

Configuration du fichier config

Ouvrir le fichier de configuration dans le dossier **Archimage2/config/**

Il se nomme config_template.txt il faut le compléter puis le renommer en config.php

Il est important de correctement saisir la partie URL et SQL afin que le site puisse se lancer.

```
<?php declare(strict_types=1);  
  
#####  
//  
//          FRAMEWORK OPTIONS  
//  
#####  
// APP NAME  
ParamManager::getInstance()->app_name="Archimage";  
// SERVER NAME  
ParamManager::getInstance()->server_name="Archimage";  
// SITE TITLE  
ParamManager::getInstance()->site_title="ARCHIMAGE";  
// DEBUG MODE  
ParamManager::getInstance()->debug=false;  
// SESSION NAME  
ParamManager::getInstance()->site_session_name="ARCHIMAGE_SESSION";  
// SERVER URL !!  
ParamManager::getInstance()->server_url="https://archimage.com/";  
ParamManager::getInstance()->site_root="https://archimage.com/";  
ParamManager::getInstance()->site_action="https://archimage.com/action.php";  
  
#####  
#####  
  
// SQL DATABASE  
ParamManager::getInstance()->sql_engine=Sql::$MYSQL;  
ParamManager::getInstance()->sql_host="localhost";  
ParamManager::getInstance()->sql_user="archimage-user";
```

```
ParamManager::getInstance()->sql_database="archimage-database";
ParamManager::getInstance()->sql_pass="password";

#####
// EMAIL PARAMS :
// EMAIL FOR ERROR
ParamManager::getInstance()->error_email="error@email.com";
// Default Public EMAIL
ParamManager::getInstance()->public_email="public@email.com";
// EMAIL for contact form
ParamManager::getInstance()->add("CONTACT_EMAIL","contact@email.com");
// EMAIL SERVER PARAM
ParamManager::getInstance()->email_transport="smtp"; // smtp gmail null
ParamManager::getInstance()->email_mail="mail@email.com";
ParamManager::getInstance()->email_username="username";
ParamManager::getInstance()->email_password="password";
// STORING EMAIL BY IMAP ~ IF EMPTY IS NOT STORED
ParamManager::getInstance()->email_server="{email.com:993/imap/ssl/novalidate-cert}"; // using
IMAP for storing
// SENDING PARAM ~ IF EMPTY NOTHING IS SEND
ParamManager::getInstance()->email_port="587"; // 25 587 ...
ParamManager::getInstance()->email_encryption="ssl"; // tls, ssl, or null
ParamManager::getInstance()->email_auth_mode="login"; // plain, login, cram-md5, ntlm or null.
ParamManager::getInstance()->email_verify_peer="false"; // if self sign certificate set false
ParamManager::getInstance()->email_smtp_server="email.com";

#####
// TMP DIR FOR :
// UPLOADS
ParamManager::getInstance()->add("TMP_UPLOAD_DIR","/home/user/dir/TMP_UPLOAD/");
// LOGS
ParamManager::getInstance()->log_directory="/home/user/dir/logs/";
// MAILES
ParamManager::getInstance()->log_dir_mail="/home/user/dir/mailes/";

// DIRECTORY FOR THE STORAGE OF THE FILES !! SPACE NEEDED
```

```

ParamManager::getInstance()->add("REPOSITORY","/home/user/dir/REPOSITORY");

// URL OF THE IIIF ENGINE
ParamManager::getInstance()->add("IIIF_ACCESS","0"); // 0 => everyone (by default) // 1 =>
only connected // 2 => off
ParamManager::getInstance()->add("IIIF_URL","https://archimage.com/image_request_iiif/");
// On the the directory image_request_iiif there is .htaccess for Apache2 but any IIIF server
can be connected

#####

#####

//
//          LOCAL PARAM
//
#####

//Defaults Images : there are currently set in the config folder but can be changed anywhere
on the server
//
// *.jpg // *.png are supported
//
// Set image not Found in results
ParamManager::getInstance()-
>add("IMAGE_NOT_FOUND",__DIR__.DIRECTORY_SEPARATOR."image_not_found.jpg");
// Set image not Accessible by rights in results
ParamManager::getInstance()-
>add("IMAGE_NOT_ACCESSIBLE",__DIR__.DIRECTORY_SEPARATOR."image_not_accessible.jpg");
// Set used Fonts in image process
ParamManager::getInstance()->add("FONT_WATERMARK",__DIR__.DIRECTORY_SEPARATOR."FreeSans.ttf");
// IMAGE DEFAULT FOR TYPE AUDIO
ParamManager::getInstance()-
>add("IMAGE_DEFAULT_AUDIO",__DIR__.DIRECTORY_SEPARATOR."image_audio.jpg");
// IMAGE DEFAULT FOR TYPE VIDEO
ParamManager::getInstance()-
>add("IMAGE_DEFAULT_VIDEO",__DIR__.DIRECTORY_SEPARATOR."image_video.jpg");
// IMAGE DEFAULT FOR TYPE TEXTE
ParamManager::getInstance()-
>add("IMAGE_DEFAULT_TEXT",__DIR__.DIRECTORY_SEPARATOR."image_text.jpg");
// IMAGE DEFAULT FOR TYPE TEXTE

