Applications Documentation

SaltOS 4.0 r1923

March 2025

Contents

1 Certs

1.1 TODO

```
apps/certs/php/certs.php
```

TODO

1.1.1 TODO

```
function __certs_list($search, $offset, $limit)
```

TODO

1.1.2 TODO

```
function __certs_insert($json)
```

TODO

1.1.3 TODO

```
function __certs_hash2nick($hash)
```

TODO

1.1.4 TODO

```
function __certs_check($hash)
```

TODO

1.1.5 TODO

```
function __certs_view($hash)
```

TODO

1.1.6 TODO

```
function __certs_delete($hash)
```

TODO

1.2 Directory helper

```
apps/certs/php/nssdb.php
```

This function returns the nssdb directory used by all functions

1.2.1 Passthru helper

```
function __nssdb_passthru_helper($cmd)
```

This function uses the ob_passthru and improve the output

- @cmd => command line to be executed

1.2.2 Init nssdb repo

```
function __nssdb_init()
```

This function tries to initialize the nssdb component with an empty repo

1.2.3 Create certificate

```
function __nssdb_create($outfile, $outpass, $subject = '', $name = '')
```

This function create a dummy self signed certificate intended to be used in test cases

$outfile => the p12 file $outpass => the password used by the p12 file

1.2.4 Add certificate

```
function __nssdb_add($file, $pass)
```

This function adds the p12 file that must contains a valid certificate to the nssdb repo using the pass if needed

- @file => the p12 file
- @pass => the p12 password

1.2.5 List certificates

```
function __nssdb_list()
```

This function returns the list of valid nicks that you can use with pdfsig

1.2.6 Info of a certificate

```
function __nssdb_info($nick, $shortnames = false)
```

This function returns the info of the nick certificate of the nssdb repo

- @nick => the desired nick used to retrieve info

1.2.7 Pdf signature

```
function __nssdb_pdfsig($nick, $input, $output)
```

This function uses the pdfsig to add the signature to the pdf using the nick certificate of the nssdb repo

- @nick => the desired nick used to sign de pdf
- @input => the desired input file that you want to sign
- @output => the signed pdf file

1.2.8 Remove certificate

```
function __nssdb_remove($nick)
```

This function remove the certificate from the nssdb repo indentified by nick

- @nick => the desired nick that you want to remove

1.2.9 Reset nssdb repo

```
function __nssdb_reset()
```

This function removes the nssdb repo, is the inverted of the init function

1.2.10 Update pdf

```
function __nssdb_update($nick, $input)
```

This function creates a new pdf document using input as source and adding to each page a new box with the important information about the certificate used in the signature process

- @nick => the desired nick used to sign de pdf
- @input => the desired input pdf file where you want to add the cert info

2 Common

2.1 profile application

```
apps/common/js/profile.js
```

This application implements the tipical features associated to profile

2.1.1 Main object

```
saltos.profile = {};
```

This object contains all SaltOS code

2.1.2 TODO

```
saltos.profile.init = arg
```

TODO

2.1.3 TODO

```
saltos.profile.authupdate = ()
```

TODO

2.2 TODO

```
apps/common/php/actions.php
```

TODO

2.2.1 Merge data actions

```
function __merge_data_actions($data, $actions)
```

This function merge the rows of a table or list with the specified actions

- @data => the data of the table or list widget
- @actions => the desired actions to use in the table or list widget

2.3 Apps helper module

```
apps/common/php/default.php
```

This file contains functions intended to be used as hepers of other functions, allowing to convert between formats like yaml to xml using a small specification

2.3.1 Make app file

```
function make_app_file($data)
```

This function returns the xml of an app using the follow specs passed in @data:

- @require => this field defines the required php that contains the make_app_file function like this file
- @_template => the xml app file used as template, this file contains the default specification of the app used in the quick apps defined in the yaml
- @screen => the screen used by the template
- @list => the list of fields used in the list screen, this must be a list of arrays with 3 elements: id, type and label, types can be as follow: -text => default field with text -select => this field resolves the text from a table and field, see the @select spec -boolean => this field uses an icon with V o X in green or red to see the value of the field (generaly 1 or 0) -hastext => this field is similar to boolean but the true or false is detedmined using the contents of the text, no text is false and some text is true
- @form => the list of fields used in the form screen, this must be a list of arrays with 3 elements: id, type and label, types can be as follow:
    - text, date, time, datetime, checkbox, switch => default widget
    - textarea, ckeditor, codemirror => widget with 10em of height

