Code documentation

SaltOS 4.0 r733

March 2024

Contents

11

14

Chapter 1

Actions

1.1 Add error action

`api/php/action/adderror.php`

This file implements the adderror action, requires a POST JSON request with the follow elements: jserror, details and backtrace, this action is called from window.onerror in order to store the details of the js error

- @jserror => text used as title in the error report
- @details => text used as details in the error report
- @backtrace => array with the backtrace used in the error report

1.2 Add files action

`api/php/action/addfiles.php`

This file implements the delete files action, requires a POST JSON request with an array of files, and each array must contain the follow entries: id, name, size, type, data, error, file, hash

This action checks that not error is found, get the data and clear the data element of the array, check the prefix of the data using the type, check the size of the data, and then, set the file and hash to the array and store the file in the upload directory

- @files => array of files, each element must contain the follow elements:
- @id => unique id that is used by the client to identify the response
- @name => the name of the file
- @size => the size of the file
- @type => the type of the file
- @data => the contents of the file encoded as inline base64
- @error => the error in case of errors
- @file => this field is used here to put the local filename used in the file
- @hash => this field contains the hash of the contents of the file

1.3 Add log action

`api/php/action/addlog.php`

This file implements the addlog action, requires a POST JSON request with an element in the json that contains the message to be added

- @msg => message that you want to add to the log file

1.4 Application action

`api/php/action/app.php`

This file implements the app action, requires a GET REST request and the order of the elements are:

- @1 => the app that you want to execute
- @2 => the subapp that tou want to use, if the app only contains one subapp, this parameter is not necesary
- @3 => the id used in some subapps, for example, to get the data of specific customer using the id

1.5 Authentication token action

`api/php/action/authtoken.php`

This file implements the login action, allowing to authenticate users using the pair of login/password to validate the credentials and get a valid token to operate in SaltOS

- @user => username used in the authentication process
- @pass => password used in the authentication process

This action not requires a valid token, all valid tokens associated to the user will be revoked when a new token is assigned, as the result of this action is a flag that indicates the validity of the token, this action returns a json with the status of te token instead of returns a json with an error in case of non validity

1.6 Authentication update action

`api/php/action/authupdate.php`

This file implements the update password action, allowing to authenticated users by a token, and providing the old password to update a new password

- @oldpass => Old password, must to validate the active password of the user associated to the token used in the action
- @newpass => New password, must to be new, must to pass the score check and never must to be used in the system for the user
- @renewpass => The repeated new password, to prevent writing errors

This action requires a valid token associated to the user that wants to do the password update

1.7 BarCode action

`api/php/action/barcode.php`

This action allow to generate a barcode, you can pass the desired message that you want to convert in barcode

- @msg => the msg that you want to codify in the qrcode
- @format => the format used to the result, only can be png or json
- @w => width of each unit's bar of the barcode
- @h => height of the barcode (without margins and text footer)
- @m => margin of the barcode (white area that surround the barcode)
- @s => size of the footer text, not used if zero
- @t => type of the barcode, C128 is the most common type used

1.8 Captcha action

`api/php/action/captcha.php`

This action allo to retrieve the captcha of a randomly number or math operation, used to prevent massive requests, can perform the action of create the captcha image and return the result as a simple image or as a json image

- @type => the type used to the result, only can be number or math
- @format => the format used to the result, only can be png or json
- @width => the width of the generated image
- @height => the height of the generated image
- @letter => the size of the letters of the generated image
- @number => the size of the numbers of the generated image
- @angle => the angle allowed to rotate the letters and numbers
- @color => the color user to paint the code
- @bgcolor => the background color of the image
- @fgcolor => the color used to paint the letters of the background of the image
- @period => parameter for the wave transformation
- @amplitude => parameter for the wave transformation
- @blur => true or false to enable or disable the blur effect

1.9 Check token action

`api/php/action/checktoken.php`

This file implements the check action, allowing to check token's validity, the check action only can be performed by the same actor that execute the login action

The unique requirement to execute this action is to have a token to be checked, as the result of this action is a flag that indicates the validity of the token, this action returns a json with the status of te token instead of returns a json with an error in case of non validity

1.10   Deauthentication token action

`api/php/action/deauthtoken.php`

This file implements the logout action, allowing to deauthenticate users using a valid token, for security reasons, the deauth action only can be performed by the same actor that execute the login action

The unique requirement to execute this action is to have a valid token

1.11   Delete action

`api/php/action/delete.php`

This action allow to delete registers in the database associated to each app

TODO

1.12   Delete files action

`api/php/action/delfiles.php`

This file implements the delete files action, requires a POST JSON request with an array of files, and each array must contain the follow entries: id, name, size, type, data, error, file, hash

This action checks that not error is found, checks the file element, the size of the file, the hash of the file, and then, remove the file and clear the file and hash element of the array

- @files => array of files, each element must contain the follow elements:
- @id => unique id that is used by the client to identify the response
- @name => the name of the file
- @size => the size of the file
- @type => the type of the file
- @data => the contents of the file encoded as inline base64
- @error => the error in case of errors
- @file => this field is used here to put the local filename used in the file
- @hash => this field contains the hash of the contents of the file

1.13   Insert action

`api/php/action/insert.php`

This action allow to insert registers in the database associated to each app

TODO

1.14 List action

```
api/php/action/list.php
```

This action tries to facility the creation of lists with the tipicals features suck as rows, actions for each row, and other improvements as the list with count and without count.

TODO: pending to add the order by from the list header

1.15 QRCode action

```
api/php/action/qrcode.php
```

This action allow to generate a qrcode with the SaltOS logo embedded in the center of the image, you can pass the desired message that you want to convert in qrcode.

- @msg => the msg that you want to codify in the qrcode
- @format => the format used to the result, only can be png or json
- @s => size of each pixel used in the qrcode
- @m => margin of the qrcode (white area that that surround the qrcode)

1.16 Reauthentication token action

```
api/php/action/reauthtoken.php
```

This file implements the renew action, allowing to renew tokens before expire, for security reasons, the deauth action only can be performed by the same actor that execute the login action

The unique requirement to execute this action is to have a valid token and have a token with available renewals

1.17 Score action

```
api/php/action/score.php
```

This action allo to retrieve the score of a password, intended to be used as helper previously to the authupdate call, can perform the action of compute the score and return the result as a simple image or as a json image

- @pass => the password that you want to compute the score
- @format => the format used to the result, only can be png or json
- @width => the width of the generated image
- @height => the height of the generated image
- @size => the size of the font of the generated image

1.18   Update action

`api/php/action/update.php`

This action allow to update registers in the database associated to each app and requires the app, id, data and a valid token.

TODO

Chapter 2

Autoload

2.1 Apps helper module

`api/php/autoload/apps.php`

This file contains functions intended to be used as hepers of other functions, allowing to convert between formats as the name of the app to and app id, or viceversa

2.1.1 Apps helper function

`function __apps($fn, $arg)`

This function is used by the XXX2YYY functions as helper, it stores the dictionary of all conversions and resolves the data using it

- @fn => the caller function
- @arg => the argument passed to the function

2.1.2 Subtables helper

`function __apps_subtables_helper($subtables)`

This function helps the follow apps to returns the subtables information into a structure instead of string, this is very helpfull by the code that consumes this information, allow to use it quickly without parsing the original string.

- @subtables => the string that contains the subtables specification

2.1.3 Id to App

`function id2app($id)`

This function resolves the code of the app from the app id

- @id => the app id used to resolve the code

2.1.4 App to Id

```
function app2id($app)
```

This function resolves the id of the app from the app code

- @app => the code used to resolve the id

2.1.5 Id to Table

```
function id2table($id)
```

This function resolves the table of the app from the app id

- @id => the app id used to resolve the table

2.1.6 App to Table

```
function app2table($app)
```

This function resolves the table of the app from the app code

- @app => the app code used to resolve the table

2.1.7 Table to Id

```
function table2id($table)
```

This function resolves the id of the app from the app table

- @table => the app table used to resolve the id

2.1.8 Table to App

```
function table2app($table)
```

This function resolves the code of the app from the app table

- @table => the app table used to resolve the app code

2.1.9 Id to Subtables

```
function id2subtables($id)
```

This function resolves the subtables of the app from the app id

- @id => the app id used to resolve the subtables

2.1.10 App to Subtables

```
function app2subtables($app)
```

This function resolves the subtables of the app from the app code

- @app => the app code used to resolve the subtables

2.1.11 Table to Subtables

```
function table2subtables($table)
```

This function resolves the subtables of the app from the app table

- @table => the app table used to resolve the subtables

2.1.12 App Exists

```
function app_exists($app)
```

This function detect if an app exists

- @app => the app that you want to check if exists

2.1.13 Detect apps files

```
function detect_apps_files($file)
```

This function returns the files found in the main path and in the apps path

- @file => the pattern used to search files

2.2 Array helper module

```
api/php/autoload/array.php
```

This file contain useful array functions

2.2.1 Null to array converter

```
function null2array($arr)
```

This function convert all nulls into an array, is intended to be used as helper for example in the glob output, to force to have an array in all cases

- @arr => the input, generally must to be an array, if a null is passed, then a void array will be returned

2.2.2 Join for array

```
function join4array($array)
```

This function allow to join the #attr and value to get only an associative array, it is intended to be used to simplify the specification of some elements and to simplify the usage of this elements in the client side

- @array => the input that can contains an array with #attr and value

2.3 Array to XML helper module

```
api/php/autoload/array2xml.php
```

This fie is a part of the most old files of the SaltOS, accomplish the task to convert an array to an XML string, currently it is little used because the most important module compared to this is the inverse xml2array module

2.3.1 Check Node Name

```
function __array2xml_check_node_name($name)
```

This function acts as helper of the array2xml function, is intended to return if the node name is valid

- @name => the name that you want to validate

2.3.2 Check Attr Name

```
function __array2xml_check_node_attr($name)
```

This function acts as helper of the array2xml function, is intended to return if the attribute name is valid

- @name => the name that you want to validate

2.3.3 Write Nodes array2xml helper

```
function __array2xml_write_nodes(&$array, $level = null)
```

This function acts as helper of the array2xml function, is intended to return a string with the tree array

- @array => the tree array that you want to convert to XML
- @level => can be null to minify the output zero to indent the XML contents

2.3.4 Array to XML

```
function array2xml($array, $usecache = true, $usexmlminify = true)
```

This function returns a string with the contents of array converted into a XML language file, to do it, uses some helpers as __array2xml_* functions

- @array => the array that contains the tree structure that you want to convert to XML

- @usecache => a boolean to enable or disable the cache usage
- @usexmlminify => a boolean to enable or disable the minify feature

2.4 Barcode helper module

```
api/php/autoload/barcode.php
```

This fie contains useful functions related to barcodes

2.4.1 BarCode function

```
function __barcode($msg, $w, $h, $m, $s, $t)
```

This function allow to generate a barcode, you can pass the desired message that you want to convert in barcode and it returns an image with the data

- @msg => Contents of the barcode
- @w => width of each unit's bar of the barcode
- @h => height of the barcode (without margins and text footer)
- @m => margin of the barcode (white area that surround the barcode)
- @s => size of the footer text, not used if zero
- @t => type of the barcode, C128 is the most common type used

Notes:

The normal behavior is returns a png image, but if something was wrong, the function can returns an empty string

2.5 Captcha helper module

```
api/php/autoload/captcha.php
```

This fie contains useful functions related to captchas

2.5.1 Captcha Is Prime Number

```
function __captcha_isprime($num)
```

This function is a detector of prime numbers, uses some optimizations and ideas from www.polprimos.com, please, see the previous web to understand the speedup of this function in the prime numbers validation

- @num => the number that you want to check if it is a primer numner

Notes:

See www.polprimos.com for understand it

2.5.2 Captcha Image

```
function __captcha_image($code, $args = [])
```

This function returns an image with the code drawed in a background that contains white noise to prevent that robots read the code

- @code => the code that you want to paint
- @width => the width of the generated image
- @height => the height of the generated image
- @letter => the size of the letters of the generated image
- @number => the size of the numbers of the generated image
- @angle => the angle allowed to rotate the letters and numbers
- @color => the color user to paint the code
- @bgcolor => the background color of the image
- @fgcolor => the color used to paint the letters of the background of the image
- @period => parameter for the wave transformation
- @amplitude => parameter for the wave transformation
- @blur => true or false to enable or disable the blur effect

Notes:

The main idea to program this captcha was obtained from this post:

- http://sentidoweb.com/2007/01/03/laboratorio-ejemplo-de-captcha.php

Too appear in ther posts if you search for it in google:

- http://www.google.es/search?q=captcha+alto_linea

2.5.3 Captcha Make Number function

```
function __captcha_make_number($length)
```

This function returns a random number of the desired length and as trick, checks that the output is a prime number

- @length => the length of the desired output string

2.5.4 Captcha Make Math Operation function

```
function __captcha_make_math($length)
```

This function returns a random math operation of the desired length and as trick, checks that the output operation is performed by prime numbers

- @length => the length of the desired output string

2.6 Color utils helper module

```
api/php/autoload/color.php
```

This fie contains useful functions related to colors

2.6.1 Color To Dec function

```
function color2dec($color, $component)
```

This function is a helper that allow to get from a RGB hex color the value in decimal of the specified component, useful to get the amount of color red, green or blue in decimal base from an string

Is able to understand colors with the formats #abcdef, abcdef, #000, #fff

- @color => The color that you want to parse
- @component => The component that you want to retrieve their value

2.7 Compatibility helper module

```
api/php/autoload/compat.php
```

This file add some functions used by SaltOS that can not be found in all allowed versions of PHP

2.7.1 Array Key Last

```
function array_key_last(array $array)
```

This function appear in PHP 7.3, and for previous version SaltOS uses this code

- @array => the array where you want to obtain the last key

Notes:

Code copied from the follow web: https://www.php.net/manual/es/function.array-key-last.php#124007

2.7.2 Array Key First

```
function array_key_first(array $arr)
```

This function appear in PHP 7.3, and for previous version SaltOS uses this code

- @array => the array where you want to obtain the first key

Notes:

Code copied from the follow web: https://www.php.net/manual/es/function.array-key-last.php#124007

2.8   Config helper module

```
api/php/autoload/config.php
```

This fie contains useful functions related to configuration features

2.8.1   Get config

```
function get_config($key, $user_id = -1)
```

This function is intended to be used to retrieve values from the config system, as first level, the function try to get the value from the tbl_config, and if it is not found, then the function try to get the values from the config file.

