



**SENADO FEDERAL**  
**Secretaria Especial do Interlegis - SINTER**  
**Subsecretaria de Tecnologia da Informação - SSTIN**



**Produto III**  
**Sistema de Informações Gerenciais do Interlegis**  
**APO-CASA**

**Guilherme Mesquita Gondim**  
Contrato N°: 2008/000471



## Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Terminologia . . . . .	1
<b>2</b>	<b>Anexo - Arquivo LEIA-ME</b>	<b>2</b>
<b>3</b>	<b>Anexo - Arquivo visaogeral.txt</b>	<b>2</b>
<b>4</b>	<b>Anexo - Arquivo instalacao.txt</b>	<b>4</b>



# 1 Introdução

Este documento detalha as atividades desenvolvidas durante a terceira etapa do projeto de desenvolvimento do Sistema de Informações Gerenciais do Interlegis (SIGI).

A terceira etapa consiste na documentação dos processos de instalação, configuração e implantação do SIGI em um servidor GNU/Linux.

Na seção 2 está disponível o arquivo LEIA-ME.

Na seção 3 está anexado o arquivo `visaogeral.txt`, o qual apresenta uma visão geral do sistema.

A seção 4 tem anexado a documentação dos processos de instalação e configuração.

No Colab, portal colaborativo para a gerência dos projetos de software do Interlegis, foi publicada a página do projeto SIGI, que pode ser acessada através do link <http://colab.interlegis.gov.br/wiki/ProjetoSigi>.

## 1.1 Terminologia

**Sistema Gerenciador de Bancos de Dados (SGDB):** conjunto de programas de computador (softwares) responsáveis pelo gerenciamento de uma base de dados;

**Servidor web:** um programa de computador responsável por servir dados e páginas web, tais como documentos HTML com objetos embutidos (imagens, etc.);

**Django:** um framework de desenvolvimento web de alto nível escrito em linguagem Python e que estimula o desenvolvimento rápido e limpo.



## 2 Anexo - Arquivo LEIA-ME

SIGI é um projeto para um Sistema de Informações Gerenciais do Interlegis, escrito na linguagem de programação Python com o framework para desenvolvimento web Django.

Copyright (c) 2008 Interlegis

SIGI é um software livre; você pode redistribuí-lo e/ou modificá-lo dentro dos termos da Licença Pública Geral GNU como publicada pela Fundação do Software Livre (FSF); na versão 3 da Licença, ou (na sua opinião) em qualquer versão mais recente.

SIGI é distribuído na esperança que possa ser útil, mas SEM NENHUMA GARANTIA; sem uma garantia implícita de ADEQUAÇÃO a qualquer MERCADO ou APLICAÇÃO EM PARTICULAR. Veja a Licença Pública Geral GNU para maiores detalhes.

Você deve ter recebido uma cópia da Licença Pública Geral GNU junto com este programa; veja o arquivo COPYING. Se não, veja <http://www.gnu.org/licenses/>.

. . .

SIGI is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.

SIGI is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; see the file COPYING. If not, see <http://www.gnu.org/licenses/>.

## 3 Anexo - Arquivo visaogeral.txt

O SIGI é um projeto para um Sistema de Informações Gerenciais do [Interlegis](#), escrito na linguagem de programação [Python](#) com o framework



para desenvolvimento web [Django](#).

Página do projeto: <http://colab.interlegis.gov.br/wiki/ProjetoSigi>

## Características

Lista das principais características do SIGI:

- Serviço web cliente/servidor, podendo ser disponibilizado tanto na internet quanto na intranet;
- Multi-plataforma;
- Baseado na interface de administração nativa do Django;
- Gerencia convênios, equipamentos e inventários, serviços prestados e composição de Mesas Diretoras das Casas Legislativas;
- Autenticação no sistema baseada em usuários e grupos, com perfis diferentes;
- Emissão de relatórios.