  - select => widget that uses the select spec to resolve the contents

- @select => this spec is intended to defined the selects used in the list and forms, requires an array of 3 elements: id, table and field, if the last field is not specified, saltos tries to resolve it using the manifest information (internally use the table2field feature)
- @attr => this part contain an array with extra features added to the forms widgets, the spec requires to use a named array that defines the id of the field and each array entry must to be another pair of key and val with the name of the property and the value of it

2.4 TODO

```
apps/common/php/files.php
```

TODO

2.4.1 TODO

```
function __files_list($search, $offset, $limit)
```

TODO

2.4.2 TODO

```
function __files_getfile($file)
```

TODO

2.4.3 TODO

```
function __files_check($file)
```

TODO

2.4.4 TODO

```
function __files_view($file)
```

TODO

2.5 Matrix functions

```
apps/common/php/matrix.php
```

This file contain all functions needed by the excel widget

2.5.1 TODO

```
function make_matrix_version($app, $id)
```

TODO

2.5.2 TODO

```
function make_matrix_log($app, $id)
```

TODO

3 Dashboard

3.1 Dashboard application

```
apps/dashboard/js/dashboard.js
```

This application implements the tipical features associated to dashboard

3.1.1 Main object

```
saltos.dashboard = {};
```

This object contains all SaltOS code

3.1.2 TODO

```
saltos.dashboard.init = arg
```

TODO

3.2 Dashboard widgets application

```
apps/dashboard/js/dashboard_widgets.js
```

This application implements the tipical features associated to dashboard

3.2.1 Main object

```
saltos.dashboard_widgets = {};
```

This object contains all SaltOS code

3.2.2 TODO

```
saltos.dashboard_widgets.init = arg
```

TODO

4 Emails

4.1 Email application

```
apps/emails/js/driver.js
```

This application implements the tipical features associated to emails

4.1.1 Driver emails object

```
saltos.driver.__types.emails = {};
```

This object stores the functions used by the emails driver

4.1.2 TODO

```
saltos.driver.__types.emails.template = arg
```

TODO

4.1.3 TODO

```
saltos.driver.__types.emails.init = saltos.driver.__types.type5.init;
```

TODO

4.2 Email application

```
apps/emails/js/emails.js
```

This application implements the tipical features associated to emails

4.2.1 Main object

```
saltos.emails = {};
```

This object contains all SaltOS code

4.2.2 TODO

```
saltos.emails.init = arg
```

TODO

4.2.3 TODO

```
saltos.emails.server = ()
```

TODO

4.2.4 TODO

```
saltos.emails.delete1 = ()
```

TODO

4.2.5 TODO

```
saltos.emails.delete2 = ()
```

TODO

4.2.6 TODO

```
saltos.emails.send = ()
```

TODO

4.2.7 TODO

```
saltos.emails.setter = what
```

TODO

4.2.8 TODO

```
saltos.emails.signature = ()
```

TODO

4.2.9 TODO

```
saltos.emails.viewpdf1 = ()
```

TODO

4.2.10 TODO

```
saltos.emails.download1 = ()
```

TODO

4.2.11 TODO

```
saltos.emails.source = ()
```

TODO

4.2.12 TODO

```
saltos.emails.viewpdf2 = ()
```

TODO

4.2.13 TODO

```
saltos.emails.download2 = ()
```

TODO

4.2.14 TODO

```
saltos.emails.reply = ()
```

TODO

4.2.15 TODO

```
saltos.emails.replyall = ()
```

TODO

4.2.16 TODO

```
saltos.emails.forward = ()
```

TODO

4.3 TODO

```
apps/emails/php/filter.php
```

TODO

4.3.1 TODO

```
function make_where_query_emails($json)
```

TODO

4.4 Get email library

```
apps/emails/php/getmail.php
```

This library provides the necesary functions to download, parse and manage emails.

