



# Documentação de Implantação

SISCOF - Sistema de controle de frequência

Autor: Katharine Schaeffer Fertig  
Revisor: Prof. Emerson Mello

São José, 22 de Junho de 2017

# 1 | Ambiente de Desenvolvimento

Para este projeto utilizou-se das ferramentas e softwares MySQL Workbench, Oracle Developer, editores de texto gedit e notepad++. Os ambientes de desenvolvimento foram os sistemas operacionais Windows 8.1 e Ubuntu 14.04 e 16 para programação da aplicação em linguagem PHP, configuração de acesso aos bancos (MySQL e Oracle), testes de geração de PDF e execução de Crontab, documentação e diagramação. Para a configuração de acesso aos bancos também utilizou-se duas abordagens: configuração de PHP, drivers e servidor Apache por XAMP em Windows e manualmente em Linux. Utilizou-se quatro Notebooks principais para testes e integração do sistema: três com sistema operacional linux (dois com arquitetura 32bits e uma com 64bits); e um com sistema operacional Windows 8.1 (arquitetura 64bits). Para as diagramações utilizou-se as ferramentas MySQLWorkbench, DIA e plataforma online draw.io. Além da linguagem principal de programação do sistema (PHP) utilizou-se a linguagem python para programação da funcionalidade crontab (em execução em linux) e demais pequenos detalhes em linguagem HTML5, CSS e framework Bootstrap, assim como APIs para PHP como FPDF.

## 2 | Implantação

Este capítulo tem como objetivo apresentar informações relevantes para a implantação e funcionamento do sistema em produção.

### 2.1 Configuração do servidor

Nesta seção será descrito a configuração do Apache e PHP para executar a aplicação. Restrições do sistema: Ubuntu 14.04.5 ou 14.04 LTS.

```
# sudo apt-get update
# sudo apt-get upgrade
```

#### 2.1.1 Instalação do Apache

O apache a ser instalado deve ser a versão 2.4.

```
# sudo apt-get install apache2
# apache2 -v
Server version: Apache/2.4.7 (Ubuntu)
Server built:   May  9 2017 16:13:38
```

#### 2.1.2 Instalação do PHP

O PHP a ser instalado deve ser a versão 5.5 (ou no máximo 5.6).

```
# sudo apt-get install php5
# php -v
PHP 5.5.9-1ubuntu4.21 (cli) (built: Feb  9 2017 20:40:05)
Copyright (c) 1997-2014 The PHP Group
Zend Engine v2.5.0, Copyright (c) 1998-2014 Zend Technologies
with Zend OPcache v7.0.3, Copyright (c) 1999-2014, by Zend Technologies
```

Para testar se tudo foi instalado corretamente, crie o arquivo index.php da seguinte forma:

```
# sudo vi /var/www/index.php
```

Copie o conteúdo abaixo para o novo arquivo e o salve.

```
<?php
phpinfo();
?>
```

Para que o serviço rode corretamente junto com o PHP5, reinicie o Apache:

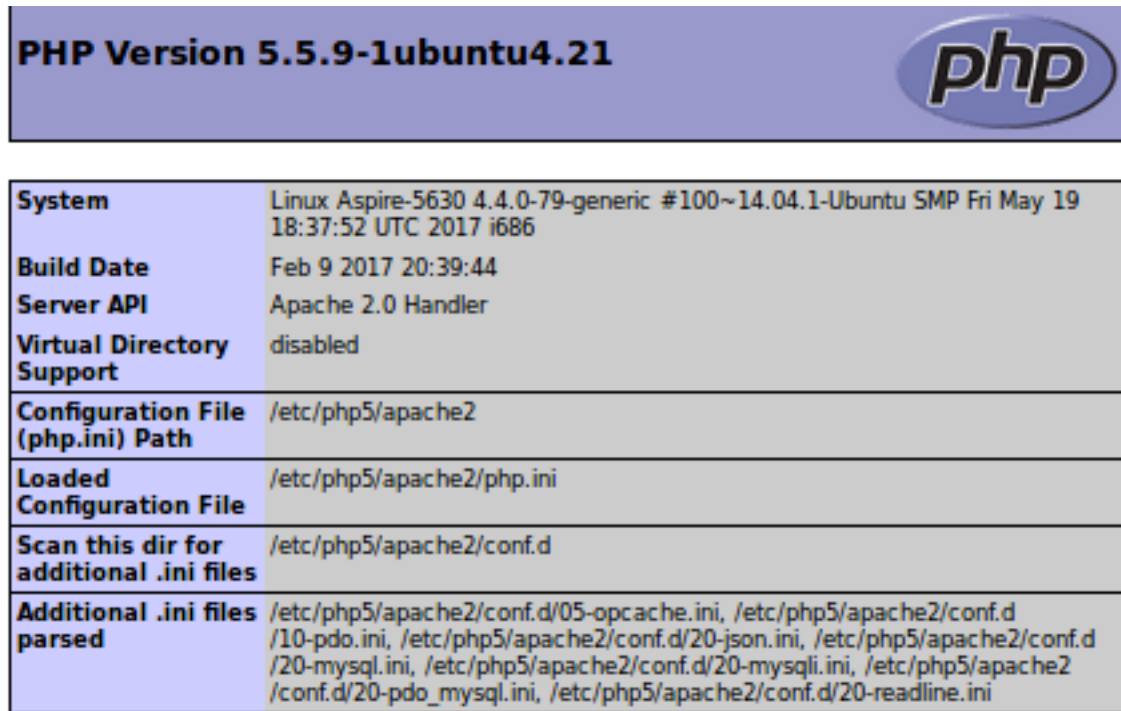
```
# sudo /etc/init.d/apache2 restart
```

Se o seguinte erro aparecer: AH00558 siga o passo a passo desta fonte.

Para a correta visualização, também é preciso alterar o acesso da pasta /www.

```
# sudo chmod -R 777 /var/www/
```

Agora, acesse o endereço <http://localhost/index.php>. Se as informações do PHP aparecerem segunda a figura 2.1, tudo está instalado corretamente.



The image shows a screenshot of a web browser displaying the PHP version banner and a table of configuration details. The banner at the top is purple with the text 'PHP Version 5.5.9-1ubuntu4.21' and the PHP logo. Below it is a table with configuration details.

<b>System</b>	Linux Aspire-5630 4.4.0-79-generic #100~14.04.1-Ubuntu SMP Fri May 19 18:37:52 UTC 2017 i686
<b>Build Date</b>	Feb 9 2017 20:39:44
<b>Server API</b>	Apache 2.0 Handler
<b>Virtual Directory Support</b>	disabled
<b>Configuration File (php.ini) Path</b>	/etc/php5/apache2
<b>Loaded Configuration File</b>	/etc/php5/apache2/php.ini
<b>Scan this dir for additional .ini files</b>	/etc/php5/apache2/conf.d
<b>Additional .ini files parsed</b>	/etc/php5/apache2/conf.d/05-opcache.ini, /etc/php5/apache2/conf.d/10-pdo.ini, /etc/php5/apache2/conf.d/20-json.ini, /etc/php5/apache2/conf.d/20-mysql.ini, /etc/php5/apache2/conf.d/20-mysqli.ini, /etc/php5/apache2/conf.d/20-pdo_mysql.ini, /etc/php5/apache2/conf.d/20-readline.ini

Figura 2.1: Visualização da tabela PHP confirmando sua correta configuração.

Se o erro 404 Not found aparecer, será preciso realizar as seguintes configurações.

Acessar o diretório /etc/apache2/sites-available/: 

```
# cd /etc/apache2/sites-available/
```

Para os dois arquivos .conf existentes neste diretório, edite todos os campos com o seguinte conteúdo “/var/www/html” para somente “/var/www”. Reinicie novamente o Apache e faça o teste para acessar mais uma vez o index.php.

