5.12*
- *PostgreSQL 9.4.3*

2.1. Instalando PHP, Apache, PostgreSQL e extensions

Abrir o terminal de linha de comando e executar:

```
sudo apt-get update  
  
sudo apt-get install apache2 php5 libapache2-mod-php5  
postgresql php5-pgsql phppgadmin php-gettext php5-gd
```

2.2. Instalando a aplicação

Deve-se extrair o pacote da versão (*sgce_X.X.X.zip*) em */var/www/html/*

A estrutura de diretórios ficará algo como:

```
/var/www/html/sgce/index.php
```

NOTA: Caso não exista permissão de escrita em */var/www/html/*, deve-se conceder a permissão com o comando *chmod*.

2.3. Configurando o Apache e PHP

2.3.1. Habilitando a reescrita de URLs

Para utilizar 'URLs limpas', no diretório raiz do SGCE existe um arquivo de nome `.htaccess` com as diretivas de configuração. Para que este recurso funcione o módulo `Rewrite` do Apache deve estar habilitado. Para habilitar o módulo, utilize o comando:

```
sudo a2enmod rewrite
```

2.3.2. Configuração das diretivas de permissão de acesso do Apache

Para permitir o acesso ao sistema e para que a reescrita de URLs funcione adequadamente, utilize no arquivo de configuração do seu servidor Apache (normalmente em `/etc/apache2/apache2.conf`) uma chave semelhante à seguinte:

```
<Directory /var/www/html/sgce>  
  
    Options Indexes FollowSymLinks  
  
    AllowOverride All  
  
    Allow from All  
  
</Directory>
```


2.3.3. Configuração das diretivas do php.ini

É necessário habilitar `short_open_tag` e desabilitar o `display_errors` no arquivo `"php.ini"` (normalmente em `/etc/php5/apache2/php.ini`).

Essas diretivas devem ficar assim:

```
short_open_tag = on
```

```
display_errors = off
```

Após salvar o arquivo, deve-se reiniciar o apache:

```
sudo service apache2 restart
```

2.4. Configurando phpPgAdmin

Execute os comandos a seguir para habilitar o acesso ao phpPgAdmin via browser (exemplo: `http://localhost/phppgadmin`):

```
sudo cp /etc/apache2/conf.d/phppgadmin /etc/apache2/conf-enabled/phppgadmin.conf
```

```
sudo service apache2 restart
```

2.5. Criando um usuário no PostgreSQL para o sistema

Execute os comandos a seguir para criar um usuário no PostgreSQL. O sistema irá utilizar este usuário para conectar ao banco de dados.

```
sudo -u postgres psql
```

```
CREATE USER sgce SUPERUSER INHERIT CREATEDB CREATEROLE;
```

```
ALTER USER sgce PASSWORD '12345678';
```

Atenção: Deve-se definir uma senha segura.

2.6. Criando o banco de dados

Abrir o phpPgAdmin no browser (<http://localhost/phppgadmin/>) e logar com o usuário 'sgce', criado no passo anterior. Logo após, deve-se criar um banco de dados com o nome 'sgce' (UTF-8). Após criar o banco de dados, ir na opção "SQL" do phpPgAdmin e executar o script de criação das tabelas (o script encontra-se em sgce/sql/sgce.sql). O script pode ser enviado por *upload*, conforme Imagem 1. Não esquecer de desmarcar a opção "Paginate results".

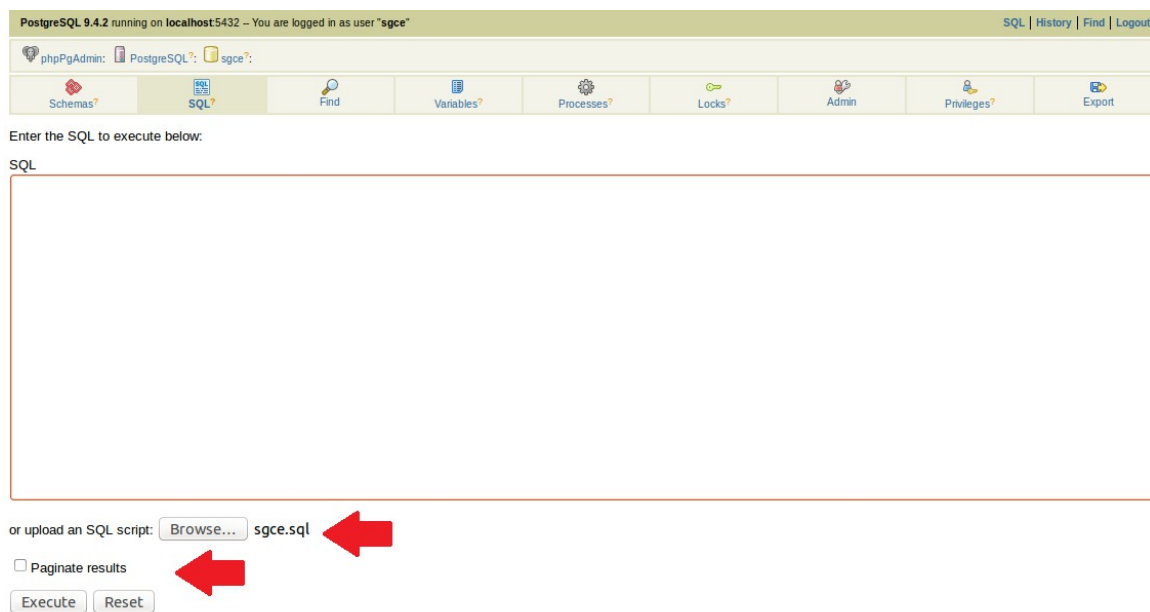


Imagem 1 - Upload do script SQL

2.7. Configurando a aplicação para conectar ao banco de dados

Depois de criar o banco de dados em seu servidor, abra o arquivo `sgce/system/application/config/database.php`, e edite as seguintes chaves:

`$db['default']['hostname'] = Endereço do servidor de banco de dados.`

`$db['default']['port'] = Porta padrão de escuta do banco.`

`$db['default']['username'] = Nome do usuário do banco de dados.`

`$db['default']['password'] = Senha do usuário do banco de dados.`

`$db['default']['database'] = Nome da base de dados.`

Exemplo:

`$db['default']['hostname'] = 'localhost';`

`$db['default']['port'] = 5432;`

`$db['default']['username'] = 'sgce';`

`$db['default']['password'] = '12345678';`

`$db['default']['database'] = 'sgce';`

2.8. Localização do Sistema na Web

Edite a chave `$config['base_url']` no arquivo `sgce/system/application/config/config.php`. Esta chave deverá conter o endereço web da raiz do sistema.

Edite também o arquivo `sgce/system/application/config/constants.php` e modifique as constantes de uso do SGCE no final do arquivo para refletir seu ambiente de uso, principalmente no que se refere à constante `URL_CERTIFICADO` que deverá conter o endereço do sistema para gerar corretamente os links de emissão e validação dos certificados.

2.9. Criptografia e Chaves

Este sistema utiliza uma chave de criptografia de 256 bits para auxiliar na geração dos certificados. Atribua sua chave editando o arquivo `sgce/system/application/config/config.php` e alterando o valor de `$config['encryption_key']`. Uma chave válida possui 32 caracteres e é composta de caracteres alfanuméricos.

NOTA: É recomendável que se utilize uma chave diferente para cada base de dados que houver. O uso desta chave implica na verificação de autenticidade e na emissão do certificado.

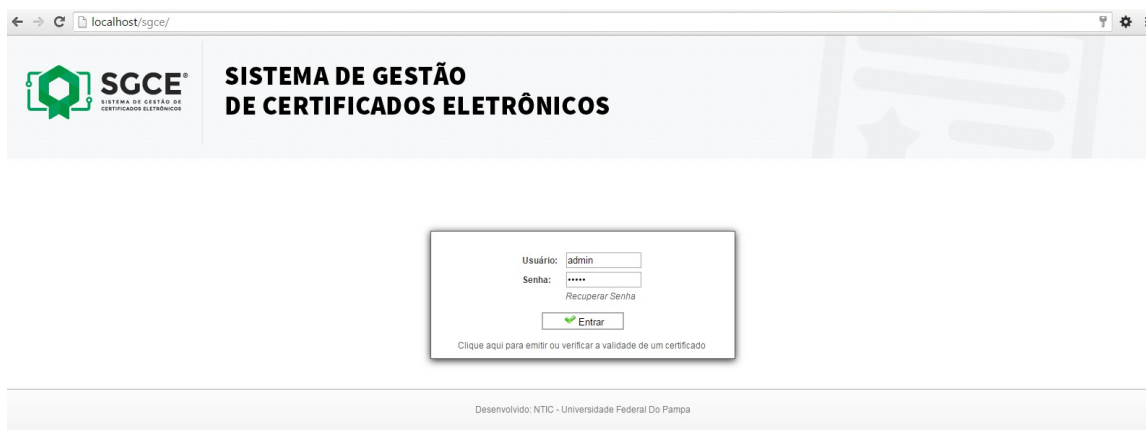
2.10. Permissões de Arquivo

Para que o envio de arquivos de fundo dos certificados (espelhos) funcione corretamente, é recomendável que se executem os seguintes comandos, para estabelecer o dono e as permissões dos arquivos, conforme o exemplo a seguir, utilizando como dono o usuário e grupo padrão do apache no Ubuntu (www-data):

```
sudo chown www-data:www-data /var/www/html/sgce/  
sudo chown www-data:www-data /var/www/html/sgce/* -R  
sudo chmod 775 /var/www/html/sgce/  
sudo chmod 775 /var/www/html/sgce/* -R
```

2.11 Primeiro uso

Para o primeiro acesso ao sistema, navegue até a URL onde o sistema está instalado e utilize o nome de usuário 'admin' e senha 'admin' (sem aspas), conforme Imagem 2.



The image shows a web browser window displaying the login page for the SGCE (Sistema de Gestão de Certificados Eletrônicos). The browser's address bar shows 'localhost/sgce/'. The page header includes the SGCE logo and the text 'SISTEMA DE GESTÃO DE CERTIFICADOS ELETRÔNICOS'. The main content area features a login form with fields for 'Usuário:' (containing 'admin') and 'Senha:' (containing '*****'). Below the password field is a link for 'Recuperar Senha'. A green 'Entrar' button is positioned below the form. A small text link at the bottom of the form reads 'Clique aqui para emitir ou verificar a validade de um certificado'. The footer of the page states 'Desenvolvido: NTIC - Universidade Federal Do Pampa'.

Imagem 2 - Tela de Login do SGCE