4.4.1 Requires section

```
require_once 'apps/emails/lib/mimeparser/mime_parser.php';
```

This requires loads the external libraries needed to run this library.

4.4.2 Defines section

```
// phpcs:disable Generic.Files.LineLength
```

This defines allow to define some useful standards to do html pages and more.

4.4.3 Remove all body

```
function __getmail_removebody($array)
```

This function removes the body entry in the array, it is only for debug purposes

- @aarray => The array that you want to process

4.4.4 Process message

```
function __getmail_processmessage($disp, $type)
```

This function returns a boolean that identify if the disposition and the type allow the node to be processed.

- @disp => disposition, can be inline or attachment
- @type => type, can be message, html or plain

4.4.5 Process plain html

```
function __getmail_processplainhtml($disp, $type)
```

This function returns a boolean that identify if the disposition and the type allow the node to be processed.

- @disp => disposition, can be inline or attachment
- @type => type, can be message, html or plain

4.4.6 Process file

```
function __getmail_processfile($disp, $type)
```

This function returns a boolean that identify if the disposition and the type allow the node to be processed.

- @disp => disposition, can be inline or attachment
- @type => type, can be message, html or plain

4.4.7 Check permissions

```
function __getmail_checkperm($id)
```

This function allow to check if the current user has permissions to view the message identified by the id argument

- @id => id of the email

4.4.8 Get GZ File

```
function __getmail_gzfile($id)
```

This function returns the file that contains the email

- @id => id of the email

4.4.9 Get source

```
function __getmail_getsource($id)
```

This function returns the original RFC822 message as string

- @id => id of the email

Notes:

This function returns the email source without the base64 attachments

4.4.10 Mime decode protected

```
function __getmail_mime_decode_protected($input)
```

This function decodes the input string that contains the RFC822 message using the mime_parser_class to do it, and returns the decoded array.

- @input => the RFC822 string that contains the message

4.4.11 Get mime

```
function __getmail_getmime($id)
```

This function returns the decoded array of the email identified by the id argument, to do this more optimal, this function uses an internal cache file to improve the performance for repeated executions.

- @id => id of the email

4.4.12 Get Node

```
function __getmail_getnode($path, $array)
```

This function returns the node using a xpath notation

- @path => xpath that identify the desired path that must to be returned
- @array => the decoded message in an array format

4.4.13 Get type

```
function __getmail_gettype($array)
```

This function tries to unify the differents content-type to standarize it into the follow formats: html, plain, messsage, alternative, multipart or other.

- @array => the decoded message in an array format

4.4.14 Get disposition

```
function __getmail_getdisposition($array)
```

This function tries to unify the differents content-dispoaition to standarize it into the follow formats: attachment, inline or other.

- @array => the decoded message in an array format

4.4.15 Get files

```
function __getmail_getfiles($array, $level = 0)
```

This function returns an array with the attachment files of the message

- @array => the decoded message in an array format
- @level => this parameter is internally used to detect recursion

4.4.16 Get info

```
function __getmail_getinfo($array)
```

Returns all information of the decoded message in a structured format

- @array => the decoded message in an array format

4.4.17 Fix string

```
function __getmail_fixstring($arg)
```

This function is a helper used by all functions that pcoesses the headers of the decoded message.

- @arg => the string that must to be checked and fixed if needed

4.4.18 Get text body

```
function __getmail_gettextbody($array, $level = 0)
```

This function returns all text body concatenated as an unique string

- @array => the decoded message in an array format
- @level => this parameter is internally used to detect recursion

4.4.19 Get full body

```
function __getmail_getfullbody($array)
```

This function returns all body and attachments information as an array

- @array => the decoded message in an array format

4.4.20 Get cid

```
function __getmail_getcid($array, $hash)
```

This function returns the requested attachment indentified by the hash argument

- @array => the decoded message in an array format
- @hash => the hash of the content requested

4.4.21 Insert

```
function __getmail_insert(
```

This function do the insert in the app_emails table, and

- @file => the gzfile that contains the message in RFC822 format
- @messageid => the id of the message (account_id/uidl)
- @state_new => the 0/1 that sets the state new flag
- @state_reply => the 0/1 that sets the state reply flag
- @state_forward => the 0/1 that sets the state forward flag
- @state_wait => the 0/1 that sets the state wait flag
- @id_correo => the id of the related email (used to create relations between emails)

- @is_outbox => the 0/1 that sets the is outbox flag
- @state_sent => the 0/1 that sets the state sent flag
- @state_error => the string that contains the error (if exists an error)

4.4.22 Update

```
function __getmail_update($field, $value, $id)
```

This function updates the field with the value of the app_emails for the register identified by the id argument.