### 2.1.3 Instalação do MySQL

O mysql a ser instalado deve ser a versão 5.5.

```
# sudo apt-get install mysql-server-5.5 php5-mysql
```

Insira uma senha para acesso posterior ao BD mysql. O padrão é mysqladmin.

```
# sudo mysqladmin -u root seu_password
```

**OBS.:** Se não for possível acessar o mysql, veja mais detalhes nesta fonte.

```
# mysql --version (versão utilizada no projeto)
```

```
mysql Ver 14.14 Distrib 5.5.55, for debian-linux-gnu (i686) using readline 6.3
```

Reinicie novamente o Apache e faça o teste para acessar mais uma vez o index.php. Se as informações da tabela do mysql ou mysqli aparecerem como a seguir nas figuras 2.2 e 2.3 respectivamente, tudo está instalado corretamente.

### mysql

MySQL Support	enabled
Active Persistent Links	0
Active Links	0
Client API version	5.5.55
MYSQL_MODULE_TYPE	external
MYSQL_SOCKET	/var/run/mysqld/mysqld.sock
MYSQL_INCLUDE	-I/usr/include/mysql
MYSQL_LIBS	-L/usr/lib/i386-linux-gnu -lmysqlclient_r

Figura 2.2: Visualização da tabela *mysql* confirmando sua correta configuração.

### mysqli

Mysqli Support	enabled
Client API library version	5.5.55
Active Persistent Links	0
Inactive Persistent Links	0
Active Links	0
Client API header version	5.5.54
MYSQLI_SOCKET	/var/run/mysqld/mysqld.sock

Figura 2.3: Visualização da tabela *mysqli* confirmando sua correta configuração.

#### 2.1.4 Instalação do Oracle Instant Client

Faça o download dos pacotes Basic e SDK do Instant Client do tipo ".rpm" na página da Oracle atentando para a versão do seu sistema (32bits = x86 e 64bits = x86-64). Link para download se encontra aqui aqui.

No terminal, instale o Alien: `# sudo apt-get install alien`.

Na pasta onde os arquivos .rpm foram baixados, execute o seguinte comando para ambos os pacotes: `# sudo alien nome_pacote.rpm`

A partir de então você terá os pacotes .deb com mesmo nome gerados. Execute para ambos os pacotes .deb o seguinte comando: `# sudo dpkg -i nome_pacote.deb`

Após a execução, o primeiro arquivo RPM (basic) salva as bibliotecas Oracle no diretório `/usr/lib/oracle/12.2/client64/lib`. Já o segundo (devel) cria headers no diretório `/usr/include/oracle/12.2/client64`. Se o seu sistema for 32bits, o diretório muda respectivamente para `/usr/lib/oracle/12.2/client/lib` e `/usr/include/oracle/12.2/client`.

#### 2.1.5 Instalação do PHP OCI8

O OCI8 se trata do driver para o php que permite acesso à bancos de dados oracle, neste caso, o sistema já existente da catraca.

Primeiramente baixe a versão OCI2.0.8 da página PECL (formato .tgz para linux). Instale então o php5-dev. # sudo apt-get install php5-dev

Mova o pacote compactado para um diretório permanente, para futuras procuras pelo pacote e para a garantia de que este não será deletado ou perdido. Descompacte o pacote .tgz do oci baixado:

```
# tar -xzf oci8-2.0.8.tgz
# cd oci8-2.0.8
# sudo phpize5
Este irá gerar:
Configuring for:
PHP Api Version:      20121113
Zend Module Api No:   20121212
Zend Extension Api No: 220121212
```

Ainda no diretório oci-2.0.8 torne o script shell configure.sh executável e o execute como a seguir: # sudo chmod +x configure.sh # sudo ./configure --with-oci8=instantclient,<caminho>