```

```
ParamManager::getInstance()-
>add("IMAGE_DEFAULT_SPREADSHEET",__DIR__.DIRECTORY_SEPARATOR."image_spreadsheet.jpg");

// CONFIGURATION OF THE EXTERNAL DATA STORAGE
//#####
// ACTIVATE EFA API
ParamManager::getInstance()->add("EFA_API","0");
// URL OF THE JSON API FOR THE EFA RESSOURCES
ParamManager::getInstance()->add("MISSIONS_API","");
ParamManager::getInstance()->add("MANIFESTATIONS_API","");
ParamManager::getInstance()->add("CHRONIQUE_API","");
// URL OF THE ONIX FOR THE PUBLICATIONS
ParamManager::getInstance()->add("PUBLICATIONS_API","");
// Key to access the API Data export
ParamManager::getInstance()->add("DATA_API_TOKEN","");

// URL OF THE IDREF API
ParamManager::getInstance()->add("IDREF_API","https://www.idref.fr");

// URL FOR THE PUBLIC PUBLICTIONS CATALOG
// DATACITE API CONFIGURATION
ParamManager::getInstance()->add("DATACITE_API","https://api.test.datacite.org");
// "https://api.datacite.org"
ParamManager::getInstance()->add("DATACITE_BASE_URL","https://handle.test.datacite.org/");
// "https://doi.org/"
ParamManager::getInstance()->add("DATACITE_HOSTED_PAGES",
"https://doi.test.datacite.org/does/");
// "https://doi.datacite.org/does/"
ParamManager::getInstance()->add("DATACITE_PREFIX","");
ParamManager::getInstance()->add("DATACITE_USER","");
ParamManager::getInstance()->add("DATACITE_PASSWORD","");
ParamManager::getInstance()->add("DATACITE_EFE","");
ParamManager::getInstance()->add("DATACITE_SUFFIX","");
ParamManager::getInstance()->add("DATACITE_EMAIL","");
ParamManager::getInstance()->add("DATACITE_DEFAULT_AUTHOR","");
```

```
// SET A TEMPLATE FOR THE PUBLIC SITE -> exact dir name in the Archimage2/template/ dir
//ParamManager::getInstance()->add("TEMPLATE_STATIC_HTML","TEMPLATE_NAME");

// SET URL FOR THE ANALYTICS (MATOMO SUPPORT)
ParamManager::getInstance()->add("MATOMO_URL","");
ParamManager::getInstance()->add("MATOMO_ID","");
//#####

// CONFIGURATION OF THE MAPS BY LEAFLET OPTIONS
ParamManager::getInstance()->add("LEAFLET_HEIGHT","400");
ParamManager::getInstance()->add("LEAFLET_MAX_ZOOM","19");
ParamManager::getInstance()->add("LEAFLET_DEFAULT_ZOOM","6");
ParamManager::getInstance()->add("LEAFLET_DEFAULT_CENTER_X","37.39");
ParamManager::getInstance()->add("LEAFLET_DEFAULT_CENTER_Y","25.26");
ParamManager::getInstance()->add("LEAFLET_GEO_COLOR","#6B82AC");
ParamManager::getInstance()->add("LEAFLET_GEO_WEIGHT","1");
ParamManager::getInstance()->add("LEAFLET_GEO_OPACITY","0.1");
ParamManager::getInstance()->add("LEAFLET_MARKER_PRIVATE_TOO_LARGE","100000");
ParamManager::getInstance()->add("LEAFLET_MARKER_PRIVATE_LARGE","50000");
ParamManager::getInstance()->add("LEAFLET_MARKER_PRIVATE_MEDIUM","10000");
ParamManager::getInstance()->add("LEAFLET_MARKER_PRIVATE_SMALL","500");
ParamManager::getInstance()->add("LEAFLET_MARKER_PRIVATE_VERY_SMALL","50");
ParamManager::getInstance()->add("LEAFLET_MARKER_PUBLIC_TOO_LARGE","10000");
ParamManager::getInstance()->add("LEAFLET_MARKER_PUBLIC_LARGE","3000");
ParamManager::getInstance()->add("LEAFLET_MARKER_PUBLIC_MEDIUM","500");
ParamManager::getInstance()->add("LEAFLET_MARKER_PUBLIC_SMALL","100");
ParamManager::getInstance()->add("LEAFLET_MARKER_PUBLIC_VERY_SMALL","10");
ParamManager::getInstance()->add("LEAFLET_RESEARCH_HEIGHT","600");
ParamManager::getInstance()->add("LEAFLET_SECTEUR_LVL_0","#6B82AC");
ParamManager::getInstance()->add("LEAFLET_SECTEUR_LVL_0_W","1");
ParamManager::getInstance()->add("LEAFLET_SECTEUR_LVL_0_0","0.1");
ParamManager::getInstance()->add("LEAFLET_SECTEUR_LVL_1","#6B82AC");
ParamManager::getInstance()->add("LEAFLET_SECTEUR_LVL_1_W","1");
ParamManager::getInstance()->add("LEAFLET_SECTEUR_LVL_1_0","0.1");
ParamManager::getInstance()->add("LEAFLET_SECTEUR_LVL_2","#A2D5CD"); // 6EED7 98B6B1
```

```

ParamManager::getInstance()->add("LEAFLET_SECTEUR_LVL_2_W", "8");
ParamManager::getInstance()->add("LEAFLET_SECTEUR_LVL_2_0", "0.5");
ParamManager::getInstance()->add("LEAFLET_SECTEUR_LVL_3", "#74C796"); // 74C796 629677
ParamManager::getInstance()->add("LEAFLET_SECTEUR_LVL_3_W", "5");
ParamManager::getInstance()->add("LEAFLET_SECTEUR_LVL_3_0", "0.7");
ParamManager::getInstance()->add("LEAFLET_SECTEUR_LVL_4", "#495D63");
ParamManager::getInstance()->add("LEAFLET_SECTEUR_LVL_4_W", "1");
ParamManager::getInstance()->add("LEAFLET_SECTEUR_LVL_4_0", "1");

#####
//
//          END LOCAL PARAM
//
#####

#####
//
//          Visualization Options
//
#####

//Show Period open ? =>
ParamManager::getInstance()->add("FICHE_PUBLIQUE_PERIODES_EXPAND", "1"); // Opened => 1 //
Closed => 0
//Show Period link ? =>
ParamManager::getInstance()->add("FICHE_PUBLIQUE_PERIODES_LIENS", "2"); // nothing => 0 // full
url=> 1 // just ico =>2
// is Sector expanded
ParamManager::getInstance()->add("FICHE_PUBLIQUE_SECTEURS_EXPAND", "1"); // Opened => 1 //
Closed => 0
//
// ARCHIVES MANUSCRITES
ParamManager::getInstance()->add("AM_LABEL", "BOTH"); // REFERENCE LEGENDE BOTH
ParamManager::getInstance()->add("AM_TYPE", "NONE"); // NORMAL NONE
ParamManager::getInstance()->add("AM_TREE_FULL", "1"); // 1 #all tree in fiche

// Show author form
ParamManager::getInstance()->add("AUTEUR_FORME", "NORMAL"); // NORMAL IDREF

