

# 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 or MySQL/MariaDB database server.

For RosarioSIS to work you must first have your web server working, PostgreSQL (or MySQL/MariaDB) working, PHP working (including the `pgsql`, `mysqli`, `gettext`, `intl`, `mbstring`, `gd`, `curl`, `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 7.1.18
- macOS Monterey with Apache 2.4.54, Postgres 14.4, and PHP 8.0.21
- Ubuntu 14.04 with Apache 2.4.18, Postgres 9.3.10, MySQL 5.6.33 and PHP 5.5.9
- Ubuntu 20.04 with Apache 2.4.48, Postgres 12.11, and PHP 7.4.22
- Debian Bullseye with Apache 2.4.54, Postgres 13.7, MariaDB 10.5.15 and PHP 8.1.4
- 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)

Minimum requirements: **PHP 5.5.9** & **PostgreSQL 8.4** or **MySQL 5.6**/MariaDB

Installation directions for:

- [Windows](#)
- [Mac](#)
- [cPanel](#)
- [Softaculous](#)
- [Docker](#)
- **Ubuntu** (or any Debian-based Linux distribution), see below

## 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`.

- `$DatabaseType` Type of the database server: either `mysql` or `postgresql`.
- `$DatabaseServer` Host name or IP for the database server.
- `$DatabaseUsername` Username used for authenticating the database.
- `$DatabasePassword` Password used for authenticating the database.
- `$DatabaseName` Database name.
- `$DatabaseDumpPath` Full path to the database dump utility, `pg_dump` (PostgreSQL) or `mysqldump` (MySQL).
- `$wkhtmltopdfPath` Full path to the PDF generation utility, `wkhtmltopdf`.
- `$DefaultYear` 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

- `$DatabasePort` Port number to access the database server. Default is 5432 for PostgreSQL & 3306 for MySQL.
- `$RosarioPath` Full path to RosarioSIS installation.
- `$wkhtmltopdfAssetsPath` Path to the `assets/` directory for `wkhtmltopdf`. Possibly different than how the web browser finds it. Empty string means no translation.
- `$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_POST_MAX_SIZE_LIMIT', 16 * 1024 * 1024 );` Limit `$_POST` array size (default is 16MB). More info [here](#).
- `define( 'ROSARIO_DEBUG', true );` Debug mode activated.
- `define( 'ROSARIO_DISABLE_ADDON_UPLOAD', true );` Disable add-ons (modules & plugins) upload.
- `define( 'ROSARIO_DISABLE_ADDON_DELETE', true );` Disable add-ons (modules & plugins) delete.

## Create database

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

The following instructions are for **PostgreSQL** (for MySQL see below):

1. Login to PostgreSQL as the postgres user:

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

2. Create the rosariosis user:

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

3. Create the rosariosis database:

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

4. 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
```

The following instructions are for **MySQL** (RosarioSIS version 10 or higher):

1. Login to MySQL as the root user:

```
server$ sudo mysql
```

or

```
server$ mysql -u root -p
```

2. Allow function creation:

```
mysql> SET GLOBAL log_bin_trust_function_creators=1;
```

3. Create the rosariosis user:

```
mysql> CREATE USER 'rosariosis_user'@'localhost' IDENTIFIED BY 'rosariosis_user_password';
```

4. Create the rosariosis database:

```
mysql> CREATE DATABASE rosariosis_db CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_520_ci;
mysql> GRANT ALL PRIVILEGES ON rosariosis_db.* TO 'rosariosis_user'@'localhost';
```

5. Logout of MySQL:

```
mysql> \q
```

## Install database

To install the database, point your browser to: [http://yourdomain.com/INSTALL\\_LOCATION/InstallDatabase.php](http://yourdomain.com/INSTALL_LOCATION/InstallDatabase.php)

That's it!... now, point your browser to: [http://yourdomain.com/INSTALL\\_LOCATION/index.php](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 20.04:

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

## php.ini

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

```
; Maximum time in seconds a PHP script is allowed to run
max_execution_time = 240
```

```
; Maximum accepted input variables ($_GET, $_POST)
; 4000 allows submitting lists of up to 1000 elements, each with multiple inputs
max_input_vars = 4000
```

```
; Maximum memory (RAM) allocated to a PHP script
memory_limit = 512M
```

```
; Session timeout: 1 hour
session.gc_maxlifetime = 3600
```

```
; Maximum allowed size for uploaded files
upload_max_filesize = 50M
```

```
; Must be greater than or equal to upload_max_filesize
post_max_size = 51M
```

Restart PHP and Apache.

## Other languages

Install instructions for Ubuntu 20.04. Install the *Spanish* language:

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

Then restart the server.

## wkhtmltopdf

Install instructions for Ubuntu 20.04 (Focal):

```
server$ wget https://github.com/wkhtmltopdf/packaging/releases/download/0.12.6-1/wkhtmltox_0.12.6-1.focal_amd64.deb
server$ sudo apt install ./wkhtmltox_0.12.6-1.focal_amd64.deb
```

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

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

## Send email

Install instructions for Ubuntu 20.04. Activate the PHP `mail()` function:

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

## Additional configuration

[Quick Setup Guide](#)

[Secure RosarioSIS](#)