- @field => field that you want to update
- @value => value that you want to set
- @id => id of the register to do the update

4.4.23 Add bcc

```
function __getmail_add_bcc($id, $bcc)
```

This function adds the bbc to the database, this is because the messages does not contains the bcc field (is hidden in theory), and only is available if the current execution is the sender of the message.

- @id => id of the email
- @bcc => an array with the addresses of the emails

4.4.24 Gzfile size

```
function gzfilesize($filename)
```

This function is copied from http://php.net/manual/es/function.gzread.php#110078 and allow to know the file size of the file after a gzip descompression.

- @filename => the gzip filename that you want to know the size

4.4.25 Get email body

```
function getmail_body($id, $images = false)
```

This function returns the string that contains the body of the email intended to be rendered in an iframe, for example

- @id => id of the email

4.4.26 TODO

```
function __getmail_head_helper($decoded, $email_id)
```

TODO

4.4.27 TODO

```
function __getmail_body_helper($decoded, $images = false)
```

TODO

4.4.28 Get email source

```
function getmail_source($id)
```

This function returns the string that contains the source of the email intended to be rendered in an iframe, for example

- @id => id of the email

4.4.29 Get email files

```
function getmail_files($id)
```

This function returns an arryy that contains the files of the email intended to be rendered in an table, for example

- @id => id of the email

4.4.30 Get cid

```
function getmail_cid($id, $cid)
```

This function returns the requested attachment indentified by the cid argument

- @id => id of the email
- @cid => the cid of the content requested

4.4.31 Is outbox

```
function getmail_field($field, $id)
```

Returns the field of the email identified by the id argument

- @field => field requested
- @id => id of the email

4.4.32 Server

```
function getmail_server()
```

This function implements the old getmail action of the old saltos.

4.4.33 Delete

```
function getmail_delete($ids)
```

This function implements the old delete action of the old saltos.

- @ids => array with the emails id

4.4.34 Get viewpdf

```
function getmail_viewpdf($id, $cid)
```

This function returns the requested attachment indentified by the cid argument in a pdf format for the viewpdf widget

- @id => id of the email
- @cid => the cid of the content requested

4.4.35 Get download

```
function getmail_download($id, $cid)
```

This function returns the requested attachment indentified by the cid argument in an array format for the download feature

- @id => id of the email
- @cid => the cid of the content requested

4.4.36 TODO

```
function getmail_setter($ids, $what)
```

TODO

4.4.37 TODO

```
function getmail_pdf($ids)
```

TODO

4.4.38 TODO

```
function __iframe_srcdoc_helper($html)
```

TODO

4.5 Html helper module

```
apps/emails/php/html.php
```

This file contain useful html helper functions

4.5.1 Defines section

```
define(’__GIF_IMAGE__’, ’data:image/gif;base64,R0lGODlhCgABAIABAOns7////
    ywAAAAACgABAAACA4SPBQA7’);
```

This defines allow to define some useful needed resources by this file.