- @key => the key that you want to retrieve the value
- @user_id => the user_id used in the first search step

Notes:

This function is a new release of the olds getConfig and getDefault, depending of the user_id argument, it tries to search in the config file or in the database, for negative values the function uses the config file and for zero or positive values, tries to search it in the database

2.8.2   Set config

```
function set_config($key, $val, $user_id = -1)
```

This function sets a value to a config key, the data will be stored in the database using the tbl_config for zero or positive values of user_id, and in the memory of the config file for negative user_id values

- @key => the key that you want to set
- @val => the value that you want to set
- @user_id => the user_id used as filter

2.8.3   Detect config files

```
function detect_config_files($file)
```

This function returns the files found in the main path, in the apps path and in the files path

- @file => the pattern used to search files

2.9   Control helper module

```
api/php/autoload/control.php
```

This fie contains useful functions related to the control and version system, they allow to relationate registers with users and groups, and to add and retrieve the versions of a register

2.9.1 Make Control function

```
function make_control($app, $reg_id, $user_id = null, $datetime = null)
```

This function allow to insert and delete the control registers associacted to any application and to any register of the application

- @app => code of the application that you want to index
- @reg_id => register of the app that you want to index
- @user_id => user id of the owner of the app register
- @datetime => time mark used as creation time of the app register

Notes:

You can pass a null user_id and/or null datetime, in these cases, the function will determine the user_id and datetime automatically

This function returns an integer as response about the control action:

- +1 => insert executed, this is because the app register exists and the control register not exists
- +2 => delete executed, this is because the app register not exists and the control register exists
- -1 => app not found, this is because the app requested not have a table in the apps config
- -2 => control table not found, this is because the has_control feature is disabled by dbstatic
- -3 => data not found, this is because the app register not exists and the control register too not exists
- -4 => control exists, this is because the app register exists and the control register too exists

As you can see, negative values denotes an error and positive values denotes a successfully situation

2.9.2 Add Version function

```
function add_version($app, $reg_id, $user_id = null, $datetime = null)
```

This function allow to add a new version to a reg_id of an app, to do it, the function requires to specify the app, reg_id, the original data and the new data to compute the diff patch that must to be stored in the data field and create the register for the new version

To do this, the function validate the input data, checks the existence of registers of data and versions, prepare the data patch to store, get the old hash to do the blockchain, get the last ver_id and compute all needed things to do the insert of the new version register

- @app => code of the application that you want to add a new version
- @reg_id => register of the app that you want to add a new version
- @user_id => user id of the owner of the version register
- @datetime => time mark used as creation time of the version register

Notes:

You can pass a null user_id and/or null datetime, in these cases, the function will determine the user_id and datetime automatically

This function returns an integer as response about the control action:

- +1 => insert executed, this is because the app register exists and they can add a new version register

- +2 => delete executed, this is because the app register not exists and the version register exists

- -1 => app not found, this is because the app requested not have a table in the apps config

- -2 => version table not found, this is because the has_version feature is disabled by dbstatic

- -3 => data not found, this is because the app register not exists and the control register too not exists

As you can see, negative values denotes an error and positive values denotes a successfully situation

2.9.3 Get Version

```
function get_version($app, $reg_id, $ver_id)
```

This function allow to get an specific version of a register and app, intended to get the data used in a specific version to compare with other versions and to restore data to the requested version

- @app => code of the application that you want to add a new version

- @reg_id => register of the app that you want to add a new version

- @ver_id => the version that you want to get

Notes:

This function is not a simple select of the register that matches with the ver_id requested, it does an accumulative merge to get the register data in the moment where the version will be stored, to do it, they must to restore versions from 1 to ver_id, and must to discard the next versions

2.10   Database helper module

```
api/php/autoload/database.php
```

This fie contains useful functions related to database, allow to connect, check queries, execute queries, disconnect, retrieve rows and manipulate resultsets of the database

2.10.1   DB Connect

```
function db_connect($args = null)
```

This function is intended to stablish the connection to the database

- @args => is an array with key val pairs

- @host => the host for the connection

- @port => the port used for the connection

- @name => name of the database for the connection

- @user => user used to stablish the connection

- @pass => pass used to stablish the connection

- @file => the file that contains the database

Notes:

The parameters can be different depending of each database driver, in general the arguments can be the host, port, name, user and pass for database's servers and only file for the sqlite database

If the args argument is null, the the function try to use the configuration stored in the config file and depending of the argument presense, it will return the database object or store it in the config to be used by the nexts functions of this php file

2.10.2 DB Check

```
function db_check($query)
```

This function is intended to check that the query execution will not trigger an error

- @query => the query that you want to validate

2.10.3 DB Query

```
function db_query($query, $fetch = "query")
```

This public function is intended to execute the query and returns the resultset

- @query => the query that you want to execute
- @fetch => the type of fetch that you want to use, can be auto, query, column or concat

Notes:

The fetch argument can perform an speed up in the execution of the retrieve action, and can modify how the result is returned

auto: this fetch method try to detect if the resultset contains one or more columns, and sets the fetch to column (if the resultset only contains one column) or to query (otherwise)

query: this fetch method returns all resultset as an array of rows, and each row contain the pair of key val with the name of the field and the value of the field

column: this fetch method returns an array where each element is each value of the field of the each row, this is useful when for example do you want to get all ids of a query, with this method you can obtain an array with each value of the array is an id of the resultset

concat: this fetch method is an special mode intended to speed up the retrieve of large arrays, this is useful when you want to get all ids of a query and you want to get a big sized array, in this case, is more efficient to get an string separated by commas with all ids instead of an array where each element is an id

2.10.4 DB Disconnect

```
function db_disconnect()
```

This function close the database connection and sets the link to null

2.10.5 DB Fetch Row

```
function db_fetch_row(&$result)
```

This function returns the next row of the resultset queue

- @result => this argument is passed by reference and contains the resultset queue obtained by the db_query

2.10.6 DB Fetch All

```
function db_fetch_all(&$result)
```

This function returns all rows of the resultset queue

- @result => this argument is passed by reference and contains the resultset queue obtained by the db_query

2.10.7 DB Num Rows

```
function db_num_rows($result)
```

This function returns the total number of the results in the resultset queue

- @result => this argument is passed by reference and contains the resultset queue obtained by the db_query

2.10.8 DB Free

```
function db_free(&$result)
```

This function releases all memory used by the resultset queue

- @result => this argument is passed by reference and contains the resultset queue obtained by the db_query

2.10.9 DB Num Fields

```
function db_num_fields($result)
```

This function returns the number of fields of the resultset queue

- @result => this argument is passed by reference and contains the resultset queue obtained by the db_query

2.10.10 DB Field Name

```
function db_field_name($result, $index)
```

This function returns the name of the field identified by the index field

- @result => this argument is passed by reference and contains the resultset queue obtained by the db_query

2.11 Datetime helper module

```
api/php/autoload/datetime.php
```

This fie contains useful functions related to datetime, allow to get current timestamp in various formats, allow to evaluate timestamps and some helper more

2.11.1 Current Date

```
function current_date($offset = 0)
```

This function returns the current date in a YYYY-MM-DD format, this is used by a lot of functions in SaltOS, allow to specify a bias used to move the current time mark

- @offset => the bias added to the current time mark

2.11.2 Current Time

```
function current_time($offset = 0)
```

This function returns the current time in a HH:II:SS format, this is used by a lot of functions in SaltOS, allow to specify a bias used to move the current time mark

- @offset => the bias added to the current time mark

2.11.3 Current DateTime

```
function current_datetime($offset = 0)
```

This function returns the current date and time in a YYYY-MM-SS HH:II:SS format, this is used by a lot of functions in SaltOS, allow to specify a bias used to move the current time mark

- @offset => the bias added to the current time mark

2.11.4 Current Decimals

```
function current_decimals($offset = 0, $size = 4)
```

This function returns the current decimals to be added to the seconds as a decimal part, this function uses the microtime function to get this level of precision that can not be obtained using the original date and time functions

- @offset => the bias added to the current time mark
- @size => the size of the returned decimal part

Notes:

This function is used by current_datetime_decimals, and don't have more uses that provice more precision in the logs files

2.11.5 Current DateTime Decimals

```
function current_datetime_decimals($offset = 0, $size = 4)
```

This function returns the current date and time with decimals in the seconds in a YYYY-MM-DD HH:II:SS.XXXX format, useful when do you want to log information more accuracy to debug issues, for example

- @offset => the bias added to the current time mark
- @size => the size used by the decimal part

2.11.6 Dateval

```
function dateval($value)
```

This function try to do the same thing that intval or strval, but for date values, to do this, this function try to separate all elements and identify the year position and the other elements, the result will be of the format YYYY-MM-DD

- @value => the input value to validate

Notes:

This function try to cast the year, month and day from 0000-00-00 to valid values, this is because the databases accepts the 0000-00-00 date and is used as emulated null, the month are limited to 12 and the day is limited to the days of the month and year, this is useful because the dates that are more greather that zero, will have a valid and an existing value

2.11.7 Day of a Month helper

```
function __days_of_a_month($year, $month)
```

This function is a helper used by other date and datetime functions, this is useful because allow to fix problems in dates that use days out of range

- @year => year that you want to use in the validation
- @month => month that you want to use in the validation

2.11.8 Timeval

```
function timeval($value)
```

This function try to do the same thing that intval or strval, but for time values, to do this, this function try to separate all elements and identify the elements, the result will be of the format HH:II:SS

- @value => the input value to validate

2.11.9 Datetimeval

```
function datetimeval($value)
```

This function try to do the same thing that intval or strval, but for datetime values, to do this, this function try to separate all elements and identify the year position and the other elements, the result will be of the format YYYY-MM-DD HH:II:SS

- @value => the input value to validate

Notes:

This function try to cast the year, month and day from 0000-00-00 to valid values, this is because the databases accepts the 0000-00-00 date and is used as emulated null, the month are limited to 12 and the day is limited to the days of the month and year, this is useful because the dates that are more greather that zero, will have a valid and an existing value

2.11.10 Time to Seconds

```
function __time2secs($time)
```

This function converts the time format into seconds

- @time => time to be converted into seconds, the format will be HH:II:SS

2.11.11 Seconds to Time

```
function __secs2time($secs)
```

This function converts the seconds into time format

- @secs => seconds to be converted to time format, the format will be a number

2.11.12 Current Day Of Week

```
function current_dow($offset = 0)
```

This function returns the current day of week as integer between 1 and 7 range, this is used by some functions in SaltOS, allow to specify a bias used to move the current time mark

- @offset => the bias added to the current time mark

2.12 Database schema helper module

```
api/php/autoload/dbschema.php
```

This fie contains useful functions related to database schema, allow to manage the entire database schema, and too, allow to maintain contents of some tables using the dbstatic feature

2.12.1 DB Schema

```
function db_schema()
```

This function try to maintain the database structure, to do it, this feature uses the dbschema.xml file to store the database structure.

2.12.2 DB Static

```
function db_static()
```

This function try to maintain the database contents, to do it, this feature uses the dbstatic.xml file to store the database contents that must to be maintaned.

This version of the db_static allow you to use a comma separated values in fields as "id", start by "id_" or end by "_id"

2.12.3 DB Static insert

```
function __dbstatic_insert($table, $row)
```

This function is a helper of previous function, is intended to be used by db_static and allow to use a comma separated values in fields as "id", start by "id_" or end by "_id"

- @table => the table that you want to use in the insert process
- @row => the row that you want to add in the table

Notes:

This feature allow to you to use comma separated lists of values, commonly used for id fields as user_id, perms_id, or similar.

2.12.4 Get Tables from DB Schema

```
function get_tables_from_dbschema()
```

This function returns the tables from the DB Schema file

2.12.5 Get Fields from DB Schema

```
function get_fields_from_dbschema($table)
```

This function returns the fields from the DB Schema file

- @table => the table that you want to request the fields

2.12.6 Get Indexes from DB Schema

```
function get_indexes_from_dbschema($table)
```

This function returns the indexes from the DB Schema file

- @table => the table that you want to request the indexes

2.12.7 Get Ignores from DB Schema

```
function get_ignores_from_dbschema()
```

This function returns the ignores tables from the DB Schema file

2.12.8 Get Fulltext from DB Schema

```
function get_fulltext_from_dbschema()
```

This function returns the fulltext tables from the DB Schema file

2.12.9 Get Fkeys from DB Schema

```
function get_fkeys_from_dbschema($table)
```

This function returns the fkeys from the DB Schema file

- @table => the table that you want to request the fkeys

2.12.10 DB Schema helper

```
function __dbschema_helper($fn, $table)
```

This function is a helper for the previous functions, is intended to be used to returns the tables of the DB Schema or the fields of a table

- @fn => the caller function name
- @table => the table used by some features

2.12.11 DB Schema Auto Apps

```
function __dbschema_auto_apps($dbschema)
```

This function is a helper to the dbschema functions, to create an indexing table for each app

- @dbschema => the dbschema array

Notes:

This feature creates a table and try to use Mroonga storage engine with one field, the main idea of this tables is to store all contents of the register to do quick searchs using a fulltext search engine

2.12.12 DB Schema Auto Fkey

```
function __dbschema_auto_fkey($dbschema)
```

This function is a helper to the dbschema functions, to create an index for each fkey

- @dbschema => the dbschema array

Notes:

By default, MariaDB creates an index for each foreign key, but SQLite not does is by default and for this reason, SaltOS creates an index automatically, to improve the performance

This function checks that the field not exists in the defined indexes to prevent error in duplicates indexes

2.12.13 DB Schema Auto Name

```
function __dbschema_auto_name($dbschema)
```

This function is a helper to the dbschema functions, to auto name the indexes

- @dbschema => the dbschema array

Notes:

This function allow to specify indexes only specifying the fields that you want to conform the index, but the engines as MariaDB and SQLite, requires that each index have a unique name, and for this reason, we add this feature to automate this part of the process

You can see how the name of the index is different for MySQL and SQLite, this is because in MySQL, the name can be repeated in different tables, but in SQLite, the name must be unique in the database

2.12.14 Get Apps From DBStatic

```
function get_apps_from_dbstatic()
```

This function returns the list of apps that have a table and field defined in the dbstatic file

2.12.15 Get Tables From DBStatic

```
function get_tables_from_dbstatic()
```

This function returns the list of tables that have a table and field defined in the dbstatic file

2.12.16 Get Field From DBStatic

```
function get_field_from_dbstatic($table, $field = "field")
```

This function return the field associated to the table in the dbstatic file and associated to the apps table

- @table => the table of the dbstatic that want to convert to field

Notes:

This function uses the special feature in the helper that allow to use as table parameter an app code to retrieve the field, this is useful if you want some field of the app table and you want to use the app code instead of the app table to identify what row do you want to use

2.12.17 DB Static helper

```
function __dbstatic_helper($fn, $table, $field)
```

This function is intended to act as helper of the dbstatic ecosystem, this function can return the apps that contain table and field definitions and too, can return the field associated to a apps table, useful for the indexing feature

- @fn => the caller function name
- @table => the table used by some features

2.13 Error helper module

`api/php/autoload/error.php`

This fie contains useful functions related to error management, allow to trigger and manage errors, too contains the code used for the error and exception handlers

2.13.1 Show PHP Error

`function show_php_error($array)`

This function allow to SaltOS to trigger the errors in a multiple levels:

1) Check if the error is caused by a memory allocation error, and in this case, try to setup more memory to continue executing the error function, sometimes if the system is using all memory, this function can not do all tasks and don't know whats be happening

2) Add some extra traces if they are not found in the input array

3) Create a human readable message in text and json format, the text will be used to log the error using a regular file and the json will be used as stdout response

4) If the error is caused by a deprecation, the error will be logged in the log file but the execution of the code will continue (if it can continue!!!)