```

```

// DESCRIPTION FIELDS OPTIONS
//ParamManager::getInstance()->add("desc1","Example field name"); // label to replace desc1
//ParamManager::getInstance()->add("desc1Type","Text"); // Type : Int // Text // TextArea
// etc to ... 20
//ParamManager::getInstance()->add("desc2","Example field name"); // label que l'on veut
//ParamManager::getInstance()->add("desc2Type","Int"); // Int // Text // TextArea

//

// DC OPTIONS
ParamManager::getInstance()->add("DC_SOURCE","Institution Archive department");
ParamManager::getInstance()->add("DC_PUBLISHER","Institution Publisher");
ParamManager::getInstance()->add("DC_DEFAULT_LANGUAGE","en"); // fre ....

// EAD OPTIONS
ParamManager::getInstance()->add("EAD_CONTRIBUTOR","Institution contributor");
ParamManager::getInstance()->add("EAD_TITLE","Institution Archive department catalog");
ParamManager::getInstance()->add("EAD_HEADER_SITE_TITLE","");
ParamManager::getInstance()->add("EAD_HEADER_PROFILE_DESC","");
ParamManager::getInstance()->add("EAD_IDENTIFIER","");
ParamManager::getInstance()->add("EAD_COUNTRY_CODE","");
ParamManager::getInstance()->add("EAD_REPOSITORY_CODE","");
ParamManager::getInstance()->add("EAD_CORPNAME","");
ParamManager::getInstance()->add("EAD_SOURCE","");
ParamManager::getInstance()->add("EAD_SOURCE_SUBJECT","");
ParamManager::getInstance()->add("EAD_AUTH_NUMBER","");
ParamManager::getInstance()->add("EAD_ADDRESS_STREET","");
ParamManager::getInstance()->add("EAD_ADDRESS_CODE_CITY","");
ParamManager::getInstance()->add("EAD_ADDRESS_COUNTRY","");
ParamManager::getInstance()->add("EAD_ADDRESS_PHONE","");
ParamManager::getInstance()->add("EAD_ADDRESS_EMAIL","");
ParamManager::getInstance()->add("EAD_ACCESS_RESTRICT",""); // link allowed <extref
href="https://link.com\" show="new">for more informations ...</extref>
ParamManager::getInstance()->add("EAD_CREATION_PROFILE_1","");
ParamManager::getInstance()->add("EAD_CREATION_PROFILE_2","");
ParamManager::getInstance()->add("EAD_DEFAULT_LANGUAGE_ISO","en");
ParamManager::getInstance()->add("EAD_DEFAULT_LANGUAGE_LABEL","");
ParamManager::getInstance()->add("EAD_GENREFORM_TYPE","");
ParamManager::getInstance()->add("EAD_GENREFORM_LABEL","");
ParamManager::getInstance()->add("EAD_GEOGNAME_ROLE","");