4.5.2 Remove Script Tag

```
function remove_script_tag($html)
```

This function tries to remove all <script> tags of the string

- @temp => the string that you want to process

4.5.3 Remove Style Tag

```
function remove_style_tag($html)
```

This function tries to remove all <style> tags of the string

- @temp => the string that you want to process

4.5.4 Remove Comment Tag

```
function remove_comment_tag($html)
```

This function tries to remove all <!– –> tags of the string

- @temp => the string that you want to process

4.5.5 Remove Meta Tag

```
function remove_meta_tag($html)
```

This function tries to remove all <meta > tags of the string

- @temp => the string that you want to process

4.5.6 Remove Link Tag

```
function remove_link_tag($html)
```

This function tries to remove all <link > tags of the string

- @temp => the string that you want to process

4.5.7 Inline Img Tag

```
function inline_img_tag($html)
```

This function tries to convert all imgs to an inline imgs

- @temp => the string that you want to process

4.5.8 Inline Img Style

```
function inline_img_style($html)
```

This function tries to convert all imgs to an inline imgs

- @temp => the string that you want to process

4.5.9 Inline Img Background

```
function inline_img_background($html)
```

This function tries to convert all imgs to an inline imgs

- @temp => the string that you want to process

4.5.10 TODO

```
function __inline_img_helper($src)
```

TODO

4.5.11 TODO

```
function extract_img_tag($html)
```

TODO

4.5.12 TODO

```
function extract_img_style($html)
```

TODO

4.5.13 TODO

```
function extract_img_background($html)
```

TODO

4.5.14 TODO

```
function fix_img_tag($html)
```

TODO

4.5.15 TODO

```
function fix_img_style($html)
```

TODO

4.5.16 TODO

```
function fix_img_background($html)
```

TODO

4.5.17 TODO

```
function fix_file_b64($html)
```

TODO

Notes:

We are using {48} to force to search a base64 data of 48 of bytes, where decoded must to be a 36 bytes string, this match with the cache style file like a md5 hash of 32 bytes with 4 bytes more by the extension file (.b64)

4.6 Make indexing action

```
apps/emails/php/indexing.php
```

This fie contains useful functions related to the indexing action that internally uses the mroonga engine to search in the fulltext string generated by this action

4.7 Send email library

```
apps/emails/php/sendmail.php
```

This library provides the necesary functions to send emails.

4.7.1 Used libraries

```
use PHPMailer\PHPMailer\PHPMailer;
```

This use loads the external libraries needed to run this library.

4.7.2 Sendmail

```
function sendmail($account_id, $to, $subject, $body, $files = '', $async =
    true)
```

This function send an email in synchronous and/or asynchronous mode

$account_id => the account id used to detect the source of the email $to => can be an string with the destination email or an array with the follow prefixes => to:, cc:, bcc:, crt:, priority:, sensitivity:, replyto $subject => the subject string $body => the body string $files => an array with files

Notes:

This function must return the last_id if success, and a string if error

4.7.3 Debugoutput helper

```
function __sendmail_debugoutput_helper($str, $level)
```

This function tries to echoed all debug information with the SMTP Error: prefix, with this feature we can capture smtp errors with more details like the error stored in $mail->ErrorInfo that always contains SMTP connect() failed, the signature of this function will accomplish the neested callback arguments

- @str => the debug trace string that can contain the errors
- @level => unused in this function

4.7.4 Parser

```
function __sendmail_parser($oldaddr)
```

This function gets an address and tries to detect the name part and the addr part of the argument. It's returns an array with two elements, the first is for the addr and the second is for the name.

- @oldaddr => the string that must to be processed

4.7.5 Message Id

```
function __sendmail_messageid($account_id, $from)
```

This function returns the message id for a new email, to do it, tries to detect the outbox directory, compute an aproximation to the newest value and checks that is unique in the system to prevent concurrence.

- @account_id => the account id used to send the new email
- @from => the from used to compute the crc32

4.7.6 Eml saver

```
function __sendmail_emlsaver($message, $messageid)
```

This function is intended to save the RFC822 message into the eml gzfile

- @message => the contents in RFC822 format of the message
- @messageid => the message id computed previously

4.7.7 Obj saver

```
function __sendmail_objsaver($mail, $messageid)
```

This function is intended to save the PHPMailer object into the obj file

- @mail => the PHPMailer object of the asynchronous transaction
- @messageid => the message id computed previously