## Aplicações

Lista das aplicações Django desenvolvidas para o SIGI:

**sigi.apps.casas:** Gerência de Casas Legislativas.

**sigi.apps.contatos:** Gerência de Contatos do Interlegis com Casas Legislativas, fornecedores de equipamentos, serviços e etc.

**sigi.apps.convenios:** Convênios do Interlegis com as Casas Legislativas.

**sigi.apps.inventario:** Inventário de equipamentos disponibilizados pelo Interlegis para as Casas Legislativas.

**sigi.apps.mesas:** Composição das Mesas Diretoras das Casas Legislativas.

**sigi.apps.parlamentares:** Gerência de Parlamentares.

**sigi.apps.servicos:** Serviços prestados às Casas Legislativas conveniadas ao Interlegis.



## Requisitos básicos de software

- Python  $\geq 2.4 < 3.0$
- Django 1.0

## Instalação e documentação

Para informações quanto a instalação e demais configurações, veja o arquivo `docs/instalacao.txt`.

**Casos de uso:** `docs/arquivos/casosdeuso.dia`

**Grafo dos Modelos:** `docs/arquivos/models.png`

**Esquema de dados:** `docs/arquivos/esquema.sql`

A documentação do SIGI está disponível no diretório `docs/`.

## Licença de uso

O SIGI é disponibilizado como [software livre](#), isto significa que você pode redistribuí-lo e/ou modifica-lo dentro dos termos da Licença Pública Geral GNU como publicada pela Fundação do Software Livre (FSF); na versão 3 da Licença, ou (na sua opinião) em qualquer versão mais recente.

Veja o arquivo **LEIA-ME** para maiores informações a respeito das condições de cópia e redistribuição.

## 4 Anexo - Arquivo instalacao.txt

Este documento descreve como efetuar a instalação do Sistema de Informações Gerenciais do Interlegis (SIGI). Os passos são baseados nas distribuições GNU/Linux Debian e Ubuntu, porém podem ser aplicados sem dificuldades em outras distribuições, com um pouco de adaptação.

### Preparação do ambiente para a instalação

Há dois tipos de instalação do SIGI, uma para desenvolvimento e outra para colocar em produção.

Para ambas instalações, certifique-se que os seguintes softwares estejam instalados em seu sistema:

- Python  $\geq 2.4 < 3.0$
- Django 1.0



O Python pode ser instalado à partir do pacote `python` com uma ferramenta de instalação de pacotes como o `apt-get` ou `aptitude`.

Se sua distribuição não possui o pacote `python-django`, para o Django na versão 1.0, será necessário obter e configurar o mesmo manualmente. A próxima seção traz mais detalhes para esta tarefa.

## Instalação do Django

O Django 1.0 pode ser obtido através da [página de download](#) do [Django Project](#) via *tarball* (`tar.gz`) ou via Subversion (revisão 8961).

## Obtendo e instalando o Django

Primeiramente baixe o [tarball do Django 1.0](#) ou dê *checkout* na tag 1.0 disponível no repositório do Subversion do projeto:

```
svn checkout http://code.djangoproject.com/svn/django/tags/releases/1.0/
```

Após isso será necessário colocar o pacote Python do `django` (diretório `django` disponível dentro do diretório/*tarball* baixado) no *path* do Python.

Para saber quais são os diretórios que estão no *path* do Python, execute em linha de comando:

```
python -c 'import sys; print sys.path'
```

Você tem opção de mover o pacote `django` para algum diretório coberto pelo *path* ou adicionar um outro local no *path* utilizando a variável de ambiente `PYTHONPATH` do usuário que rodará o sistema. O formato do `PYTHONPATH` é o mesmo do `PATH` do sistema. Exemplo:

```
PYTHONPATH=/path/to/django:$PYTHONPATH
```

De maneira simplificada, a instalação do Django é como a instalação de qualquer outro pacote Python.

## Preparando o ambiente para desenvolvimento

Se você irá desenvolver pro SIGI, é necessário possuir os seguintes pacotes instalados em sua máquina:

- `sqlite3`, para SQLite 3;
- `python-pysqlite2`, interface Python para SQLite 3.