Em <caminho> insira o diretório respectivo à sua arquitetura: /usr/lib/oracle/12.2/client64/lib para x64 (64 bits) ou /usr/lib/oracle/12.2/client/lib para x86 (32 bits)

```
make install
Dentre várias mensagens, irá gerar:
Libraries have been installed in: /home/maedel/Downloads/oci8-2.0.8/modules
Installing shared extensions:     /usr/lib/php5/20121212+lfs/
```

Habilite a extensão PHP OCI8 editando o arquivo /etc/php5/apache2/php.ini e adicionando: extension=oci8.so

Também confirme que o campo extension\_dir aponte para o diretório que o oci8.so foi copiado durante a compilação. (durante o make install) Em nosso caso, temos: extension\_dir = "/home/maedel/Downloads/oci8-2.0.8/modules"

### **Crie um ambiente ORACLE\_HOME:**

Crie um arquivo: /etc/ld.so.conf.d/oracle.conf e na primeira linha do script escreva:

```
/usr/lib/oracle/12.2/client64/lib (Para instalação 64bits)\
/usr/lib/oracle/12.2/client/lib (Para instalação 32bits)
```

```
export ORACLE_HOME=/usr/lib/oracle/12.2/client (ou /client64 para 64bits)
export LD_LIBRARY_PATH=$ORACLE_HOME/lib
sudo ldconfig
```

**OBS.:** É muito importante setar TODAS as variáveis do ambiente Oracle antes de reiniciar o Apache para que o processamento do OCI8 seja corretamente inicializado. Setar os campos e variáveis em scripts PHP podem gerar erros óbvios e não óbvios.

Reinicie novamente o Apache e faça o teste para acessar mais uma vez o index.php. Se as informações da tabela do OCI8 aparecerem como na figura 2.4, tudo está instalado corretamente.

## oci8

<b>OCI8 Support</b>	enabled
<b>OCI8 DTrace Support</b>	disabled
<b>OCI8 Version</b>	2.0.8
<b>Revision</b>	\$Id: f04114d4d67cffe4cdc2ed3b7f0229c2caa5016 \$
<b>Oracle Run-time Client Library Version</b>	12.2.0.1.0
<b>Oracle Compile-time Instant Client Version</b>	12.2

Directive	Local Value	Master Value
<b>oci8.connection_class</b>	<i>no value</i>	<i>no value</i>
<b>oci8.default_prefetch</b>	100	100
<b>oci8.events</b>	Off	Off
<b>oci8.max_persistent</b>	-1	-1
<b>oci8.old_oci_close_semantics</b>	Off	Off
<b>oci8.persistent_timeout</b>	-1	-1
<b>oci8.ping_interval</b>	60	60
<b>oci8.privileged_connect</b>	Off	Off
<b>oci8.statement_cache_size</b>	20	20

Statistics	
<b>Active Persistent Connections</b>	0
<b>Active Connections</b>	0

Figura 2.4: Visualização da tabela OCI confirmando sua correta configuração.

## 2.2 Configuração do Crontab

### 2.2.1 Instalação do Python

Pré-requisitos: sistema operacional Ubuntu 14.04.5, servidor apache2, php5, mysql, oracle instant client basic, oracle SDK.

Instalar o python pip e o mysql-connector-python.

```
apt-get install python-pip
sudo apt-get install python-dev
pip install mysql-connector-python
```

Faça o download do cx\_Oracle-5.1.3, tar ball, do pypi e extraia.

```
tar -xzf cx\Oracle-5.1.3.tar.gz
cd cx_Oracle-5.1.3
```