```

```

ParamManager::getInstance()->add("EAD_SITE_CORPNAME_ORIGINATION","");
ParamManager::getInstance()->add("EAD_SITE_SCOPE_CONTENT","");
ParamManager::getInstance()->add("EAD_SITE_ACCRUALS","");
ParamManager::getInstance()->add("EAD_SITE_MISSION_PREFIX","");
ParamManager::getInstance()->add("EAD_SITE_DECENNIE_PREFIX","");
ParamManager::getInstance()->add("EAD_SITE_ANNEE_PREFIX","");
ParamManager::getInstance()->add("EAD_SITE_ARRANGEMENTS","");
ParamManager::getInstance()->add("EAD_SITE_URL_REPOSITORY","");
ParamManager::getInstance()->add("EAD_SITE_GENREFORM_LABEL","");
ParamManager::getInstance()->add("EAD_DEFAULT_DATE_NOT_SET","_Date_not_set");
ParamManager::getInstance()->add("EAD_DEFAULT_MISSION_NOT_SET","mission_not_specified");
ParamManager::getInstance()->add("EAD_DEFAULT_SECTEUR_NOT_SET","sector_not_specified");

// LICENCE USED
ParamManager::getInstance()->add("LICENCE_DESCRIPTIONS","Licence...");

// ACTIVATE LIASSES For IFAO
ParamManager::getInstance()->add("ACTIVATE_LIASSES","0"); // 0 disable or 1 activated

// PROXY THE IMAGES OF the ONIX IFAO
ParamManager::getInstance()->add("PUBLICATIONS_IMG_PROXY","0"); // 0 disable or 1 activated

// IIIF SHARE OPTIONS
ParamManager::getInstance()->add("DOWNLOAD_MESSAGE_TITLE","");
ParamManager::getInstance()->add("DOWNLOAD_MESSAGE","");
ParamManager::getInstance()->add("IIIF_ATTRIBUTION","Archimage Licensed");
ParamManager::getInstance()-
>add("IIIF_LICENCE","http://creativecommons.org/licenses/by/4.0/");
/* SALT FOR API KEY For CONSULTANT => if empty default one will be used */
ParamManager::getInstance()->add("API_KEY","");

// DEACTIVATE PUBLIC SEARCH
//ParamManager::getInstance()->add("DEACTIVATE_SEARCH","1,2,3,4");
/*
    RECHERCHE_SIMPLE=1;
    RECHERCHE_AVANCEE=2;
    RECHERCHE_INVENTAIRE=3;
    RECHERCHE_AM=4;
    RECHERCHE_GEO=5;