## Preparando o ambiente para produção

Para disponibilizar o SIGI em um ambiente para produção você necessitará do MySQL Server ou PostgreSQL, ou qualquer outro Sistema Gerenciador de Banco de Dados (SGBD) compatível com Django 1.0, e de suas seguintes interfaces para Python, como os pacotes `python-mysqldb` e `python-psycopg2` (respectivamente para MySQL e PostgreSQL).

Será necessário também possuir o Apache ( $\geq 2.2$ ) instalado para disponibilizar o sistema via HTTP, de forma cliente/servidor, ou outro servidor web compatível com Python e WSGI (ou FastCGI).

Opcionalmente, você pode utilizar o servidor web Lighttpd para servir os arquivos estáticos do sistema.

Você encontrará configurações para Apache com WSGI neste documento, juntamente com instruções para configurar o SIGI com o banco de dados escolhido.

## Instalação e configuração do SIGI

O SIGI está disponível no [Colab](#), um portal colaborativo para a gerência dos projetos de software do [Interlegis](#).

A página do projeto SIGI no Colab pode ser acessada através do link <http://colab.interlegis.gov.br/wiki/ProjetoSigi>.

## Obtendo e instalando o SIGI

Para baixar o SIGI, é necessário ter o Subversion instalado em sua máquina (pacote `subversion`). Para *checkout* do sistema via Subversion, execute o comando abaixo:

```
svn checkout http://repositorio.interlegis.gov.br/SIGI/trunk/ /path/to/SIGI
```

Substitua `/path/to/SIGI` para o local onde deseja instalar o SIGI. (Iremos considerar o diretório SIGI como a raiz do projeto durante o restante deste documento.)

## Configuração do SIGI

Dentro da raiz do projeto, encontra-se um diretório de nome `sigi` (pacote Python do projeto). Sua configuração padrão, se encontra no arquivo `settings.py`.

Para alterar as configurações do projeto, é recomendado que copie o arquivo `local_settings.template` para `local_settings.py` e altere os parâmetros neste arquivo.





## Instalação em ambiente para desenvolvimento

As configurações padrões do projeto já estão direcionadas para um ambiente de desenvolvimento e não exige demais configurações.

(Os comandos abaixo deverão ser executados dentro do diretório `sigi`, onde se encontra o arquivo `manage.py`.)

## Base de dados em ambiente para desenvolvimento

Antes de executar o projeto, é necessário preencher o banco de dados com suas tabelas e valores padrão.

Considerando que possua o SQLite instalado, basta executar o comando abaixo:

```
python manage.py syncdb
```

## Execução do projeto em ambiente para desenvolvimento

Para rodar o SIGI, execute em linha de comando:

```
python manage.py runserver
```

O projeto poderá ser acessado através do endereço <http://127.0.0.1:8000/>.

## Instalação em ambiente para produção

Para instalar o SIGI em ambiente para produção, devemos configurar um Sistema Gerenciador de Banco de Dados (SGBD), o qual armazenará os dados do sistema, e configurar um servidor HTTP, como o Apache, para disponibilizar o sistema na web.

## Configuração do SGBD

Modifique em seu `local_settings.py` as variáveis `DATABASE_ENGINE`, `DATABASE_NAME`, `DATABASE_USER`, `DATABASE_PASSWORD`, `DATABASE_HOST` e `DATABASE_PORT` de acordo com o banco de dados disponibilizado para o SIGI.

Em `DATABASE_ENGINE`, suas opções são `postgresql_psycopg2`, `postgresql`, `mysql`, `sqlite3` ou `ado_mssql`.

Após isso, é necessário preencher o banco de dados com suas tabelas e valores padrão. Para tal, execute em linha de comando:

```
python manage.py syncdb
```

## Configuração do servidor web Apache com WSGI

Exemplos de configurações, de um `VirtualHost`, estão disponíveis no diretório `etc/apache/` dentro do diretório raiz do projeto (`SIGI/`).