

```
# Use the front controller as index file. It serves as a fallback solution when
# every other rewrite/redirect fails (e.g. in an aliased environment without
# mod_rewrite). Additionally, this reduces the matching process for the
# start page (path "/") because otherwise Apache will apply the rewriting rules
# to each configured DirectoryIndex file (e.g. index.php, index.html, index.pl).
DirectoryIndex app.php
```

```
<IfModule mod_rewrite.c>
```

```
    RewriteEngine On
```

```
    # Determine the RewriteBase automatically and set it as environment variable.
    # If you are using Apache aliases to do mass virtual hosting or installed the
    # project in a subdirectory, the base path will be prepended to allow proper
    # resolution of the app.php file and to redirect to the correct URI. It will
    # work in environments without path prefix as well, providing a safe, one-size
    # fits all solution. But as you do not need it in this case, you can comment
    # the following 2 lines to eliminate the overhead.
```

```
    RewriteCond %{REQUEST_URI}::$1 ^(/.+)/(.*):;\2$
```

```
    RewriteRule ^(.*) - [E=BASE:%1]
```

```
    # Redirect to URI without front controller to prevent duplicate content
    # (with and without `/app.php`). Only do this redirect on the initial
    # rewrite by Apache and not on subsequent cycles. Otherwise we would get an
    # endless redirect loop (request -> rewrite to front controller ->
    # redirect -> request -> ...).
    # So in case you get a "too many redirects" error or you always get redirected
    # to the startpage because your Apache does not expose the REDIRECT_STATUS
    # environment variable, you have 2 choices:
```

```
    # - disable this feature by commenting the following 2 lines or
    # - use Apache >= 2.3.9 and replace all L flags by END flags and remove the
    #   following RewriteCond (best solution)
```

```
    RewriteCond %{ENV:REDIRECT_STATUS} ^$
```

```
    RewriteRule ^app\.php(/(.*)|)$ %{ENV:BASE}/$2 [R=301,L]
```

```
    # If the requested filename exists, simply serve it.
    # We only want to let Apache serve files and not directories.
```

```
    RewriteCond %{REQUEST_FILENAME} -f
```

```
    RewriteRule .? - [L]
```

```
    # Rewrite all other queries to the front controller.
```

```
    RewriteRule !- [E=BASE:%1] [L]
```

```
    RewriteRule .? %ENV{HOME}/app.php [L]
</IfModule>

<IfModule !mod_rewrite.c>
    <IfModule mod_alias.c>
        # When mod_rewrite is not available, we instruct a temporary redirect of
        # the startpage to the front controller explicitly so that the website
        # and the generated links can still be used.
        RedirectMatch 302 ^/$ /app.php/
        # RedirectTemp cannot be used instead
    </IfModule>
</IfModule>

Order Allow,Deny
Allow from all

<IfModule mod_php5.c>
    php_flag zend.zel_compatibility_mode off
    php_flag short_open_tag off
    php_flag register_globals off
    php_flag display_errors on
    php_flag magic_quotes_gpc off
    php_flag track_vars on
    php_flag session.auto_start off
    #php_value session.gc_maxlifetime 3600
    #php_value session.cookie_lifetime 1200 # doesn't have any influence, set it in the
    app/config/config.yml: framework.session.cookie_lifetime
    php_value date.timezone "Europe/Rome"
    php_value memory_limit "254M"
    php_value post_max_size "201M"
    php_value upload_max_filesize "100M"
    php_value max_execution_time "300"
    # The same precision on MySQL:
    php_value precision "16"
    php_value serialize_precision "17"
</IfModule>
```