4.7.8 TODO

```
function sendmail_prepare($action, $email_id)
```

TODO

4.7.9 TODO

```
function sendmail_action($json, $action, $email_id)
```

TODO

4.7.10 TODO

```
function sendmail_server()
```

TODO

4.7.11 TODO

```
function sendmail_files($action, $email_id)
```

TODO

4.7.12 TODO

```
function sendmail_signature($json)
```

TODO

5 Invoices

5.1 invoices application

```
apps/invoices/js/invoices.js
```

This application implements the tipical features associated to invoices

5.1.1 Main object

```
saltos.invoices = {};
```

This object contains all SaltOS code

5.1.2 TODO

```
saltos.invoices.init = arg
```

TODO

5.1.3 TODO

```
saltos.invoices.compute_total = ()
```

TODO

5.1.4 TODO

```
saltos.invoices.add_item = ()
```

TODO

5.1.5 TODO

```
saltos.invoices.remove_item = (obj)
```

TODO

5.1.6 TODO

```
saltos.invoices.viewpdf = ()
```

TODO

5.1.7 TODO

```
saltos.invoices.download = ()
```

TODO

6 Tester

6.1 Tester application

```
apps/tester/js/tester.js
```

This application implements the tipical features associated to tester

6.1.1 Tester object

```
saltos.tester = {};
```

This object stores all function used by this app

6.1.2 Campo 8

```
saltos.tester.campo8 = ()
```

TODO

6.1.3 Campo 9

```
saltos.tester.campo9 = ()
```

TODO

6.1.4 Campo 10

```
saltos.tester.campo10 = ()
```

TODO

6.1.5 Campo 11

```
saltos.tester.campo11 = ()
```

TODO

6.1.6 Campo 22

```
saltos.tester.campo23 = ()
```

TODO

7 Users

7.1 Login application

```
apps/users/js/login.js
```

This application implements the tipical features associated to login

7.1.1 Main object

```
saltos.login = {};
```

This object contains all SaltOS code

7.1.2 Authenticate login function

```
saltos.login.authenticate = async ()
```

This function tries to authenticate the user using the user and pass fields of the form, to do it uses the authenticate function that send data to the authtoken action

7.2 Days functions

```
apps/users/php/days.php
```

This file contain all functions needed by the days feature

7.2.1 Days to bin

```
function days2bin($days)
```

This function tries to convert the days format used by the multiselect to the string expected by the database formed by ones and zeroes to represent if a day is operative for the user or not, for example, the selection 64,32,16,8,4 is returned like from monday to friday (1111100)

- @days => the string containing the days in power of two separated by comma

7.2.2 Bin to days

```
function bin2days($days)
```

This function tries to do the reverse action that the previous function, is able to get an string like 1111100 and returns the list of all bits in decimal like 64,32,16,8,4.

- @days => the string containing the days in binary format

7.2.3 Fix for days

```
function fix4days($data)
```

This function is intended to be used as wrapper in the result of the query that contains an element called days, in the database the days is stored using the binary notation like 1111100, and for the user interface, is needed to translate this string into a decimal string like 64,32,16,8,4.

- @data => the data obtained from an execute_query, for example, they must contain an entry called days.

7.3 Groups functions

```
apps/users/php/groups.php
```

This file contain all functions needed by the groups app

7.3.1 Insert group action

```
function insert_group($data)
```

This action allow to insert registers in the database associated to the groups app and only requires the data.

TODO

7.3.2 Update group action

```
function update_group($group_id, $data)
```

This action allow to update registers in the database associated to the groups app and requires the group_id and data.

TODO

7.3.3 Delete group action

```
function delete_group($group_id)
```

This action allow to delete registers in the database associated to the groups app and only requires the group_id.

TODO

7.4 Matrix functions

```
apps/users/php/matrix.php
```

This file contain all functions needed by the excel widget

7.4.1 Make matrix

```
function make_matrix_data($perms, $apps, $main, $user)
```

TODO

7.4.2 Make matrix

```
function make_matrix_cell($perms, $apps, $main, $user)
```

TODO

7.4.3 Make matrix

```
function unmake_matrix_data($perms, $apps, $main, $json)
```

TODO

7.5 Users functions

```
apps/users/php/users.php
```

This file contain all functions needed by the users app

7.5.1 Insert user action

```
function insert_user($data)
```

This action allow to insert registers in the database associated to the users app and only requires the data.

TODO

7.5.2 Update user action

```
function update_user($user_id, $data)
```

This action allow to update registers in the database associated to the users app and requires the user_id and data.

TODO

7.5.3 Delete user action

```
function delete_user($user_id)
```

This action allow to delete registers in the database associated to the users app and only requires the user_id.

TODO