5) Try to categorize the error and log the text in the specific log file, this part is optimized to prevent the addition of repeated errors using a hash as a trick

6) Send a json to the stdout using the output handler.

THs input @array can contain pairs of key val:

- @dberror => The text used in the DB Error section
- @phperror => The text used in the PHP Error section
- @xmlerror => The text used in the XML Error section
- @jserror => The text used in the JS Error section
- @dbwarning => The text used in the DB Warning section
- @phpwarning => The text used in the PHP Warning section
- @xmlwarning => The text used in the XML Warning section
- @jswarning => The text used in the JS Warning section
- @source => The text used in the Source section
- @exception => The text used in the Exception section
- @details => The text used in the Details section
- @query => The text used in the Query section
- @backtrace => The text used in the Backtrace section
- @debug => The text used in the Debug section

Notes:

The unset for the pid and the time keys of the debug array is justificate because each execution modify the

pid and the time entries and break the optimization of the hash with the checklog to prevent repetitions in the log file

2.13.2 Do Message Error

```
function do_message_error($array)
```

This function acts as a helper of the show_php_error, is intended to build the text and the json messages used to the log file and for the stdout channel

THs input @array can contain pairs of key val:

- @dberror => The text used in the DB Error section
- @phperror => The text used in the PHP Error section
- @xmlerror => The text used in the XML Error section
- @jserror => The text used in the JS Error section
- @dbwarning => The text used in the DB Warning section
- @phpwarning => The text used in the PHP Warning section
- @xmlwarning => The text used in the XML Warning section
- @jswarning => The text used in the JS Warning section
- @source => The text used in the Source section
- @exception => The text used in the Exception section
- @details => The text used in the Details section
- @query => The text used in the Query section
- @backtrace => The text used in the Backtrace section
- @debug => The text used in the Debug section

Returns an array with the text and the json formated output ready to be used in the log file and in the stdout channel

2.13.3 Program Handlers

```
function program_handlers()
```

This function program all error handlers

2.13.4 Error Handler

```
function __error_handler($type, $message, $file, $line)
```

This function is the callback function used by the set_error_handler

Ths arguments are defined by the set_error_handler:

- @type => The code of the error
- @message => The descriptive message of the error

- @file => The filename of the file that trigger the error

- @line => The line where the error will occurred

2.13.5   Exception Handler

```
function __exception_handler($e)
```

This function is the callback function used by the set_exception_handler

Ths arguments are defined by the set_exception_handler:

- @e => object that contains the getMessage, getCode, getFile, getLine and getTrace methods

2.13.6   Shutdown Handler

```
function __shutdown_handler()
```

This function is the callback function used by the register_shutdown_function, try to detect if an error is the cause of the shutdown of the script, note that a correct execution will execute this function and only it must to trigger an error if a real error is in the stack of the errors events, to do it this function uses the error_get_last to check if the value in in the list of typified errors

2.13.7   Get code from trace

```
function __get_code_from_trace($trace = null, $index = 0)
```

This function acts as helper of the show_json_error, and try to get the filename and the line where the error will be triggered, for example, an error triggered from the index.php at line 23 will generate a code index:23, this information will be useful for our technical service to help the users when live issues with our API

- @trace => the array returned by the debug_backtrace function

- @index => the position of the array used to get the filename and the line

2.13.8   Show JSON Error

```
function show_json_error($msg)
```

This function is triggered from the code in a controlate errors, the idea is to have a simple way to send controled errors to the user using a json output channel, and to do it, we have this function that can be called with a simple message and the code is created automatically to help the backtrace of the issues

- @msg => this contains a simple text that is used in the json output

2.13.9   Get code from file and line

```
function __get_code_from_file_and_line($file, $line)
```

This function returns the string that contains the PATHINFO_FILENAME and the line to idenfify the launcher of an error, for example

- @file => filename used to obtain the first part of the code
- @line => line used to construct the last part of the code

2.13.10 Show JSON Array

```
function show_json_array($array)
```

This function is intended to be used when you need to generate a json response based on an array contents, too, this function adds the code entry to the input array

- @array => The input array used in the output_handler_json without the code entry

2.13.11 Detect Recursion

```
function detect_recursion($fn)
```

This function allow to SaltOS to detect the recursión, to do it, uses the debug_backtrace function that returns all information about the execution of the current function, the main idea of this function is to detect in what lines of the backtrace appear the file or the function, and returns the count of times that appear

- @fn => the name of the function or file, can be multiples functions or files separated by a comma

2.14 Execution helper module

```
api/php/autoload/exec.php
```

This fie contains useful functions related to execution of external programs, allow to execute, check commands and manage some features as the cache usage or the timeout used in each execution

2.14.1 OB Passthru

```
function ob_passthru($cmd, $expires = 0)
```

This function is a join of ob_start and passthru, the main idea is to execute a program using the command line and get the output (only stdout and not stderr) and return the data

- @cmd => the command line that you want to execute
- @expires => the expires time used to compute if the cache is valid

This funtion tries to execute the command using some of the provided methods, passthru, system, exec and shell_exec, another feature is that the function detects what command are available to use

2.14.2 Check Commands

```
function check_commands($commands, $expires = 0)
```

This function tries to validate if the commands are available in the system, to do it, uses the unix command witch

- @commands => the commands that you want to check if are they available

- @expires => the expires time used to compute if the cache is valid

2.14.3 Is Disabled Function

```
function is_disabled_function($fn = "")
```

This function check if the argument contains a disabled function, this feature uses the variables disable_functions and suhosin.executor.func.blacklist to get the list of all disabled functions

- @fn => the function that you want to check if is it disabled

2.14.4 Exec Timeout

```
function __exec_timeout($cmd)
```

This helper function allow to execute commands using the external command timeout, this unix command allow to define the timeout for an execution of other command, and when the timeout is reached, then break the execution killing the process

- @cmd => the command that you want to execute with a timeout control

Returns the string that contains the command with ths timeout control

2.15 Export helper module

```
api/php/autoload/export.php
```

This fie contains useful functions related to export data, allow to generate outputs in formats suck as excel, csv, edi, json and xml

2.15.1 Export File main function

```
function export_file($args)
```

This function is intended to export data in the supported formats

- @type => can be xml, csv, xls, xlsx, edi or json

- @data => the matrix to export

- @sep => separator char used only by csv format

- @eol => enf of line char used by csv and xml format

- @encoding => charset used by csv and xml format

- @replace => array with two elements, from and to, used to do replacements of the matrix values

- @escape => array with two elements, char and mode, used to specify the escape character and the escape mode

- @title => title used only by excel format

- @file => local filename used to store the results

- @ext => extension used for the filename if provided

- @wrap => boolean argument used only for edi indentation

- @indent => boolean argument used only for json indentation

If file argument is specified, void string is returned If file argument is not specified, then they will returns all data

2.15.2 Export File XML

```
function __export_file_xml($matrix, $eol = "\n", $encoding = "UTF-8")
```

This function is intended to export data in xml format

- @matrix => the matrix to export

- @eol => enf of line char

- @encoding => charset used

They will returns all data

2.15.3 Export File CSV

```
function __export_file_csv(
```

This function is intended to export data in csv format

- @matrix => the matrix to export

- @sep => separator char

- @eol => enf of line char

- @encoding => charset used

- @replace => array with two elements, from and to, used to do replacements of the matrix values

- @escape => array with two elements, char and mode, used to specify the escape character and the escape mode

They will returns all data

2.15.4 Export File Excel

```
function __export_file_excel($matrix, $title = "", $type = "Xlsx")
```

This function is intended to export data in excel format

- @matrix => the matrix to export

- @title => title used in the excel file

- @type => can be Xls or Xlsx

They will returns all data

2.15.5 Export File JSON

```
function __export_file_json($matrix, $indent = false)
```

This function is intended to export data in json format

- @matrix => the matrix to export
- @indent => boolean argument to enable or disable the indent feature

They will returns all data

2.16 File utils helper module

```
api/php/autoload/file.php
```

This fie contains useful functions related to the file usage, allow to manage directories, files, caches, paths, permissions, remote files and more

2.16.1 Get Directory

```
function get_directory($key)
```

This function returns the directory configured to the key requested, too can define a default value, useful when the configuration still not loaded and SaltOS need some directory to do something as store data in the log file, for example

- @key => the key used in get_config to request the configured directory

2.16.2 Get Temp File

```
function get_temp_file($ext = "")
```

This function is intended to get a unique temporary file, used for temporary purposes as put contents to be used as input in a command

- @ext => the extension of the temporary file, useful for some commands that try to detect the contents using the extension

Notes:

This function uses the dirs/tempdir config key

2.16.3 Get Cache File

```
function get_cache_file($data, $ext = "")
```

This function is intended to get a cache filename, used for caching purposes

- @data => data used to compute the hash used by the cache, can be an string or an array with lot of contents
- @ext => extension of the cache filename

Notes:

This function uses the dirs/cachedir config key

2.16.4 Cache Exists

```
function cache_exists($cache, $files)
```

This function check the existence of valid cache by comparing the timestamp of the filemtime between the cache file and all files of the second argument

- @cache => cache filename
- @files => array of files that are considered as dependencies of the cache

2.16.5 URL Get Contents

```
function url_get_contents($url)
```

This file is an equivalent of the file_get_contents but intended to be used for request remote files using protocols as http or https

- @url => the url that you want to retrieve

Notes:

This function only returns the body of the request, if you are interested to get the headers of the request, try to use the __url_get_contents

2.16.6 URL Get Contents helper

```
function __url_get_contents($url, $args = [])
```

This file is an equivalent of the file_get_contents but intended to be used for request remote files using protocols as http or https

- @url => the url that you want to retrieve
- @args => Array of arguments, explained in the follow lines
- @cookies => an array with the cookies to be restored before send the request
- @method => method used in the request
- @values => an array with the post values, useful when you want to send a POST request with pairs of variables and values
- @referer => the referer string
- @headers => an array with the headers to be send in the request
- @body => the full body used of the request, useful when you want to send a json file in the body instead of pairs of keys vals

This function returns an array with three elements, body, headers and cookies

Notes:

This function uses the httpclient library

2.16.7 Extension

```
function extension($file)
```

This function returns the PATHINFO_EXTENSION of the file

- @file => file used in the pathinfo call

2.16.8 Encode Bar Chars File

```
function encode_bad_chars_file($file)
```

This function is equivalent to encode_bad_chars but intended to be used with filenames, in this case, the extension and the rest of the filename will be encoded separately and the return value will contain the dot separating the filename with the extension

- @file => filename used in the encode process

2.16.9 Realpath Protected

```
function realpath_protected($path)
```

This function returns the realpath of the path, this version of the function allow to return the path of an unexistent file, this is useful when do you want to get the realpath of a unexistent file, for example, to the output of a command that must to generate the file but at the moment of the execution of this function the file is not found

- @path => path used in the realpath call

2.16.10 Getcwd Protected

```
function getcwd_protected()
```

This function returns the same result that the getcwd function but checking that the result is not an slash, this is an issue in some cases caused by permissions problems, and a good solution for this cases is to get the directory of the script as current work directory

2.16.11 Glob Protected

```
function glob_protected($pattern)
```

This function returns the same result that the glob function but checking that the result is an array, if glob fails or not get a files by the pattern, can return other values that an array, and this can cause problems if you are expecting an array to iterate in each element, this function prevent this problem

- @pattern => pattern used in the glob command

2.16.12 Chmod Protected

```
function chmod_protected($file, $mode)
```

This function tries to change the mode of the file using the chmod function only if the fileperms of the file are different that the requested mode

- @file => file used by the chmod function

- @mode => mode used by the chmod function

2.16.13 Fsockopen Protected

```
function fsockopen_protected($hostname, $port, &$errno = 0, &$errstr = "",
    $timeout = null)
```

This function is only used by the httpclient library to avois problems with the certificates validations

Ths arguments is the same that the fsockopen function, in this case, the function uses the stream_socket_client to emulate the original fsockopen

2.16.14 File with Mtime

```
function file_with_hash($file)
```

This function returns the name of the file adding as argument the hash of the file for the http/https requests, this allow to helps the browser to know when the file has changed

- @file => the file that you want to add the hash querystring argument

2.17 Garbage collector helper module

```
api/php/autoload/gc.php
```

This fie contains useful functions related to the garbaging unused resources, currently only implements the clear of temporary files

2.17.1 Garbage Collector Executor

```
function gc_exec()
```

This function tries to clean the cache, temporary and upload directories of old files, the parameters that this function uses are defined in the config file, uses three directories (cachedir, tempdir and uploaddir) and the timeout is getted from the server/cachetimeout config file key, too is able to detect hidden files and remove except the special files as current directory, parent directory and htaccess file

2.18 Get data helper module

```
api/php/autoload/getdata.php
```

This fie contains useful functions related to the $_DATA global variable, allow to get and set values in the global $_DATA variable using xpath as key

2.18.1   Get data

```
function get_data($key)
```

This function is intended to be used to retrieve values from the data system

- @key => the key that you want to retrieve the value

2.18.2   Set data

```
function set_data($key, $val)
```

This function sets a value in the data system for the specified key

- @key => the key that you want to set
- @val => the value that you want to set

2.19   Gettext helper module

```
api/php/autoload/gettext.php
```

This fie contains useful functions related to gettext funcionality, allow to manage the SaltOS translations using a merged system of the unix locales and the old SaltOS translations system.

