



# Installation & Setup Manual

Orangescrum is a free, open source, flexible project management web application written using CakePHP.

Just 5 simple steps to start using Orangescrum in your premises. This procedure is well-tested on Windows, Linux and Mac operating system.

## Requirements

- Apache with `mod\_rewrite`
- PHP 5.3 or higher
  - Enable **curl** in php.ini
  - Change the '**post\_max\_size**' and '**upload\_max\_filesize**' to 200Mb in php.ini
- MySQL 4.1 or higher

## Installation

### Minimum Configuration

1. Extract the archive. Upload all the files and folders to your working directory.
  - 1.1. Provide proper **write permission** to “**app/tmp**” and “**app/webroot**” folders and their sub-folders.
  - 1.2. Make sure that, there are three **.htaccess** files. One in the root directory, another in the “app” directory and the last one is in the “app/webroot” directory.
  - 1.3. Make sure that, the **.htaccess** is working in your server.
2. Create a new MySQL database (`utf8\_unicode\_ci` collation)
3. Get the **database.sql** file from the root directory and import that to your database
4. Locate your `app` directory, do the changes on following files
  - 4.1. **app/Config/database.php**
    - 4.1.1. Update the database connection details. (host, login, password and database name)
  - 4.2. **app/Config/constants.php**
    - 4.2.1. Provide your valid **Gmail ID and Password** for **SMTP email sending** Or, you can use Sendgrid or Mandrill.
    - 4.2.2. `FROM_EMAIL_NOTIFY` - All the task created/updated notification email will be sent from this Email ID.
    - 4.2.3. `SUPPORT_EMAIL` - All other Emails and support related Emails will be sent from this Email ID.
    - 4.2.4. `SUB_FOLDER` - Update the folder name, If your application URL is “[http://www.my-orangescrum.com/folder\\_name/](http://www.my-orangescrum.com/folder_name/)” (the value should follow

by a forward slash. e.g. `define('SUB_FOLDER', folder_name/')`.  
Leave it blank If your application URL doesn't have an additional folder path.

5. Now you are all set, run Orangescrum as `http://your-siteurl.com/` from your browser.  
Change the `Configure::write('debug',2)` to  
`Configure::write('debug',0)` in **app/Config/core.php**

You will be asked to provide your Company Name, Email address and a Password to login and start using Orangescrum.

### Other Configurations in app/Config/constants.php

1. Create a Google Project (<https://console.developers.google.com/project>) for Google Login, Invite Google Contacts and Google Drive file attachment.
  - 1.1. Update the CLIENT\_ID, CLIENT\_ID\_NUM, CLIENT\_SECRET, API\_KEY values.
  - 1.2. Set the USE\_GOOGLE value to 1 to use Google Google Login, Invite Google Contacts and Google Drive file attachment
2. Create a Dropbox App (<https://www.dropbox.com/developers>) for Dropbox file file sharing.
  - 2.1. Update the DROPBOX\_KEY value and Set the USE\_DROPBOX value to 1 to use Dropbox file file sharing.
3. Create 2 Buckets in AWS S3 to store all the files in the S3 Bucket.
  - 3.1. Provide the S3 Bucket Names, awsAccessKey, awsSecretKey values.
  - 3.2. Set the USE\_S3 value to "1" to store all file attachment and profile photo in AWS S3 bucket.

Get the **"DropboxandGooglSetup.pdf"** on the root directory of the Orangescrum project folder to know, How to create apps and projects on Dropbox and Google Drive.

### Cron Job Settings (Linux Server)

Update your application url in `define('DOMAIN', "www.my-orangescrum.com/")` in the "app/Config/constants.php".

Below are the 4 Cron Jobs to set on a **Linux server**,

(Assuming your Application is in `"/var/www/html/orangescrum/"`)

1. `0 23 * * * php -q`  
`/var/www/html/orangescrum/app/webroot/cron_dispatcher.php`  
`/cron/email_notification`
  - 1.1. Daily/Weekly/Monthly Task Status update email in the "Email Reports" section of Orangescrum.
2. `*/15 * * * * php -q`  
`/var/www/html/orangescrum/app/webroot/cron_dispatcher.php`  
`/cron/dailyupdate_notifications`
  - 2.1. Daily Update Reports in the "Email Reports" section of Orangescrum

3. 

```
*/15 * * * * php -q
/var/www/html/orangescrum/app/webroot/cron_dispatcher.php
/cron/dailyUpdateMail
```

  - 3.1. Daily Catch Up Email Alert
4. 

```
*/30 * * * * php -q
/var/www/html/orangescrum/app/webroot/cron_dispatcher.php
/cron/weeklyusagedetails
```

  - 4.1. Weekly Usage report of your account

### You can also set the Cron Jobs using the absolute urls.

1. [http://www.your-orangescrum/cron/email\\_notification](http://www.your-orangescrum/cron/email_notification)
2. [http://www.your-orangescrum/cron/dailyupdate\\_notifications](http://www.your-orangescrum/cron/dailyupdate_notifications)
3. <http://www.your-orangescrum/cron/dailyUpdateMai>
4. <http://www.your-orangescrum/cron/weeklyusagedetails>

### Email Reply - Nohup Cron Job setup (Linux Server)

Make sure to do the following changes on the Email server connection details in the **app/webroot/EmailReply.php** file.

1. **\$username** - This will be the `FROM_EMAIL_NOTIFY` Email set on your `app/Config/constants.php`
  - 1.1. All the task created/updated notification email will be sent from `FROM_EMAIL_NOTIFY`. When somebody will reply on that task created/updated notification email, the `FROM_EMAIL_NOTIFY` will get that Email in the inbox.
  - 1.2. **EmailReply.php** page is going to read the emails from `FROM_EMAIL_NOTIFY` and It will post to them as a reply to the respective tasks in Orangescrum.
2. **\$password** - Password of `FROM_EMAIL_NOTIFY`
3. **client** - Change it, if you are not using Gmail

After this setup, you can reply to a task created/updated notification email and that email reply will be posted to Orangescrum under that Task. This will help you to respond to a task while on-the-go from your Mobile.

(Assuming your Application is in `"/var/www/html/orangescrum/"`)

1. Enable **extension=php\_imap** in your `php.ini` file.
2. Create a **orangescrum.sh** file in your server
3. **vi orangescrum.sh** (or, open that file to write the below code)

```
#!/bin/bash
while(true)
do
cd /var/www/html/orangescrum/app/webroot
php -q EmailReply.php 1>&2
sleep 1;
done
```

4. Give the execute permission for orangescrum.sh: **chmod +x orangescrum.sh**
5. Start the Nohup using the command: **nohup sh orangescrum.sh > custom-out.log &**

### Nohup Cron Job setup (Linux Server)

- Make sure that you have installed **Node.js** and **NPM**
- Install **Socket.io** using NPM
- Install **Forever**
- Find how to add a JavaScript file to run Node.js forever.
- Now enable the “NODEJS\_HOST” with your server details on “app/Config/constants.php”

---

Version 1.0  
29th October 2014

## Changelog

- 22 Sept 2014 - Release of Orangescrum Installation Manual
- 24 Sept 2014 - Changes in the Nohup Cron Job setup
- 26 Sept 2014 - Changes in the Cron Job section and added text for Drive and Dropbox API.
- 29 Oct 2014 - Added Node.js setup