```



```
//ParamManager::getInstance()->redis_port='';
//ParamManager::getInstance()->redis_username='';
//ParamManager::getInstance()->redis_password='';
//ParamManager::getInstance()->redis_dbindex='';
//ParamManager::getInstance()->redis_timeout='';
//ParamManager::getInstance()->redis_read_timeout='1.5';
//ParamManager::getInstance()->redis_retry_interval='';

// A IIIF CACHE can be set For Performance
//ParamManager::getInstance()->add("IIIF_CACHE_DIRECTORY","/home/user/dir/");

// ACTIVATE the FranceArchives page with the OAI_PMH connector
//ParamManager::getInstance()->add("FranceArchives","0"); // 0 disable or 1 activated

// SHOW BETA Title
//ParamManager::getInstance()->add("BETA","0");

// ACTIVATE TWIG TEMPLATE
ParamManager::getInstance()->add("TWIG_TEMPLATE","1"); // MANDATORY

// SET IMAGE PROVIDER -> Default imagick see documentation
ParamManager::getInstance()->add("IMAGE_ENGINE","vips"); /* vips,imagick,gd */

ParamManager::getInstance()->add("LANGUAGES","fr"); // fr,en,es,gr,it

// set route identifier in url
KRoute::overrideKrouteIdentifier("r");
```

Révision #3

Créé 2026-05-06 10:19:38 EEST par EFA

Mis à jour 2026-05-06 11:12:24 EEST par EFA