2.19.1   Get Text function

```
function T($text)
```

This function replaces the gettext abreviation _() using the SaltOS gettext feature, is based in the original system of the SaltOS 3 with improvements to do more open as the GNU gettext

2.20   Html helper module

```
api/php/autoload/html.php
```

This file contain useful html helper functions

2.20.1   Remove Script Tag

```
function remove_script_tag($temp)
```

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

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

2.20.2 Remove Style Tag

```
function remove_style_tag($temp)
```

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

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

2.20.3 TODO

```
function href_replace($temp)
```

TODO

2.21 Import file helper module

```
api/php/autoload/import.php
```

This fie contains useful functions related to import contents using differents formats suck as excel, csv, edi, json, xml and bytes. Too this module allow to manipulate data using the tree array of the core of the SaltOS, allowing to add, modify and remove nodes, too can apply patch of the memory data and paint arrays as ascii tables

2.21.1 Import File main function

```
function import_file($args)
```

This function is intended to import data in the supported formats

- @data => contents used as data instead of file
- @file => local filename used to load the data
- @type => can be xml, csv, xls, bytes, edi or json
- @sep => separator char used only by csv format
- @sheet => sheet that must to be read
- @map => map used as dictionary for each field, pos and length
- @offset => the offset added to the start position in each map field
- @nomb => boolean to disable or enable the multibyte support
- @novoid => boolean to enable or disable the removevoid feature
- @prefn => function executed between the load and the tree construction
- @notree => boolean to enable or disable the array2tree feature
- @nodes => an array with the fields that define each nodes used in the tree construction
- @nohead => if the first row doesn't contains the header of the data, put this field to one
- @noletter => if you want to use numeric index instead of excel index, put this field to one
- @postfn => function executed after the tree construction

This function returns an array with the loaded data from file Can return a matrix or tree, depending the nodes parameter

2.21.2 UTF8 BOM helper

```
function __import_utf8bom($data)
```

This function remove the bom header of the string

- @data => the data that must to be checked

Returns the data without the bom characters

2.21.3 XML to Array

```
function __import_xml2array($file)
```

This function convert an xml into an array

- @file => the file that contains the xml

Returns an array with the contents of the xml

2.21.4 Struct to Array helper

```
function __import_struct2array(&$data)
```

This function is a helper of the __import_xml2array

- @data => an array with all nodes of the xml file

Returns an array with the correct structure that matches the xml structure

2.21.5 Get Node helper

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

This function is a helper used to get a node in a xml structure

- @path => a path of the desired node
- @array => the array with nodes of the xml structure

Returns the contents of the node of the specified path

2.21.6 Get Value helper

```
function __import_getvalue($array)
```

This function is a helper used to get a value if exists of a node structure

- @array => an array

Retusn the value if exists, otherwise the same input

2.21.7 Get Attr helper

```
function __import_getattr($elem, $array)
```

This function is a helper used to get a attr element if exists of a node structure

- @elem => a string representing an element
- @array => an array containing the node

Returns the attr if exists, otherwise null

2.21.8 Set Node helper

```
function __import_setnode($path, &$array, $value)
```

This function is used to set data into a xml structure

- @path => the desired path where do you want to put the data,
- @array => the array with the xml structure
- @value => the value that do you want to put

Returns true if the function can set the value, false otherwise

2.21.9 Del Node helper

```
function __import_delnode($path, &$array)
```

This function is used to remove data of the xml structure

- @path => the desired path where do you want to remove
- @array => the array with the xml structure

Returns true if the function can remove the path, false otherwise

2.21.10 Add Node helper

```
function __import_addnode($path, &$array, $value)
```

This function is used to add data into a xml structure

- @path => the desired path where do you want to add the data
- @array => the array with the xml structure
- @value => the value that do you want to add

true if the function can add the data, false otherwise

2.21.11 Special Chars helper

```
function __import_specialchars($arg)
```

This function is a helper used by the csv2array function

- @arg => a string or array

Returns the input with the expected replacements

2.21.12 CSV to Array helper

```
function __import_csv2array($file, $sep)
```

This function is a helper of the __import_xml2array

- @file => the filename and the sheet that do you want to retrieve
- @sep => the separator field used in the csv file

Returns a matrix with the contents

2.21.13 XLS to Array helper

```
function __import_xls2array($file, $sheet)
```

This fuction can convert an excel file into a matrix structure, it has some additional features as:

- If the file exceds the 1Mbyte and the server has the xlsx2csv executable, it tries to convert the xslx to an excel to use less memory
- Do some internals trics to solve some knowed issues
- @file => the filename and the sheet that do you want to retrieve
- @sheet => the second parameter can be a number or a sheet name

Returns a matrix with the contents

2.21.14 Bytes to Array helper

```
function __import_bytes2array($file, $map, $offset, $nomb)
```

This function can read files as blocks of bytes, they can use a map, can specify an offset and can be used using multibyte if it is needed

- @file => local filename used to load the data
- @map => map used as dictionary for each field, pos and length
- @offset => the offset added to the start position in each map field
- @nomb => boolean to disable or enable the multibyte support

Returns a matrix with the contents

Notes:

The map must be an array of strings of the follow form: ["field1;0;10", "field2;10;20", "field3;20;40"]

2.21.15 Edit to Array helper

```
function __import_edi2array($file)
```

This fuction can convert an excel file into a tree structure

- @file => local filename used to load the data

2.21.16 JSON to Array helper

```
function __import_json2array($file)
```

This fuction can convert an excel file into a tree structure

- @file => local filename used to load the data

2.21.17 Check Real Matrix helper

```
function __import_check_real_matrix($array)
```

This function checks that the argument is a matrix, to do this, checks that the argument is an array, that all keys are numeric and that all entries of the main array is another array, and for each another array, checks that the keys are numeric and that all values are non arrays

- @array => the array to check

2.21.18 Remove Void helper

```
function __import_removevoid($array)
```

This function is able to remove an entire row or column if it is void

- @array => the array to fix

2.21.19 Array to Tree helper

```
function __import_array2tree($array, $nodes, $nohead, $noletter)
```

This function tries to convert the array into a tree using the nodes, specification

- @array => the matrix that you want to convert into a tree
- @nodes => the dictionary used to the conversion, must to be an array with the fields used by each node, for example ["A,B,C","D,E,F"]
- @nohead => set it to true to prevent the usage of the first row of the matrix as header, this option uses the letter as id of each element of the tree
- @noletter => set it to true to prevent the usage of letters, if the previous option is set to true

2.21.20 Array Intersect

```
function __import_array_intersect($data, $filter)
```

This function returns the same result that array_intersect_key($data,array_flip($filter)) maintaining the order of the filter array.

- @data => the array that you want to apply the filter
- @filter => the array where obtain the keys to apply the filter

2.21.21 Array to Tree Set helper

```
function __import_array2tree_set(&$result, $parts)
```

This function tries to set values in a tree structure, to do it, it uses the parts array that contains a list of paired keys and values used to move by the tree setting the values of each pair of key val

- @result => the array where do you want to put the parts
- @parts => an array with pairs of key val

2.21.22 Array to Tree Clean helper

```
function __import_array2tree_clean($array)
```

This function tries to clean the tree by setting an automatic indexes

- @array => the array to clean

2.21.23 Tree to Array

```
function __import_tree2array($array)
```

This function convert a tree into a 2D matrix, it's intended to convert a tree structure into a csv, for example

- @array => the tree that you want to convert into a 2D matrix

2.21.24 Column to Name helper

```
function __import_col2name($n)
```

This function returns the name of the column from the position n

- @n => the position number

Notes:

This function was copied from:

- http://www.php.net/manual/en/function.base-convert.php#94874

2.21.25 Name to Column helper

```
function __import_name2col($a)
```

This function returns the position number of the column from the name

- @a => the column name

Notes:

This function was copied from:

- http://www.php.net/manual/en/function.base-convert.php#94874

2.21.26 Is Name helper

```
function __import_isname($name)
```

This function returns true if the name argument contains only valid letters used in the name of the column

- @name => the name that you want to check

2.21.27 Cell to Column and Row helper

```
function __import_cell2colrow($cell)
```

This function extract the column part and the row part from a cell name

- @cell => the cell that you want to process

Notes:

This function tries to retusn an array with two elements, for example, for the cell AX23, the function returns [AX,23]

2.21.28 Get Keys

```
function __import_getkeys($array)
```

This function tries to return an array with all the keys used internally in the tree

- @array => the tree array that you want to process

2.21.29 Filter helper

```
function __import_filter($array, $filter, $eval = 0)
```

This function tries to apply a filter to a tree array, too allow to use the evaluation system to allow to pass as filter an expression like this A=M23

- @array => the tree array that you want to apply the filter
- @filter => the filter to apply
- @eval => set to 1 if you want to enable the eval feature

2.21.30 Filter Recursive helper

```
function __import_filter_rec($node, $filter, $eval, $parent = [])
```

This function is a helper of the previous function and is able to to the same but with recursivity

- @node => the tree node that you want to filter
- @filter => the filter to apply
- @eval => set to 1 if you want to enable the eval feature
- @parent => this parameter is intended to be used internaly by the function

2.21.31 Apply Patch

```
function __import_apply_patch(&$array, $key, $val)
```

This function is able to apply a patch in the tree array, this allow to update the desired branch of the tree using a xpath notation

- @array => the array that you want to apply the patch
- @key => the xpath where you want to apply the patch
- @val => the val that you want to put in the desired xpath

2.21.32 Apply Patch Recursive helper

```
function __import_apply_patch_rec(&$array, $key, $val)
```

This function is a helper of the previous function and is able to to the same but with recursivity

- @array => the array that you want to apply the patch
- @key => the xpath where you want to apply the patch
- @val => the val that you want to put in the desired xpath

2.21.33 Make Table ASCII

```
function __import_make_table_ascii($array)
```

This table is intended for debug purposes and is able to paint in ascii mode the contents of a matrix

- @rows => the contents of the matrix to paint
- @head => set to true if you want to use the first row as header
- @compact => set to true if you want to minify the ascii table

2.22 Make index helper module

```
api/php/autoload/indexing.php
```

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

2.22.1   Make Index main function

```
function make_index($app, $reg_id)
```

This function implements the make index feature of SaltOS, this consists in a concatenation of fields and subqueries to retrieve all data related to the tables involved in the desired application and the register reg_id

- @app => code of the application that you want to index
- @reg_id => register of the app that you want to index

Notes:

This function returns an integer as response about the index action:

- +1 => insert executed, this is because the app register exists and the index register not exists
- +2 => update executed, this is because the app register exists and the index register too exists
- +3 => delete executed, this is because the app register not exists and the index register exists
- -1 => app not found, this is because the app requested not have a table in the apps config
- -2 => index table not found, this is because the has_index feature is disabled by dbstatic
- -3 => data not found, this is because the app register not exists and the indexting register too not exists

As you can see, negative values denotes an error and positive values denotes a successfully situation

2.22.2   Make Index helper

```
function __make_index_helper($table, $id = "")
```

This function allow the make_index to retrieve all data of the fiels and all data of the related fields of the related tables, this is done by using the fkey information of the dbschema, this function uses some features of the dbschema functions to get the fields, types, fkeys and too, the dbstatic information of the app table

This function uses a cache technique to improve the performance, returns an array with all fields and subqueries to allow to retrieve all data related to the app register

2.23   Iniset helper module

```
api/php/autoload/iniset.php
```

This fie contains useful functions related to the evaluation of the iniset, puntenv and extra directives configures in the config.xml file

2.23.1   Eval Iniset

```
function eval_iniset($array)
```

This function evaluates the iniset section of the config file, is intended to execute all ini_set commands detecting the current values and determining if is needed to change or not the current setting, is able to understand boolean values as On/Off, and too is able to set keys as mbstring.internal_encoding or mbstring.detect_order that must to be set by using another mb_* functions

- @array => the array with the pairs of keys vals

2.23.2   Eval Putenv

```
function eval_putenv($array)
```

This function evaluates the putenv section of the config file, is intended to execute all putenv commands detecting the current values and determining if is needed to change or not the current setting

- @array => the array with the pairs of keys vals

2.23.3   Eval Extras

```
function eval_extras($array)
```

This function evaluates the extra init requirements, intended for the multibyte functions and for the gettext initialization process

- @array => the array with the pairs of keys vals

Notes:

If the val of the pair key val is an array, then each value of the array is used as an argument for the key function, only is intended at the moment to accept functions with one and two arguments, otherwise an error is triggered

2.24   Log helper module

```
api/php/autoload/log.php
```

This fie contains useful functions related to the logs files, allowing to add and check contents to the logs file, useful for debug purposes

2.24.1   Check Log

```
function checklog($hash, $file)
```

This function is a helper for the show_php_error, allow to detect repetitions of the same text in the log file to prevent to add repeated lines, the usage is very simple, only requires a hash and a file to check that the hash is not found in the contents of the file, you can think in this function as a grep replacement that is able to found the hash in the file

- @hash => the pattern that you want to search in the file
- @file => the file where search the pattern

2.24.2 Add Log

```
function addlog($msg, $file = "")
```

This function add messages to the specified log file

- @msg => message that you want to add to the log file
- @file => the log file that you want to use without directory

Notes:

If not file is specified, the debug/logfile (saltos.log) is used by default

The logs files are stored in the logsdir (/data/logs)

This function performs the log rotation is the maxfilesize is reached

2.24.3 Add Trace

```
function addtrace($array, $file)
```

This function performs the addlog to the file using as input the array, the main idea is to pass the same array that the used in the show_php_error, the difference is that addtrace, only add the backtrace and debug to the array and then, saves the log to the specified file

- @array => the array that can contains the same info that show_php_error
- @file => the file where do you want to store the log contents

2.24.4 Get Trace

```
function gettrace($array)
```

This function get an array as show_php_error, add the backtrace and debug information and convert all array into a string

- @array => the array that can contains the same info that show_php_error

2.24.5 Session Backtrace

```
function session_backtrace()
```

Returns a string with the pid, sessid and current datetime with decimals

Notes:

The fields of this array allow to do low level debug processes, this data is generally used by the semaphores and some forced addtrace calls, but causes problems in the error reporting because break the hash and checklog optimization

2.25 Math utils helper module

```
api/php/autoload/math.php
```

This fie contains useful functions related to math operations

2.25.1 Sign function

```
function sign($n)
```

This function returns 1 for positive, -1 for negative and 0 for zero.

