INSTALLATION DIRECTIONS

RosarioSIS Student Information System

RosarioSIS is a web based application which relies on other facilities such as a web server, PHP server-side scripting, and a PostgreSQL database server.

For RosarioSIS to work you must first have your web server working, PostgreSQL working, PHP working (including the `pgsql`, `gettext`, `mbstring`, `gd`, `curl`, `xmlrpc`, `xml` & `zip` extensions). Setting these up varies a lot with operating system so it is well beyond the scope of this brief install document.

RosarioSIS was tested on:

- Windows 10 x86 with Apache 2.4.16, Postgres 9.3.6, and PHP 5.4.45
- Ubuntu 14.04 with Apache 2.4.18, Postgres 9.3.10, and PHP 5.5.9
- Debian Jessie with Apache 2.4.16, Postgres 9.4, and PHP 5.6.13
- Debian Stretch with Apache 2.4.25, Postgres 9.6, and PHP 7.0.14
- Ubuntu 16.04 with Apache 2.4.18, Postgres 9.5, and PHP 7.3.4
- Debian Buster with Apache 2.4.38, Postgres 11.5, and PHP 7.4.6
- CentOS 8.2 with Apache 2.4.37, Postgres 9.6.10, and PHP 7.2.24
- Shared hosting with cPanel, nginx, Postgres 8.4, and PHP 5.6.27
- through Mozilla Firefox and Google Chrome
- through BrowserStack for cross-browser compatibility (not compatible with Internet Explorer 9 or lower)

Minimum requirements: **PHP 5.4.45** & **PostgreSQL 8.4**

Installation directions for:

- **Windows**
- **cPanel**
- **Docker**

Installing the package

Unzip RosarioSIS, or clone the repository using git to a directory that is accessible to your web browser. Edit the `config.inc.sample.php` file to set the configuration variables and rename it to `config.inc.php`.

- `$DatabaseServer` Host name or IP for the database server.
- `$DatabaseUsername` Username used for authenticating the database.
- `$DatabasePassword` Password used for authenticating the database.
- `$DatabaseName` Database name.
- `$DatabasePort` Port number for accessing the database server.
- `$pg_dumpPath` Full path to the postgres database dump utility, pg_dump.
- `$wkhtmltopdfPath` Full path to the PDF generation utility, wkhtmltopdf.
- `$DefaultSyear` Default school year. Only change after running the *Rollover* program.
- `$RosarioNotifyAddress` Email address to receive notifications (new administrator, new student / user, new registration).
- `$RosarioLocales` Comma separated list of locale codes. Check the `locale/` folder for available codes.

Optional variables

- `$RosarioPath` Full path to RosarioSIS installation.
- `$wkhtmltopdfAssetsPath` Path to the `assets/` director for wkhtmltopdf. Possibly different than how the web browser finds it. Empty string means no translation.
- `$StudentPicturesPath` Path to student pictures.
- `$UserPicturesPath` Path to user pictures.
- `$PortalNotesFilesPath` Path to portal notes attached files.
- `$AssignmentsFilesPath` Path to student assignments files.
- `$FS_IconsPath` Path to food service icons.
- `$FileUploadsPath` Path to file uploads.

- `$LocalePath` Path to language packs. Restart Apache after changes to this directory.
- `$PNGQuantPath` Path to PNGQuant (PNG images compression).
- `$RosarioErrorsAddress` Email address to receive errors (PHP fatal, database, hacking).
- `$Timezone` Default time zone used by date/time functions. List of Supported Timezones.
- `$ETagCache` Set to `false` to deactivate the ETag cache and disable "private" session cache. See Sessions and security.
- `define( 'ROSARIO_DEBUG', true );` Debug mode activated.
- `define( 'ROSARIO_DISABLE_ADDON_UPLOAD', true );` Disable add-ons (modules & plugins) upload.

Database

Now, you're ready to setup the RosarioSIS database. If you have access to the command prompt for your server, follow these instructions.

1. Open a terminal window.
2. Login to PostgreSQL as the postgres user:

   ```
   server$ sudo -u postgres psql
   ```

3. Create the rosariosis user:

   ```
   postgres=# CREATE USER rosariosis_user WITH PASSWORD 'rosariosis_user_password';
   ```

4. Create the rosariosis database:

   ```
   postgres=# CREATE DATABASE rosariosis_db WITH ENCODING 'UTF8' OWNER rosariosis_user;
   ```

5. Logout of PostgreSQL:

   ```
   postgres=# \q
   ```

Also, the `pg_hba.conf` file may have to be altered to enable password (`md5`) peer authentication:

```
# "local" is for Unix domain socket connections only
local   all             all                                     md5
```

To install the database, point your browser to: `http://yourdomain.com/INSTALL_LOCATION/InstallDatabase.php`

That's it!... now, point your browser to: `http://yourdomain.com/INSTALL_LOCATION/index.php`

and login as 'admin' password 'admin'. With this login, you can create new users, and change and delete the three template users.

Problems

To help you spot installation problems, point your browser to: `http://yourdomain.com/INSTALL_LOCATION/diagnostic.php`

PHP extensions

Install instructions for Ubuntu 16.04:

```
server$ sudo apt-get install php-pgsql php-gettext php-mbstring php-gd php-curl php-xmlrpc php-xml php-zip
```

php.ini

Recommended PHP configuration settings. Edit the `php.ini` file as follows:

```
max_execution_time = 240
max_input_vars = 4000
memory_limit = 768M
```

Restart PHP and Apache.

Other languages

Install instructions for Ubuntu 16.04. Install the *Spanish* language:

```
server$ sudo apt-get install language-pack-es
```

Then restart the server.

wkhtmltopdf

Install instructions for Ubuntu 16.04 (Xenial):

```
server$ wget https://downloads.wkhtmltopdf.org/0.12/0.12.5/wkhtmltox_0.12.5-1.xenial_amd64.deb
server$ sudo dpkg -i wkhtmltox_0.12.5-1.xenial_amd64.deb
```

Set path in the `config.inc.php` file:

```
$wkhtmltopdfPath = '/usr/local/bin/wkhtmltopdf';
```

Send email

Install instructions for Ubuntu 16.04. Activate the PHP `mail()` function:

```
server$ sudo apt-get install sendmail
```

Additional configuration

Quick Setup Guide