Application Documentation

SaltOS 4.0 r761

March 2024

Contents

1 Javascript

1.1 Customers application

```
apps/customers/js/app.js
```

This application implements the tipical features associated to customers

1.1.1 Main object

```
saltos.customers = {};
```

This object contains all SaltOS code

1.1.2 TODO

```
saltos.customers.initialize_update_list = ()
```

TODO

1.1.3 TODO

```
saltos.customers.initialize_update_view = ()
```

TODO

1.1.4 TODO

```
saltos.customers.search = ()
```

TODO

1.1.5 TODO

```
saltos.customers.reset = ()
```

TODO

1.1.6 TODO

```
saltos.customers.more = ()
```

TODO

1.1.7 TODO

```
saltos.customers.insert = ()
```

TODO

1.1.8 TODO

```
saltos.customers.update = ()
```

TODO

1.1.9 TODO

```
saltos.customers.delete1 = arg
```

TODO

1.1.10 TODO

```
saltos.customers.delete2 = ()
```

TODO

1.2 Dashboard application

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

This application implements the tipical features associated to dashboard

1.2.1 Main object

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

This object contains all SaltOS code

1.2.2 TODO

```
saltos.dashboard.initialize = ()
```

TODO

1.3 Email application

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

This application implements the tipical features associated to emails

1.3.1 Main object

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

This object contains all SaltOS code

1.3.2 Initialize emails app

```
saltos.emails.initialize = ()
```

This function initializes the emails app screen to improve the user experience.

1.3.3 TODO

```
saltos.emails.search = ()
```

TODO

1.3.4 TODO

```
saltos.emails.reset = ()
```

TODO

1.3.5 TODO

```
saltos.emails.more = ()
```

TODO

1.3.6 TODO

```
saltos.emails.getmail = ()
```

TODO

1.3.7 TODO

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

TODO

1.3.8 TODO

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

TODO

1.3.9 TODO

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

TODO

1.3.10 TODO

```
saltos.emails.download = (file)
```

TODO

1.3.11 TODO

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

TODO

2 Emails

2.1 Get email library

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

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

2.1.1 Requires section

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

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

2.1.2 Defines section

```
define("__HTML_PAGE_OPEN__", '<!DOCTYPE html><html><head><style type="text/
    css">body{margin:0px;padding:6px 12px;}</style></head><body>');
```

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

2.1.3 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

2.1.4 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

2.1.5 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

2.1.6 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

2.1.7 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

2.1.8 Get source

```
function __getmail_getsource($id, $max = 0)
```

This function returns the original RFC822 message as string

- @id => id of the email
- @max => max size that can be processed

2.1.9 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

2.1.10 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

2.1.11 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

2.1.12 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

2.1.13 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

2.1.14 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

2.1.15 Get human size

```
function __getmail_gethumansize($size)
```

This function returns an string containing the size in human format

- @size => the number of bytes to convert to human format

2.1.16 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

2.1.17 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

2.1.18 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

2.1.19 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

2.1.20 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

2.1.21 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)

2.1.22 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

2.1.23 Raw url decode

```
function __getmail_rawurldecode($temp)
```

This function tries to detect if the argument contains the %20 string to try to detect url encoded strings and fix it if is needed

- @temp => the string that you want to check and fix

2.1.24 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

2.1.25 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

2.1.26 Get email body

```
function getmail_body($id)
```

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

2.1.27 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

2.1.28 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

2.1.29 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

2.1.30 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

2.1.31 Receive

```
function getmail_receive()
```

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

2.1.32 Delete

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

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

- @ids => array with the emails id

2.1.33 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

2.1.34 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

2.1.35 TODO

```
function getmail_setter($ids, $action2)
```

TODO

2.2 Send email library

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

This library provides the necesary functions to send emails.