- @n => the number that you want to be processed

2.26 Memory helper module

```
api/php/autoload/memory.php
```

This fie contains useful functions related to memory and time usage, allow to control the usage of time and/or memory of a process, intended to prevent crashes in processes that requires more time or memory resources that the defined by the system limits

2.26.1 Memory Get Free

```
function memory_get_free($bytes = false)
```

This function returns the free memory in bytes or the percentage of the memory_limit

- @bytes => if true, returns the free bytes, if false, returns the percentage

2.26.2 Get Time Usage

```
function time_get_usage($secs = false)
```

This function returns the time usage in seconds or in percentage of the max_execution_time

- @secs => if true, returns the used seconds, if false, returns the percentage

2.26.3 Get Free Time

```
function time_get_free($secs = false)
```

This function returns the free time in seconds or in percentage of the max_execution_time

- @secs => if true, returns the used seconds, if false, returns the percentage

2.26.4 Init Time Get

```
function init_timer()
```

This function call the helper to initialize the static ini to the current microtime

2.26.5 Get Time helper

```
function __time_get_helper($fn, $secs)
```

This function is a helper of the time_get_usage and time_get_free functions, is used to check the time usage and the free time that remain to finish the execution of the script

2.26.6 Set Max Memory Limit

```
function set_max_memory_limit()
```

This function is intended to do a ini_set with a more greather value to allow an exceptionally amount of memory usage

2.26.7 Set Max Execution Time

```
function set_max_execution_time()
```

This function is intended to do a ini_set with a more greather value to allow an exceptionally amount of execution time

2.27 Mime helper module

```
api/php/autoload/mime.php
```

This file contains the mimetype feature provided by saltos using different techniques suck as the extension file, using the mime_content_type or the finfo_file functions.

2.27.1 SaltOS Content Type

```
function saltos_content_type($file)
```

This function is intended to returns the mime content-type string using different techniques.

- @file => the file of which you want to know the content-type

2.27.2 SaltOS Content Type first helper

```
function saltos_content_type0($mime)
```

This function returns the first part of the content-type, for example, if you pass the string image/jpeg, this function will returns the string image.

- @mime => the mime that you want to process

2.27.3 SaltOS Content Type second helper

```
function saltos_content_type1($mime)
```

This function returns the second part of the content-type, for example, if you pass the string image/jpeg, this function will returns the string jpeg.

- @mime => the mime that you want to process

2.28 Output helper module

```
api/php/autoload/output.php
```

This fie contains useful functions related to the output of the SaltOS, allow to send contents to the clients using the specified format and configuration, useful to return contents, too implement a specific output for the json format that is the most format used by the new SaltOS

2.28.1 Output Handler

```
function output_handler($array)
```

This function is intended to send data to the output channel, and can have the follow arguments:

- @array => array with the follow pairs of key val
- @file => file that contains the contents that you want to send
- @data => contents that you want to send to the output channel
- @type => content type header used
- @cache => boolean to enable the cache usage, includes the etag algorithm
- @name => the filename used in the content disposition attachment header
- @extra => headers that you can add to the transfer

2.28.2 Output Handler JSON

```
function output_handler_json($array)
```

This function allow to quickly send json output, the unique argument that it requires is the data that you want to send

- @array => content to convert to json and send to the output channel

2.29 Password helper module

```
api/php/autoload/password.php
```

This fie contains useful functions related to password, currently only publish one feature to check the password strength, but is open to be used to add more password features if it is needed

2.29.1 Password Strength

```
function password_strength($pass)
```

This fucntion returns a number between 0 and 100 that try to categorize the quality of the pass checked, this is useful to known if the new password is a good option or maybe is needed to request another new password

- @pass => password that do you want to check

2.30 Permissions helper module

```
api/php/autoload/perms.php
```

This fie contains useful functions related to permissions, allow to apply permissions in php core or in sql queries, to do it, uses all permissions tables and predefined configurations, more info in each function

2.30.1 Check User

```
function check_user($app, $perm)
```

This function checks the permissions using the tables apps_perms, users_apps_perms and groups_apps_perms, to do it, this function uses the user_id and groups_id (note that groups_id contains all groups where the user is associated), and try to check that the permissions permutations exists in the apps_perms, if some permission is found in the users and groups tables and it is not found in the apps_perms, an integrity error is launched.

- @app => the app to check
- @perm => the perm to check

2.30.2 Check SQL

```
function check_sql($app, $perm)
```

This function returns the fragment of SQL intended to filter by app and perm for the current user

- @app => the app to check
- @perm => the perm to check

Notes:

This function returns the portion of sql used to check permissions associated to an user with a specific permission and to an specific register, as an optimization, it detects if the all owner is on and return a true expression to improve the performance

2.30.3 Perms helper function

```
function __perms($fn, $arg)
```

This function is used by the XXX2YYY functions as helper, it stores the dictionary of all conversions and resolves the data using it

- @fn => the caller function

- @arg => the argument passed to the function

2.30.4 Id to Perm

```
function id2perm($id)
```

This function resolves the code of the perm from the perm id

- @id => the id used to resolve the perm

2.30.5 Perm to Id

```
function perm2id($perm)
```

This function resolves the id of the perm from the perm code

- @perm => the perm code used to resolve the id

Notes:

This function can return an integer or an array of integers, depending if the app is using the owner parameter or not

2.30.6 Perm Exists

```
function perm_exists($perm)
```

This function detect if a perm exists

- @perm => the perm that you want to check if exists

Notes:

This function returns true if a perm exists, and in case of the usage of the owner parameter, the function will return true for a perm that contains the owner and for the perm without the owner, for exampe, this function returns true for perm list and form perm list|user

2.30.7 Check App Perm Id

```
function check_app_perm_id($app, $perm, $id = null)
```

This function returns true if the app, the perm and the id accomplishes the expected level of permissions, it is intended to be used before the execution of each action, to guarantee the security

- @app => the app to check
- @perm => the perm to check
- @id => the id to check, if needed, you can omit in the create case

2.30.8 Check App Perm Id as Json

```
function check_app_perm_id_json($app, $perm, $id = null)
```

This function trigger a json error if the app, the perm and the id not accomplishes the expected level of permissions, it is intended to be used before the execution of each action, to guarantee the security

- @app => the app to check

- @perm => the perm to check

- @id => the id to check, if needed, you can omit in the create case

2.31 QRCode helper module

```
api/php/autoload/qrcode.php
```

This fie contains useful functions related to QRCodes

2.31.1 QRCode function

```
function __qrcode($msg, $s, $m)
```

This function allow to generate a qrcode with the SaltOS logo embedded in the center of the image, you can pass the desired message that you want to convert in qrcode and it returns an image with the data

- @msg => Contents of the qrcode

- @s => size of each pixel used in the qrcode

- @m => margin of the qrcode (white area that that surround the qrcode)

Notes:

The normal behavior is returns a png image, but if something was wrong, the function can returns an empty string

2.32 Random helper module

```
api/php/autoload/random.php
```

This fie contains useful functions related to random number generator, currently only initialize the internal generator, but in the future we can add more features if it is needed

2.32.1 Init Random

```
function init_random()
```

This function initialize the random number generator

Notes:

This function previously sets the seed using the microtime, but reading the srand php documentation, I see that the seed is not needed because if it is not provided, a randomly seed is used by default

2.33 Score image helper module

`api/php/autoload/score.php`

This fie contains useful functions related to score images

2.33.1 Score Image function

`function __score_image($score, $width, $height, $size)`

This function generates an image with a gradient from red to yellos and then, to green, depending of the score passed to the function, the params allos to define the size of the image or the size of the font used to write the score percent

- @score => a number between 0 and 100
- @width => the width of the generated image
- @height => the height of the generated image
- @size => the size of the font of the generated image

2.34 Semaphore helper module

`api/php/autoload/semaphores.php`

This fie contains useful functions related to semaphores, allow to create and use semaphores for acquiring and release semaphores that guaranty the exclusivity of the code execution

2.34.1 Semaphore Acquire

`function semaphore_acquire($name = "", $timeout = INF)`

This function implement the acquire of a semaphore

- @name => the name of the semaphore
- @timeout => the timeout used in waiting operations

2.34.2 Semaphore Release

`function semaphore_release($name = "")`

This function implement the release of the semaphore

- @name => the name of the semaphore

2.34.3 Semaphore Shutdown

`function semaphore_shutdown()`

This function implement the shutdown of all semaphores, to do it, the function will iterate in each semaphore to release and set to null the semaphore pointer

2.34.4 Semaphore File

```
function semaphore_file($name = "")
```

This function returns the associated semaphore file used by the named semaphore, useful for debug purposes

- @name => the name of the semaphore

2.34.5 Semaphore helper

```
function __semaphore_helper($fn, $name, $timeout)
```

This function implements the real semaphore functionalities, includes the code to do an acquire, the release, the shutdown and to get the file, is programmed as a function instead of a class by historical motivation, in reality, the statics fds acts as a properties of a class and each if stripos acts as a methods of a class

- @fn => the function name that call the helper, to detect the feature
- @name => the name of the semaphore
- @timeout => the timeout used in waiting operations

2.34.6 Semaphore USleep helper

```
function __semaphore_usleep($usec)
```

This function implements an usleep (micro sleeper) using sockets, this allow to break the execution of the function if a signal is received by the process, in reality, the feature is powered by the socket_select that is allowed to wait for read and write operations with a very precise timeout.

The returned value will be the difference between the end less the start, in other words, the returned value is the ellapsed time sleeped by the function

- @usec => the micro seconds that you want to sleep

2.35 Server helper module

```
api/php/autoload/server.php
```

This fie contains useful functions related to the $_SERVER global variable, currently only publish a getter function, but in the future, can store more features if it is needed

2.35.1 Get Server

```
function get_server($key)
```

This function returns the server variable requested by index if it exists

- @key => the index key used to get the value of the server

2.35.2 Set Server

```
function set_server($key, $val)
```

This function is intended to replace some server variabe in runtime mode

- @key => the index key used to get the value of the server
- @val => the value that you want to set in the server array

2.36 SQL utils helper module

```
api/php/autoload/sql.php
```

This fie contains useful functions related to SQL queries, allow to help modules that requires the entire management of the database (create tables, drop tables, create indexes, and more), too allow to prepare sql queries suck inserts, updates or wheres fragments that are procected to external injections, for example, by escaping all special characters.

Too it provides functions to do subparts of the where queries suck as special likes combinations or match again combinations for the fulltext search engine, see all detailed information by reading the list of functions of this module

2.36.1 Parse Query

```
function parse_query($query, $type = "")
```

This function is intended to apply the query filters defined by the users when write queries for multiples db engines as MySQL and/or SQLite, for example, if you want to write a fragment of SQL with one version for MySQL and another version for SQLite, you can do / *MYSQL ... * // *SQLite ... * /

Note that the previous example add a spaces between the bar and the asterisc because we can not put comments inside another comment!!!

- @query => the query that must be parsed
- @type => the db type that you want to allow by the filters

2.36.2 Parse Query Type helper

```
function __parse_query_type()
```

This function returns the type used by parse_query using as detector the dbtype of the config file, currently only allow to return MYSQL and/or SQLITE

2.36.3 Parse Query Strpos helper

```
function __parse_query_strpos($haystack, $needle, $offset = 0)
```

This function is the same that strpos, but with some improvements required by the parse_query funcion, the idea is to use the strpos functionality, but controlling that the found position must acomplish some constraints as the number of simple and double quotes must to be even

The arguments are the same that the strpos function

- @haystack => string where search the needle
- @needle => the needle text that must be found in the haystack
- @offset => bias applied to begin the search of the needle

2.36.4 Execute Query

```
function execute_query($query)
```

This function executes the query and depending in the result, returns the resultset trying to do the more good combination in the return data

- @query => the SQL query that you want to execute

Note that the db_query is executed with the "auto" fetch mode, this causes that the db_query returns an array with one dimension if the query only generates a resultset with only one column, or returns an array with two dimensions if the query generates a resultest with more that one column

To be more practice:

If you execute a query that select one field and only returns one row, the return value will be the value of the field

If you execute a query that select one field and returns more that one row, the return value will be an array of one dimension with all values of this field

If you execute a query that select multiples fields and only return one row, the return value will be an array of one dimension with all fields

Ig you execute a query that select multiples fields and returns more that one row, the return value will be an array of two dimensions with all rows and each row with all fields

Be carefully to use the output of this command in an foreach, for example because you can get for the same query differents output types, if you need to be more standarized in the output types, see the execute_query_array

2.36.5 Execute Query Array

```
function execute_query_array($query)
```

This function is the same that execute_query but guarantee that for the same query, you get the same output type if the resultet contains one row or more rows, useful is you want to use the output of this function in a foreach, for example

- @query => the SQL query that you want to execute

2.36.6 Get Fields

```
function get_fields($table)
```

This function returns the fields of the requested table

- @table => the table where that you want to know the fields

2.36.7 Get Indexes

```
function get_indexes($table)
```

This function returns the indexes of the requested table

- @table => the table where that you want to know the indexes

2.36.8 Get Tables

```
function get_tables()
```

This function returns the tables of the database

2.36.9 Get Field Type

```
function get_field_type($type)
```

This function returns an standarized type for the specific types used in the real database, for example, returns string if the field is of TEXT type

- @type => the real type in the database

2.36.10 Get Field Size

```
function get_field_size($type)
```

This function returns the size for the types used in the database, for example, returns 65535 if the field is of TEXT type

- @type => the real type in the database

2.36.11 SQL Create Table

```
function sql_create_table($tablespec)
```

This function returns the SQL needed to create the table defined in the tablespec argument

- @tablespec => the specification for the create table, see the dbschema file to understand the tablespec structure

This function creates the table, supports the primary key, supports the foreign key, and detect fulltext indexes with mroonga engines

2.36.12 Has Engine

```
function __has_engine($engine)
```

This function allow to SaltOS to ask to the database if an enxine is availabie

- @engine => the engine that you want to get information about existence

2.36.13 SQL Alter Table

```
function sql_alter_table($orig, $dest)
```

This function returns the alter table command

- @orig => source table
- @dest => destination table

2.36.14 SQL Insert From Select

```
function sql_insert_from_select($dest, $orig)
```

This function returns the insert from select command

- @orig => source table
- @dest => destination table

2.36.15 SQL Drop Table

```
function sql_drop_table($table)
```

This function returns the drop table command

- @table => table that you want to drop

2.36.16 SQL Create Index

```
function sql_create_index($indexspec)
```

This function returns the SQL needed to create the index defined in the indexspec argument