Modifique as seguintes linhas do arquivo cx\_Oracle-5.1.3/setup.py, da linha 123 (as linhas em vermelho):

```
# try to determine the Oracle home
userOracleHome = os.environ.get("ORACLE_HOME")

userOracleHome = "/usr/lib/oracle/12.2/client" (ou /client64 para instalações 64 bits)

if userOracleHome is not None:
    if not CheckOracleHome(userOracleHome):
        messageFormat = "Oracle home (%s) does not refer to an " \
            "9i, 10g, 11g or 12c installation."
        raise DistutilsSetupError(messageFormat % userOracleHome)
    else:
        for path in os.environ["PATH"].split(os.pathsep):
            if CheckOracleHome(path):
                break
        if oracleHome is None:
            #~ raise DistutilsSetupError("cannot locate an Oracle software" \
            #~ "installation")

oracleHome = "/usr/lib/oracle/12.2/client" (ou /client64 para instalações 64 bits)
```

Faça um Build e instale o cx\_Oracle 5.1.3: `python setup.py build`

Irá aparecer:

```
running build
running build_ext
```

```
# sudo python setup.py install\verb
```

Ao final irá aparecer:

```
Installed /usr/local/lib/python2.7/dist-packages/cx\Oracle-5.1.3-py2.7-linux-i686.egg
Processing dependencies for cx-Oracle==5.1.3
Finished processing dependencies for cx-Oracle==5.1.3
```



## 2.3 Configuração do Banco de Dados

### 2.3.1 Importação da Base de Dados (MySQL)

Primeiramente acesse o mysql no servidor e crie uma nova base de dados:

```
mysql -u root -p ;
Enter password: mysqladmin

mysql> CREATE DATABASE siscof;
Query OK, 1 row affected (0.05 sec)
```

```
mysql> SHOW DATABASES;
+-----+
| Database                |
+-----+
| information_schema      |
| mysql                   |
| performance_schema     |
| siscof                   |
+-----+
4 rows in set (0.10 sec)
```

Baixe para o diretório principal do servidor o banco de dados modelado encontrado no seguinte url.

Fora do mysql, no terminal, importe o banco de dados para o criado no servidor:

```
mysql -u root -p siscof < ControleDeFrequencia.sql
```

Entre no mysql novamente e selecione a base de dados siscof para ser usada.

```
mysql> use siscof;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
```

```
Database changed
mysql> show tables;
+-----+
| Tables_in_siscof        |
+-----+
| Alertas_Faltas          |
| Alertas_Horarios        |
| Ocorrencia_Faltas       |
| Ocorrencia_Horario      |
| Registro_Faltas         |
| Registro_Horario        |
| Usuarios_Permitidos     |
+-----+
7 rows in set (0.00 sec)
```

## 2.4 Configuração da Aplicação

Fazer um clone do diretório git siscof para o diretório /var/www.

Vá no diretório `/var/www/sisconf/application/app_beta3/sisconf` e edite os seguintes campos do arquivo `banco_mysql.php`:

```
//variaveis para conexao com o banco MySQL local
if (!defined("host")) define ("host","localhost");
if (!defined("user")) define ("user","root");
if (!defined("pw")) define ("pw","mysqladmin");
if (!defined("db")) define ("db","sisconf");
```

Habilite novamente acesso ao diretório `/var/www` e reinicie o apache:

```
sudo chmod -R 777 /var/www/
sudo /etc/init.d/apache2 restart
```

Acesse o endereço. Se a aplicação surgir como na figura 2.5, tudo está configurado corretamente.



Figura 2.5: Visualização da geração correta da página de aplicação.

### 2.4.1 Importação do módulo Crontab

Baixar do url todos os arquivos da pasta Crontab e salvar na raiz do diretório `/var/www/sisconf`.

### 2.4.2 Importação do módulo Gerar Relatórios

Pré-requisitos: sistema operacional Ubuntu 14.04.5, servidor apache2, php5, mysql, e acesso remoto à um banco de dados Oracle.

Baixar do url todos os arquivos da pasta `relatorio_pdf` e salvar na raiz do diretório `/var/www/sisconf`.