2.2.1 Used libraries

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

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

2.2.2 Sendmail

```
function sendmail($account_id, $to, $subject, $body, $files = "")
```

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

2.2.3 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

2.2.4 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

2.2.5 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

2.2.6 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

3 Javascript

3.1 invoices application

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

This application implements the tipical features associated to invoices

3.1.1 Main object

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

This object contains all SaltOS code

3.1.2 TODO

```
saltos.invoices.initialize_update_list = ()
```

TODO

3.1.3 TODO

```
saltos.invoices.initialize_update_view = ()
```

TODO

3.1.4 TODO

```
saltos.invoices.initialize_buttons = ()
```

TODO

3.1.5 TODO

```
saltos.invoices.initialize_inputs = ()
```

TODO

3.1.6 TODO

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

TODO

3.1.7 TODO

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

TODO

3.1.8 TODO

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

TODO

3.1.9 TODO

```
saltos.invoices.search = ()
```

TODO

3.1.10 TODO

```
saltos.invoices.reset = ()
```

TODO

3.1.11 TODO

```
saltos.invoices.more = ()
```

TODO

3.1.12 TODO

```
saltos.invoices.insert = ()
```

TODO

3.1.13 TODO

```
saltos.invoices.update = ()
```

TODO

3.1.14 TODO

```
saltos.invoices.delete1 = arg
```

TODO

3.1.15 TODO

```
saltos.invoices.delete2 = ()
```

TODO

3.2 Login application

```
apps/login/js/app.js
```

This application implements the tipical features associated to login

3.2.1 Main object

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

This object contains all SaltOS code

3.2.2 Authenticate login function

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

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

3.2.3 Access denied

```
saltos.login.access_denied = ()
```

This function displays a modal dialog with the tipical access denied message

3.3 Tester application

```
apps/tester/js/app.new.js
```

This application implements the tipical features associated to tester

3.3.1 Tester object

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

This object stores all function used by this app

3.3.2 Campo 8

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

TODO

3.3.3 Campo 9

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

TODO

3.3.4 Campo 10

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

TODO

3.3.5 Campo 11

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

TODO

3.3.6 Campo 22

```
saltos.tester.campo22 = ()
```

TODO

3.4 Tester application

```
apps/tester/js/app.old.js
```

This application implements the tipical features associated to tester

3.4.1 Main code

```
(function() {
```

This function executes the main code that allow to you to see the tester in action

4 Tester

4.1 TODO

```
apps/tester/php/indexing.php
```

4.2 TODO

```
apps/tester/php/insert.php
```

4.3 TODO

```
apps/tester/php/pepe.php
```

5 Javascript

5.1 Types application

```
apps/types/js/app.1.js
```

This application implements the tipical features associated to types

5.1.1 Main object

```
saltos.types = {};
```

This object contains all SaltOS code

5.1.2 Initialize types

```
saltos.types.initialize = ()
```

This function initializes the types screen to improve the user experience.

5.1.3 TODO

```
saltos.types.search = ()
```

TODO

5.1.4 TODO

```
saltos.types.reset = ()
```

TODO

5.1.5 TODO

```
saltos.types.more = ()
```

TODO

5.1.6 TODO

```
saltos.types.open = arg
```

TODO

5.1.7 TODO

```
saltos.types.__open_helper = arg
```

TODO

5.1.8 TODO

```
saltos.types.close = ()
```

TODO

5.1.9 TODO

```
saltos.types.insert = arg
```

TODO

5.1.10 TODO

```
saltos.types.update = arg
```

TODO

5.1.11 TODO

```
saltos.types.delete = arg
```

TODO

5.2 types application

```
apps/types/js/app.2.js
```

This application implements the tipical features associated to types

5.2.1 Main object

```
saltos.types = {};
```

This object contains all SaltOS code

5.2.2 Initialize types

```
saltos.types.initialize_update_list = ()
```

This function initializes the types screen to improve the user experience.

5.2.3 TODO

```
saltos.types.initialize_update_view = ()
```

TODO

5.2.4 TODO

```
saltos.types.search = ()
```

TODO

5.2.5 TODO

```
saltos.types.reset = ()
```

TODO

5.2.6 TODO

```
saltos.types.more = ()
```

TODO

5.2.7 TODO

```
saltos.types.insert = ()
```

TODO

5.2.8 TODO

```
saltos.types.update = ()
```

TODO

5.2.9 TODO

```
saltos.types.delete1 = arg
```

TODO

5.2.10 TODO

```
saltos.types.delete2 = ()
```

TODO

5.3 Types application

```
apps/types/js/app.3.js
```

This application implements the tipical features associated to types

5.3.1 Main object

```
saltos.types = {};
```

This object contains all SaltOS code

5.3.2 Initialize types

```
saltos.types.initialize = ()
```

This function initializes the types screen to improve the user experience.

5.3.3 TODO

```
saltos.types.search = ()
```

TODO

5.3.4 TODO

```
saltos.types.reset = ()
```

TODO

5.3.5 TODO

```
saltos.types.more = ()
```

TODO

5.3.6 TODO

```
saltos.types.open = arg
```

TODO

5.3.7 TODO

```
saltos.types.__open_helper = arg
```

TODO

5.3.8 TODO

```
saltos.types.close = ()
```

TODO

5.3.9 TODO

```
saltos.types.insert = arg
```

TODO

5.3.10 TODO

```
saltos.types.update = arg
```

TODO

5.3.11 TODO

```
saltos.types.delete = arg
```

TODO