- @indexspec => the specification for the create index, see the dbschema file to understand the indexspec structure

This function creates the index, supports fulltext indexes

2.36.17 SQL Drop Index

```
function sql_drop_index($index, $table)
```

This function returns the drop index command

- @index => index that you want to drop
- @table => table where the indes is part of

2.36.18 Make Insert Query

```
function make_insert_query($table, $array)
```

Returns the insert query for the table with all fields specified by the array param

- @table => table where you want to add the register
- @array => array with key val pairs that represent the field and the value of the field

Notes:

This function tries to cast each value to their data type getting this information from dbschema config, you can pass in array all fields that you want and not is needed to put all fields of the table, only the fields that appear in the array will be used in the insert, if some field is not a part of the fields of the table, an error will be triggered

This function uses the array_key_exists instead of isset because the check of the $array[$name] fails when the item exists but is false or null, for example

2.36.19 Make Update Query

```
function make_update_query($table, $array, $where)
```

Returns the update query for the table with all fields specified by the array param and using the specified where

- @table => table where you want to update the register
- @array => array with key val pairs that represent the field and the value of the field
- @where => where clausule used to update only the expected registers, can be the output of make_where_query

Notes:

This function tries to cast each value to their data type getting this information from dbschema config, you can pass in array all fields that you want and not is needed to put all fields of the table, only the fields that appear in the array will be used in the update, if some field is not a part of the fields of the table, an error will be triggered

This function uses the array_key_exists instead of isset because the check of the $array[$name] fails when the item exists but is false or null, for example

2.36.20 Make Where Query

```
function make_where_query($array)
```

This function allow to create where sentences joinin all fields by AND

- @array => array with key val pairs that represent the field and the value of the field

Notes:

The keys normally contains the name of the field, but if you need to use a different comparison operator, you can use the field name and add the operator that you want to use in the comparison, the allowed comparison operators are >, <, =, >=, <=, !=

2.36.21 Escape Reserved Word

```
function escape_reserved_word($word)
```

This function tries to escape the reserved words that can not be used in sql queries as field names or table names, currently is only used to escape field names but in a future, if it is needed, can be added to escape table names too

- @word => the word that must to be escape if needed

2.36.22 Make Like Query

```
function make_like_query($keys, $values, $args = [])
```

This function is intended to returns the sql fragment to be added to the where condition to filter for the specified keys and values

- @keys => an string with comma separated field names
- @values => the value of the input search
- @minsize => the minimal size of the length used in each like
- @default => sql fraement returned if some thing was wrong

Notes:

This function generates a sequence of (like or like) and (like and like) and is able to understand the prefix plus or minus in each word of the search string, this allow to the function to use the like or not like depending the sign of the word, and too to use the disjunction or conjunction in each like group

2.36.23 Make Fulltext Query Helper

```
function __make_fulltext_query_helper($values, $args = [])
```

This function is similar to the make_like_query, but uses the match agains clausule instead of the like clausule, the match agaings is used for fulltext searches and generally, this function is not intended to be used directly, it must acts as a helper of the make_fulltext_query

- @values => the value of the input search
- @minsize => the minimal size of the length used in each like
- @default => sql fraement returned if some thing was wrong

Notes:

This function differs between the make_like_query in the idea that this function only is used to search using fulltext indexes and in one unique field named search

2.36.24 Make Fulltext Query

```
function make_fulltext_query($values, $app, $args = [])
```

While the two version returns the fragment that must to be added to the query that search in the table that contains the search field, this function allow to specify the same that the two version with two fields more, the app and the prefix to be added to the id field of the in subquery

- @values => the value of the input search
- @app => the app used to detect the indexing table

- @prefix => the prefix added to the id used in the in subquery

- @minsize => the minimal size of the length used in each like

- @default => sql fraement returned if some thing was wrong

2.37 String utils helper module

`api/php/autoload/strings.php`

This fie contains useful functions related to strings manipulations, suck as get the test in UTF8, remove bad chars, of apply filters to the text, manipulate texts of other related tasks with strings

2.37.1 Remove Bar Chars

`function remove_bad_chars($temp, $pad = "")`

This function removes chars from keycodes 0 to 31 except 9, 10, 13 (tab, newline, return)

- @temp => input string that you want to fix

- @pad => padding string used as replacement for bar chars (void by default)

2.37.2 Encode Bar Chars

`function encode_bad_chars($cad, $pad = "_", $extra = "")`

This function tries to replace accender chars and other extended chars into an ascii chars, to do it, they define an array with the pairs of chars to do a quick replace, too is converted all to lower and are removed all chars that are out of range (valid range are from 0-9 and from a-z), the function allow to specify an extra parameter to add extra chars that must to be allowed in the output, all other chars will be converted to the padding argument, as a bonus extra, all padding repetitions will be removed to only allow one pading char at time

- @cad => the input string to encode

- @pad => the padding char using to replace the bar chars

- @extra => the list of chars allowed to appear in the output

2.37.3 Prepare Words

`function prepare_words($cad, $pad = " ")`

This function allow to prepare words removing repetitions in the padding char

- @cad => the input string to prepare

- @pad => the padding char using to replace the repetitions

Notes:

Apart of remove repetitions of the padding char, the function will try to remove padding chars in the start and in the end of the string

2.37.4 Sprintr

```
function sprintr($array)
```

This function is an improved version of the print_r, allow to convert an array into a string removing some extra lines that not contain information, lines that contains only contains an open or close parenthesis, or nothing, are removed, optimizing the output string

- @array => the array that do you want to convert into string

2.37.5 Get Unique ID MD5

```
function get_unique_id_md5()
```

This function returns an unique hash using the random generator

2.37.6 Intelligence Cut

```
function intelligence_cut($txt, $max, $end = "...")
```

This function allow to cut text by searching spaces to prevent to break words

- @txt => the text that you want to cut
- @max => the size of the expected output text
- @end => the suffix added if the text is cutted

2.37.7 Normalize Value

```
function normalize_value($value)
```

This function allow to detect the last letter to detect what magnitude is using (K, M or G) and multiply the numeric part by the needed factor to get the number without factor

- @value => the string that contain the number, for example "123k"

2.37.8 HTML to Text

```
function html2text($html)
```

This function uses the html2text roundcube function to convert html to plain text, this code have the issue that requires the error_reporting(0) because it have a lot of errors causes by use nondefined variables, for example

- @html => the html code that you want to convert to plain text

2.37.9 Get UTF-8

```
function getutf8($str)
```

This function returns the string codified in a UTF-8 encoding

- @str => the input string that you want to covnert to UTF-8

2.37.10 Words Exists

```
function words_exists($words, $buffer)
```

This function check that all words exists in the buffer

- @words => the string that contains words separated by spaces
- @buffer => the string where we must to found the words

2.37.11 String Replace Assoc

```
function str_replace_assoc($array, $cad)
```

This function do the same that str_replace, but using only one associative array, using the keys as search and the values as replace, intended only to do more prerry the code

- @array => the associative array with the pairs keys vals
- @cad => the string that you want to apply the replacement

2.37.12 Get Part From String

```
function get_part_from_string($input, $delim, $index)
```

This function explodes de input using delim and returns the element of the index position

- @input => the string that you want to cut in parts
- @delim => the delimiter char used to cut in parts
- @index => the index that you want to request of the explode result

2.37.13 Get Unique Token

```
function get_unique_token()
```

This function returns a string with a new and unique token

2.37.14 Check IDS

```
function check_ids()
```

This function checks the correctness of the list of ids and returns a valid list available to be used in sql queries

- @ids => the string containing the list of ids

2.38 System helper module

```
api/php/autoload/system.php
```

This fie contains useful functions related to system checks, allow to detect dependencies not installed on the system, or misconfigurations on the SaltOS installation

2.38.1 Check System

```
function check_system()
```

This function checks the system to detect if all knowed dependencies are found in the system, to do it, defines an array with the type (class or function), the name and some extra info for the error message that is triggered if the dependency is not satisfied

Too, check all directories of the data directory to validate that the process can write inside it

2.39 TODO

```
api/php/autoload/unoconv.php
```

TODO

2.39.1 TODO

```
function unoconv2pdf($input)
```

TODO

2.39.2 TODO

```
function unoconv2txt($input)
```

TODO

2.39.3 TODO

```
function __unoconv_list()
```

TODO

2.39.4 TODO

```
function __unoconv_pdf2txt($input, $output)
```

TODO

2.39.5 TODO

```
function __unoconv_all2pdf($input, $output)
```

TODO

2.39.6 TODO

```
function __unoconv_convert($input, $output, $format)
```

TODO

2.39.7 TODO

```
function __unoconv_img2ocr($file)
```

TODO

2.39.8 TODO

```
function __unoconv_pdf2ocr($pdf)
```

TODO

2.39.9 TODO

```
function __unoconv_histogram($values, $usage1, $usage2)
```

TODO

2.39.10 TODO

```
function __unoconv_rotate($posx, $posy, $angle)
```

TODO

2.39.11 TODO

```
function __unoconv_node2attr($node)
```

TODO

2.39.12 TODO

```
function __unoconv_node2value($node)
```

TODO

2.39.13 TODO

```
function __unoconv_lines2matrix($lines, $width, $height)
```

TODO

2.39.14 TODO

```
function __unoconv_fixline($line, $pos1, $pos2, $pos3, $pos4)
```

TODO

2.39.15 TODO

```
function __unoconv_hocr2txt($hocr)
```

TODO

2.39.16 TODO

```
function __unoconv_substr($string, $start, $length, $reference)
```

TODO

2.39.17 TODO

```
function __unoconv_substr2d($page, $x1, $x2, $x3, $y1, $y2, $y3)
```

TODO

2.39.18 TODO

```
function __unoconv_remove_margins($page)
```

TODO

2.40 User helper module

```
api/php/autoload/user.php
```

This file contains the functions associated to the user validation, to improve the performance, all functions are using a cache based trick that performs an important speed up

2.40.1 Current Token

```
function current_token()
```

This function returns the id of the current token, this info is retrieved using the token of the request, for security reasons, this validation only can be performed by the same origin that execute the login action

2.40.2 Current User

```
function current_user()
```

This function returns the id of the current user, this info is retrieved using the token of the request

2.40.3 Current Group

```
function current_group()
```

This function returns the id of the current group, this info is retrieved using the token of the request

2.40.4 Current Groups

```
function current_groups()
```

This function returns the id of all current groups, this info is retrieved using the token of the request and the main idea of this function is to returns the list of all groups associated to the current user to facily the permissions checks

2.40.5 Crontab Users

```
function crontab_users()
```

This function executes the maintenance queries to update the active field in the passwords and tokens tables, it's intended to be used as helper

Notes:

This function uses an internal static variable to detect repeated executions and only accepts the first execution, this is to prevent that multiples calls to other actions and functions that requires the integrity of the passwords and tokens

2.41 Version helper module

```
api/php/autoload/version.php
```

This fie contains useful functions related to the version of the SaltOS of the php engine

2.41.1 Get Name Version Revision

```
function get_name_version_revision($copyright = false)
```

This function returns a string with the SaltOS name, version, revision and copyright if needed

- @copyright => boolean to specify if you want to add the copyright to the output

2.41.2 SVN Version

```
function svnversion($dir = ".")
```

This function tries to return the svn version of the project

- @dir => allow to specify where do you want to execute the svnversion command

2.41.3 GIT Version

```
function gitversion($dir = ".")
```

This function tries to return the git version of the project

- @dir => allow to specify where do you want to execute the gitversion command

2.41.4 IS PHP

```
function isphp($version)
```

This function returns a boolean as a response about the comparison between the version requested and the current version.

- @version => the version where do you want to compare

2.42 XML to Array helper module

```
api/php/autoload/xml2array.php
```

This fie contains useful functions related to the xml manipulation, this code is a part of the main version of the SaltOS because the main idea defined some years ago continues active, if you compare this code, you can see more accuracy in the specification to be more specific and precise but this code is a part of all the SaltOS versions

2.42.1 Eval Protected

```
function eval_protected($input, $global = "")
```

This function allow to execute PHP code using the eval function in a controlled environment, you can specify some global variables to improve the eval execution

- @input => the code to be executed
- @global => the list (separated by comma) of global variables that you want to use

2.42.2 Set Array

```
function set_array(&$array, $name, $value)
```

This function allow to specify multiples entries in an array with the same key, to do this, the function will add #num where num is a unique number, in reality if you want to set multiples entries for the key "test", you get in reality an array with entries as "test", "test#1", "test#2"

This function works in concordance of the fix_key, that is able to get the key as "test#1" and return only "test" that is the original key without the suffix added to allow multiples instances of the same key in an associative array

- @array => array that you want to add the key with the value (by reference)
- @name => the key used in the array, if exists, it will try to add the suffix to prevent collisions
- @value => the value that you want to set in this position of the array

2.42.3   Unset Array

```
function unset_array(&$array, $name)
```

This function remove all entries of the array that matches with the name of the key, for example, if you specify the name "test", the function unset all entries as "test" or begin by "test#", in the example of the previous function will remove "test", "test#1" and "test#2"

- @array => array that you want to remove the key (by reference)
- @name => the key used in the array and as prefix of the entries of the array

2.42.4   Fix Key

```
function fix_key($arg)
```

This function returns the "real" part of the key removing the suffix added to prevent collisions in the associative array, for the above example, if you request the fix_key of the "test#2", the function will returns "test"

- @arg => the name of the key that you want to remove the suffix part (if exists)

2.42.5   XML Files to Array

```
function xmlfiles2array($files, $usecache = true)
```

This function allow to convert all XML files to an array, allow to use cache to optimize repetitive calls of the same file

As an special mention, this function internally implements the old xml_join feature that allow to merge multiple files into one using the fix_key of the keys in the first level as key to join.

- @files => the files that you want to convert from xml to array
- @usecache => if do you want to enable the cache feature

2.42.6   XML File to Array

```
function xmlfile2array($file, $usecache = true)
```

This function allow to convert a XML file to an array, allow to use cache to optimize repetitive calls of the same file

As an special mention, this function internally uses semaphores to prevent multiple instances of the same execution with the same file, too uses a cache management to optimize the usage

- @file => the file that you want to convert from xml to array

- @usecache => if do you want to enable the cache feature

2.42.7   XML to Array

```
function xml2array($xml, $file = "")
```

This function allow to convert a XML string to an array

- @xml => xml code to be converted to an array

- @file => filename of the contents, only used when an errors occurs

2.42.8   XML to Struct

```
function xml2struct($xml, $file = "")
```

This function is a helper of the xml2array function, the main purpose of this function is to convert the xml string into a struct to be processed by the struct2array function

The motivation to use the xml_parse_into_struct function is that this function is the more quick to parse xml files, after a lot of tests, the more quickly execution is to use the xml_parse_into_struct, reverse the array and then program a simple recursive function that convert a unidimensional array into a tree

At the begining of this function, we will try to detect the enconding of the xml file, the main objective is to convert all xml to UTF-8 that is the default enconding of SaltOS

The returned value is the result of the xml_parse_into_struct function, that is the key of this feature and this function

- @xml => xml fragment that must be converted into struct

- @file => the source filename, it is used only to generate error reports

2.42.9   Struct to Array

```
function struct2array(&$data, $file = "")
```

This function is the second part in the xml2array convertsion, here, the function receives an unidimensional array with commands to open, close, and their respective values and attributes, with this information, this function is able to generate a tree with the xml converted to an array tree

- @data => the struct array, by reference

- @file => the source filename, it is used only to generate error reports

Notes:

This function uses recursivity to accomplish the objetive, returns the portion of xml between the open and close command, in each call, the data array passed by reference will decrement in size because the array_pop removes the last element of the array

Remember that previously of call this function, the array is reversed, this is because is more efficient to do a reverse and then pops instead of use directly the array_shift to get the next element, the reason is that array_shift must to reorder all keys of the resulted array and this add a very big cost if the xml is big, this problem was detected in 2014 and was optimized by add the reverse and the pop instead of only shift

2.42.10 Eval Attributes

```
function eval_attr($array)
```

This function is very special in SaltOS, is part of the initial code an is used by a lot of parts of the program, currently are using a simplified version of the original function and have improvements that allow to return arrays with attributes without evaluate and without causing an error, this allow to define xml with attributes that can be used by other processes and SaltOS only interpret three attributes

- @array => the array that contains a tree representation of the xml

The three attributes are:

- @global => this attribute allow to SaltOS to prepare what variales must to be global in the eval_protected call

- @eval => this attribute must be a boolean and allow to evaluate the value of the node

- @ifeval => this attribute must contains an expression that must evaluate as true or false, and allow to maintain or remove the entire node thas contains the ifeval attribute, this is useful when you need a node in some conditions

The great change between the eval_attr of the previous versions of SaltOS is that this version only accepts three internal commands and the other attributes can be maintained in order to be used by other processes (internally or externally)

2.42.11 Eval Bool

```
function eval_bool($arg)
```

This function returns a boolean depending on the input evaluation, the main idea is to get an string, for example, and determine if must be considered true or false otherwise will finish in an error

The valid inputs are the strings one, zero, void, true, false, on, off, yes and no

- @arg => the value that do you want to evaluates as boolean

2.43 Main execution module

```
api/php/autoload/zindex.php
```

This file contains the old index.php file, this was moved here to simplify the index.php and to allow some php checks found in the current index.php

This code implements the main method to access to the SaltOS API using rest and json requests, to use it, you can use the follow methods:

1) Rest using GET requests

This kind of requests requires that you send a GET request with a querystring of the follow form:

- @https://127.0.0.1/saltos/code4/?app/invoices/view/2

And the system process it of the follow form:

- @rest/1 => invoices
- @rest/2 => view
- @rest/3 => 2

And you can programm any action that uses these parameters to do the desired task

2) Json using POST requests

This other kind of requests requires that you send a POST request with the appropiate header for the content-type as application/json and a json in the body of the request, with this call, saltos can map all contents of the json to the json/????? variables.

As an extra bonus, this module defines some usefull server variables used in a lot of features of saltos, like the follow vars:

- @request_method => can be GET or POST
- @content_type => used to check the content type for the JSON requests
- @token => used to validate the HTTP_TOKEN send as authentication
- @remote_addr => used internally for security reasons
- @user_agent => used internally for security reasons

As a brief resume, you can use the follow keys in get_data or set_data:

- @rest => to get an array with all rest data, for the above example they must return some thing like this: ["app", "invoices", "view", "2"]
- @rest/1 => to get only the element that contains "invoices"
- @rest/2 => to get only the element that contains "view"
- @rest/3 => to get only the element that contains "2"
- @json => to get an array with all json data, for the above example they must return some thing like this: ["action"=>"authtoken", "user"=>"xxx", "pass"=>"xxx"]
- @json/action => to get only the element that contains "authtoken"
- @json/user => to get only the element that contains the user
- @json/pass => to get only the element that contains the pass
- @server => to get an array with all server data
- @server/request_method => can be GET or POST
- @server/content_type => used to check the content type for the JSON requests
- @server/token => used to validate the HTTP_TOKEN send as authentication
- @server/remote_addr => used internally for security reasons
- @server/user_agent => used internally for security reasons

Chapter 3

Database

3.1 SQLite3 functions library

```
api/php/database/libsqlite.php
```

SQLite's database allow to define external functions that can be used from the SQL language, this is a great feature that allow to use SQLite as MySQL, and using this feature of the database, the SQLite drivers use the libsqlite to add a lot of features found in MySQL and used in a lot of queries by SaltOS

More info about this feature by searching:

- @PDO => sqliteCreateFunction/sqliteCreateAggregate
- @SQLite3 => createFunction/createAggregate

3.1.1 GROUP_CONCAT

```
function __libsqlite_group_concat_step($context, $rows, $string, $separator
    = ",")
```

This function add the GROUP_CONCAT to the SQLite database

3.1.2 GROUP_CONCAT

```
function __libsqlite_group_concat_finalize($context, $rows)
```

This function add the GROUP_CONCAT to the SQLite database

3.1.3 REPLACE

```
function __libsqlite_replace($subject, $search, $replace)
```

This function add the REPLACE to the SQLite database

3.1.4   LPAD

```
function __libsqlite_lpad($input, $length, $char)
```

This function add the LPAD to the SQLite database

3.1.5   CONCAT

```
function __libsqlite_concat()
```

This function add the CONCAT to the SQLite database

3.1.6   CONCAT_WS

```
function __libsqlite_concat_ws()
```

This function add the CONCAT_WS to the SQLite database

3.1.7   UNIX_TIMESTAMP

```
function __libsqlite_unix_timestamp($date)
```

This function add the UNIX_TIMESTAMP to the SQLite database

3.1.8   FROM_UNIXTIME

```
function __libsqlite_from_unixtime($timestamp)
```

This function add the FROM_UNIXTIME to the SQLite database

3.1.9   YEAR

```
function __libsqlite_year($date)
```

This function add the YEAR to the SQLite database

3.1.10   MONTH

```
function __libsqlite_month($date)
```

This function add the MONTH to the SQLite database

3.1.11   WEEK

```
function __libsqlite_week($date, $mode = 0)
```

This function add the WEEK to the SQLite database

3.1.12 TRUNCATE

```
function __libsqlite_truncate($n, $d)
```

This function add the TRUNCATE to the SQLite database

3.1.13 DAY

```
function __libsqlite_day($date)
```

This function add the DAY to the SQLite database

3.1.14 DAYOFYEAR

```
function __libsqlite_dayofyear($date)
```

This function add the DAYOFYEAR to the SQLite database

3.1.15 DAYOFWEEK

```
function __libsqlite_dayofweek($date)
```

This function add the DAYOFWEEK to the SQLite database

3.1.16 HOUR

```
function __libsqlite_hour($date)
```

This function add the HOUR to the SQLite database

3.1.17 MINUTE

```
function __libsqlite_minute($date)
```

This function add the MINUTE to the SQLite database

3.1.18 SECOND

```
function __libsqlite_second($date)
```

This function add the SECOND to the SQLite database

3.1.19 MD5

```
function __libsqlite_md5($temp)
```

This function add the MD5 to the SQLite database

3.1.20   REPEAT

```
function __libsqlite_repeat($str, $count)
```

This function add the REPEAT to the SQLite database

3.1.21   FIND_IN_SET

```
function __libsqlite_find_in_set($str, $strlist)
```

This function add the FIND_IN_SET to the SQLite database

3.1.22   IF

```
function __libsqlite_if($condition, $value_if_true, $value_if_false)
```

This function add the IF to the SQLite database

3.1.23   POW

```
function __libsqlite_pow($base, $exp)
```

This function add the POW to the SQLite database

3.1.24   DATE_FORMAT

```
function __libsqlite_date_format($date, $format)
```

This function add the DATE_FORMAT to the SQLite database

3.1.25   NOW

```
function __libsqlite_now()
```

This function add the NOW to the SQLite database

3.2   MySQL improved driver

```
api/php/database/mysqli.php
```

This file implements the MySQL improved driver. This driver was coded later that the initial mysql driver and contains lots of improvements respect to the old mysql driver. The old mysql driver was removed in the PHP 7.0 release.

3.2.1   Database MySQL improved class

```
class database_mysqli
```

This class allow to SaltOS to connect to MySQL databases using the MySQL improved driver

3.2.2   Private link variable

```
private $link = null;
```

This private variable contains the link to the database

3.2.3   Constructor

```
public function __construct($args)
```

This public function is intended to stablish the connection to the database

- @args => is an array with key val pairs
- @host => the host for the connection
- @port => the port used for the connection
- @name => name of the database for the connection
- @user => user used to stablish the connection
- @pass => pass used to stablish the connection

Notes of this driver:

MYSQLI_REPORT_ERROR | MYSQLI_REPORT_STRICT are the same that PDO::ERRMODE_EXCEPTION
MYSQLI_USE_RESULT is the same that PDO::MYSQL_ATTR_USE_BUFFERED_QUERY = false

3.2.4   DB Check

```
public function db_check($query)
```

This public function is intended to check that the query execution will not trigger an error

- @query => the query that you want to validate

3.2.5   DB Query

```
public function db_query($query, $fetch = "query")
```

This public function is intended to execute the query and returns the resultset

- @query => the query that you want to execute
- @fetch => the type of fetch that you want to use, can be auto, query, column or concat

Notes:

The fetch argument can perform an speed up in the execution of the retrieve action, and can modify how the result is returned

auto: this fetch method try to detect if the resultset contains one or more columns, and sets the fetch to column (if the resultset only contains one column) or to query (otherwise)

query: this fetch method returns all resultset as an array of rows, and each row contain the pair of key val with the name of the field and the value of the field

column: this fetch method returns an array where each element is each value of the field of the each row, this is useful when for example do you want to get all ids of a query, with this method you can obtain an array with each value of the array is an id of the resultset

concat: this fetch method is an special mode intended to speed up the retrieve of large arrays, this is useful when you want to get all ids of a query and you want to get a big sized array, in this case, is more efficient to get an string separated by commas with all ids instead of an array where each element is an id

3.2.6   DB Disconnect

```
public function db_disconnect()
```

This function close the database connection and sets the link to null

3.3   PDO MsSQL driver

```
api/php/database/pdo_mssql.php
```

This file implements the MySQL improved driver. This driver uses the dblib library to connect with SQL servers.

3.3.1   Database PDO MsSQL class

```
class database_pdo_mssql
```

This class allow to SaltOS to connect to MsSQL databases using the PDO driver

3.3.2   Private link variable

```
private $link = null;
```

This private variable contains the link to the database

3.3.3   Constructor

```
public function __construct($args)
```

This public function is intended to stablish the connection to the database

- @args => is an array with key val pairs

- @host => the host for the connection
- @port => the port used for the connection
- @name => name of the database for the connection
- @user => user used to stablish the connection
- @pass => pass used to stablish the connection

3.3.4 DB Check

```
public function db_check($query)
```

This public function is intended to check that the query execution will not trigger an error

- @query => the query that you want to validate

3.3.5 DB Query

```
public function db_query($query, $fetch = "query")
```

This public function is intended to execute the query and returns the resultset

- @query => the query that you want to execute
- @fetch => the type of fetch that you want to use, can be auto, query, column or concat

Notes:

The fetch argument can perform an speed up in the execution of the retrieve action, and can modify how the result is returned

auto: this fetch method try to detect if the resultset contains one or more columns, and sets the fetch to column (if the resultset only contains one column) or to query (otherwise)

query: this fetch method returns all resultset as an array of rows, and each row contain the pair of key val with the name of the field and the value of the field

column: this fetch method returns an array where each element is each value of the field of the each row, this is useful when for example do you want to get all ids of a query, with this method you can obtain an array with each value of the array is an id of the resultset

concat: this fetch method is an special mode intended to speed up the retrieve of large arrays, this is useful when you want to get all ids of a query and you want to get a big sized array, in this case, is more efficient to get an string separated by commas with all ids instead of an array where each element is an id

3.3.6 DB Disconnect

```
public function db_disconnect()
```

This function close the database connection and sets the link to null

3.4 PDO MySQL driver

```
api/php/database/pdo_mysql.php
```

This file implements the MySQL improved driver. This is the recommended driver when you want to use MySQL servers, it uses the PDO extension and alow to connect to MySQL servers with a good performance.

3.4.1   Database PDO MySQL class

```
class database_pdo_mysql
```

This class allow to SaltOS to connect to MySQL databases using the PDO driver

3.4.2   Private link variable

```
private $link = null;
```

This private variable contains the link to the database

3.4.3   Constructor

```
public function __construct($args)
```

This public function is intended to stablish the connection to the database

- @args => is an array with key val pairs
- @host => the host for the connection
- @port => the port used for the connection
- @name => name of the database for the connection
- @user => user used to stablish the connection
- @pass => pass used to stablish the connection

3.4.4   DB Check

```
public function db_check($query)
```

This public function is intended to check that the query execution will not trigger an error

- @query => the query that you want to validate

3.4.5   DB Query

```
public function db_query($query, $fetch = "query")
```

This public function is intended to execute the query and returns the resultset

- @query => the query that you want to execute
- @fetch => the type of fetch that you want to use, can be auto, query, column or concat

Notes:

The fetch argument can perform an speed up in the execution of the retrieve action, and can modify how the result is returned

auto: this fetch method try to detect if the resultset contains one or more columns, and sets the fetch to column (if the resultset only contains one column) or to query (otherwise)

query: this fetch method returns all resultset as an array of rows, and each row contain the pair of key val with the name of the field and the value of the field

column: this fetch method returns an array where each element is each value of the field of the each row, this is useful when for example do you want to get all ids of a query, with this method you can obtain an array with each value of the array is an id of the resultset

concat: this fetch method is an special mode intended to speed up the retrieve of large arrays, this is useful when you want to get all ids of a query and you want to get a big sized array, in this case, is more efficient to get an string separated by commas with all ids instead of an array where each element is an id

3.4.6   DB Disconnect

```
public function db_disconnect()
```

This function close the database connection and sets the link to null

3.5   PDO SQLite driver

```
api/php/database/pdo_sqlite.php
```

This file implements the MySQL improved driver. This is the recommended driver when you want to use SQLite3 files as database server and it uses the PDO extension to do it, this driver solves the concurrence problem using POSIX semaphores, generally it is a good option for setups that don't require a fulltext search optimizations suck as mroonga, intended for a personal usage or demos.

3.5.1   Database PDO SQLite class

```
class database_pdo_sqlite
```

This class allow to SaltOS to connect to SQLite databases using the PDO driver

3.5.2   Private link variable

```
private $link = null;
```

This private variable contains the link to the database

3.5.3   Constructor

```
public function __construct($args)
```

This public function is intended to stablish the connection to the database

- @args => is an array with key val pairs
- @file => the file that contains the database

Notes:

This database allow to define external functions that can be used from the SQL language, this is a great feature that allow to use SQLite as MySQL, and using this feature of the database, this driver uses the libsqlite to add a lot of features found in MySQL and used in a lot of queries by SaltOS

3.5.4 DB Check

```
public function db_check($query)
```

This public function is intended to check that the query execution will not trigger an error

- @query => the query that you want to validate

3.5.5 DB Query

```
public function db_query($query, $fetch = "query")
```

This public function is intended to execute the query and returns the resultset

- @query => the query that you want to execute
- @fetch => the type of fetch that you want to use, can be auto, query, column or concat

Notes:

The fetch argument can perform an speed up in the execution of the retrieve action, and can modify how the result is returned

auto: this fetch method try to detect if the resultset contains one or more columns, and sets the fetch to column (if the resultset only contains one column) or to query (otherwise)

query: this fetch method returns all resultset as an array of rows, and each row contain the pair of key val with the name of the field and the value of the field

column: this fetch method returns an array where each element is each value of the field of the each row, this is useful when for example do you want to get all ids of a query, with this method you can obtain an array with each value of the array is an id of the resultset

concat: this fetch method is an special mode intended to speed up the retrieve of large arrays, this is useful when you want to get all ids of a query and you want to get a big sized array, in this case, is more efficient to get an string separated by commas with all ids instead of an array where each element is an id

3.5.6 DB Disconnect

```
public function db_disconnect()
```

This function close the database connection and sets the link to null

3.6 SQLite3 driver

```
api/php/database/sqlite3.php
```

This file implements the MySQL improved driver. This is the recommended driver when you want to use SQLite3 files as database server, this driver solves the concurrence problem using POSIX semaphores, generally it is a good option for setups that don't require a fulltext search optimizations suck as mroonga, intended for a personal usage or demos.

3.6.1 Database SQLite3 class

```
class database_sqlite3
```

This class allow to SaltOS to connect to SQLite databases using the SQLite3 driver

3.6.2 Private link variable

```
private $link = null;
```

This private variable contains the link to the database

3.6.3 Constructor

```
public function __construct($args)
```

This public function is intended to stablish the connection to the database

- @args => is an array with key val pairs
- @file => the file that contains the database

Notes:

This database allow to define external functions that can be used from the SQL language, this is a great feature that allow to use SQLite as MySQL, and using this feature of the database, this driver uses the libsqlite to add a lot of features found in MySQL and used in a lot of queries by SaltOS

3.6.4 DB Check

```
public function db_check($query)
```

This public function is intended to check that the query execution will not trigger an error

- @query => the query that you want to validate

3.6.5 DB Query

```
public function db_query($query, $fetch = "query")
```

This public function is intended to execute the query and returns the resultset

- @query => the query that you want to execute
- @fetch => the type of fetch that you want to use, can be auto, query, column or concat

Notes:

The fetch argument can perform an speed up in the execution of the retrieve action, and can modify how the result is returned

auto: this fetch method try to detect if the resultset contains one or more columns, and sets the fetch to column (if the resultset only contains one column) or to query (otherwise)

query: this fetch method returns all resultset as an array of rows, and each row contain the pair of key val with the name of the field and the value of the field

column: this fetch method returns an array where each element is each value of the field of the each row, this is useful when for example do you want to get all ids of a query, with this method you can obtain an array with each value of the array is an id of the resultset

concat: this fetch method is an special mode intended to speed up the retrieve of large arrays, this is useful when you want to get all ids of a query and you want to get a big sized array, in this case, is more efficient to get an string separated by commas with all ids instead of an array where each element is an id

3.6.6   DB Disconnect

```
public function db_disconnect()
```

This function close the database connection and sets the link to null

Chapter 4

Javascript

4.1 Application helper module

```
web/js/app.js
```

This fie contains useful functions related to the main application, at the end of this file you can see the main code that launch the application to be executed from the browser

4.1.1 Application helper object

```
saltos.app = {};
```

This object stores all application functions and data

4.1.2 Alert function

```
saltos.app.alert = (title, message, extra)
```

This function tries to implement an alert box, the main difference between the tipical alert is that this alert allow to you to specify the title and a more complex message, but it only shows one button to close it.

- @title => title of the alert modal dialog
- @message => message of the alert modal dialog
- @extra => object with array of buttons and color

4.1.3 Toast function

```
saltos.app.toast = (title, message, extra)
```

This function tries to implement a toast notice.

- @title => title of the toast
- @message => message of the toast
- @extra => object with array of buttons and color

107

4.1.4 Show error helper

```
saltos.app.show_error = error
```

This function allow to show a modal dialog with de details of an error

4.1.5 Check response helper

```
saltos.app.check_response = response
```

This function is intended to process the response received by saltos.core.ajax and returns if an error is detected in the response.

4.1.6 Send request helper

```
saltos.app.send_request = hash
```

This function allow to send requests to the server and process the response

4.1.7 Process response helper

```
saltos.app.process_response = response
```

This function process the responses received by the send request

4.1.8 Form constructor helper object

```
saltos.app.form = {};
```

This object allow to the constructor to use a rational structure for a quick access of each helper

4.1.9 Data helper object

```
saltos.app.__form = {
```

This object allow to the app to store the data of the fields map

4.1.10 Form data helper

```
saltos.app.form.data = data
```

This function sets the values of the request to the objects placed in the document, too as bonus extra, it tries to search the field spec in the array to update the value of the field spec to allow that the get_data can differ between the original data and the modified data.

4.1.11 Data template helper

```
saltos.app.form.__data_template_helper = (template_id, data, index)
```

This function allow to convert the data object that contains the values for the fields idenfidied by the keys of the associative array into a data object with the keys ready to be used by the fields of a template, this fields are of the follow structure: TEMPLATE_ID#ID#INDEX

- @template_id => the template identity used in the spec
- @data => the object that contains the data associated to the row
- @index => the index used in all fields of the template

4.1.12 Layout template helper

```
saltos.app.form.__layout_template_helper = (template_id, index)
```

This function returns the template identified by the template_id for the specified index, ready to be used by the saltos.app.form.layout function.

- @template_id => the template identity used in the spec
- @index => the index used in all fields of the template

4.1.13 Form layout helper

```
saltos.app.form.layout = (layout, extra)
```

This function process the layout command, its able to process nodes as container, row, col and div and all form_field defined in the bootstrap file.

Each objects as container, rows and cols, have two modes of works:

1) normal mode => requires that the user specify all layout, container, row, col and fields.

2) auto mode => only requires set auto='true' to the container, row or col, and with this, all childrens of the node are created inside the correct element, if the auto appear in the container then a container, a row and one col for each field are created, if the auto appear in the row then a row with one col for each field are created, if the auto appear in the col then one col for each field are created, this mode allow to optimize and speedup the creation of each screen by allow to reduce the amount of xml needed to define each screen but allowing to be a lot of specific and define from all to zero auto features.

Notes:

This function add the fields to the saltos.app.__form.fields, this allow to the saltos.app.get_data can retrieve the desired information of the fields.

4.1.14 Form layout auto helper object

```
saltos.app.form.__layout_auto_helper = {};
```

This object stores the functions used as layout_auto_helper for containers, rows and cols

4.1.15   Form layout auto helper for containers

```
saltos.app.form.__layout_auto_helper.container = layout
```

This functions implements the auto feature used by the layout function, allow to specify the follow arguments:

- @id => defines the id used by the container element
- @container_class => defines the class used by the container element
- @container_style => defines the style used by the container element
- @row_class => defines the class used by the row element
- @row_style => defines the style used by the row element
- @col_class => defines the class used by the col element
- @col_style => defines the style used by the col element

4.1.16   Form layout auto helper for rows

```
saltos.app.form.__layout_auto_helper.row = layout
```

This functions implements the auto feature used by the layout function, allow to specify the follow arguments:

- @id => defines the id used by the container element
- @row_class => defines the class used by the row element
- @row_style => defines the style used by the row element
- @col_class => defines the class used by the col element
- @col_style => defines the style used by the col element

4.1.17   Form layout auto helper for cols

```
saltos.app.form.__layout_auto_helper.col = layout
```

This functions implements the auto feature used by the layout function, allow to specify the follow arguments:

- @col_class => defines the class used by the col element
- @col_style => defines the style used by the col element

Notes:

This function not allow to specify the id because in this functions, each object of the layout is embedded inside of a col object and is not suitable to put an id attribute to each col without repetitions.

4.1.18   Form style helper

```
saltos.app.form.style = data
```

This function allow to specify styles, you can use the inline of file key to specify what kind of usage do you want to do.

Note that as some part of this code appear in the core.require function, we have decided to replace it by a call to the saltos.core.require

- @data => the object that contains the styles requirements (can be file or inline)

4.1.19 Form javascript helper

```
saltos.app.form.javascript = data
```

This function allow to specify scripts, you can use the inline of file key to specify what kind of usage do you want to do.

Note that as some part of this code appear in the core.require function, we have decided to replace it by a call to the saltos.core.require

- @data => the object that contains the javascript requirements (can be file or inline)

4.1.20 Form title helper

```
saltos.app.form.title = title
```

This function sets the document title, too it checks the existence of the x-powered-by header received in the ajax calls and stored in the saltos object to be used in the last part of the title.

- @title => The title that you want to set in the page

4.1.21 Form screen helper

```
saltos.app.form.screen = action
```

This function adds and removes the spinner to emulate the loading effect screen, too is able to clear the screen by removing all contents of the body

- @action => use loading, unloading or clear to execute the desired action

4.1.22 Form navbar helper

```
saltos.app.form.navbar = navbar
```

This function create a navbar with their menus on the top of the page, more info about this feature in the bootstrap documentation

- @navbar

4.1.23 Source helper

```
saltos.app.__source_helper = field
```

This function is intended to provide an asynchronous sources for a field, using the source attribute, you can program an asynchronous ajax request to retrieve the data used to create the field.

This function is used in the fields of type table, alert, card and chartjs, the call of this function is private and is intended to be used as a helper from the builders of the previous types opening another way to pass arguments.

- @id => the id used to set the reference for to the object

- @type => the type used to set the type for to the object

- @source => data source used to load asynchronously the contents of the table (header, data, footer and divider)

Notes:

At the end of the object replacement, the load event is triggered to the old object to notify that the update was finished.

4.1.24   Get data

```
saltos.app.get_data = full
```

This function retrieves the data of the fields in the current layout. to do this it uses the saltos.app.__form.fields that contains the list of all used fields in the layout, this function can retrieve all fields or only the fields that contains differences between the original data and the current data.

- @full => boolean to indicate if you want the entire form or only the differences

4.1.25   Get data ids helper

```
saltos.app.__get_data_ids_helper = data
```

This function allow to retrieve the needed ids in the fields used as template

- @data => the contents of the object with data

Notes:

The main idea of this function is to add the id fields if it is needed, for example if some field of the template is modified and their value contains a different value that void.

4.1.26   Parse data

```
saltos.app.parse_data = data
```

This function allow to join in an object the values that share the same prefix part of the key, for example, if you have an object with two ventries (a.a and a.b), then the resulted value will be an object a with two entries (a and b).

- @data => the contents of the object with data

Notes:

This function is used to allow the specification of multiples parameters, for example, in the navbar widget where the widget expects an object for the brand configuration, too is used by get_data to separate in a more pretty structure some fields as the details used in the invoices.

4.1.27 Checkbox ids

```
saltos.app.checkbox_ids = obj
```

Retrieve the selected checkboxes contained in an obj, usefull for the checkboxes that appear in the bootstrap table widget.

- @obj => the object that contains the checkboxes, generally the table widget

4.1.28 Check form

```
saltos.app.check_required = ()
```

This function tries to check if all required fields contain data, if the required field are right, the is-valid class will be applied to all required elements and true is returned, otherwise the is-invalid class will be added to the void required elements and false is returned.

4.1.29 Form disabled

```
saltos.app.form_disabled = bool
```

This function disables all elements of the form, it is intended to be used when you need to do screen for view mode.

4.1.30 Form readonly

```
saltos.app.form_readonly = bool
```

This function set all elements of the form as readonly, it is intended to be used when you need to do screen for view mode.

4.1.31 Form helper

```
saltos.app.__form_helper = (attr, bool)
```

This function is a helper used by the form_disabled and form_readonly functions

4.1.32 ParentNode Search helper

```
saltos.app.parentNode_search = (obj, search)
```

This function helps the user interface to search for parentNodes that can be identified by some class in the classList, it is intended to prevent the call of the parentNode in locations where sometimes can contains a different structure, for example, when you want to get the col div that contains a button with and without labels, depending on the usage of the label, the component can contains a different structure and you may need more or less parentNode calls, thanks to this function, the calls can be automated returning the correct object of the structure with independence of the source of the search.

- @obj => the initial obj where do you want to do the search

- @search => the class name that the parentNode destination must contains

Notes:

As you can see, this search can be performed only by 100 times to prevent infinites loop in case of not found the search pattern.

4.1.33   Main code

This is the code that must to be executed to initialize all requirements of this module

4.2   Authentication helper module

```
web/js/auth.js
```

This file contains all code needed to manage the hash feature, includes the code to process the onhashchange and too, includes the code to get and set the hash value, too includes all code to manage tokens and to do authentications with all features suck as the main authentication using a user and password pair, the reauthtoken, the deauthtoken and the checktoken to control it.

4.2.1   Hash helper object

```
saltos.hash = {};
```

This object stores all hash functions to get, set and trigger a change

4.2.2   Get hash

```
saltos.hash.get = ()
```

Function intended to return the current hash without the pillow

4.2.3   Set hash

```
saltos.hash.set = hash
```

Function intended to replace the hash in the current url, adds the pilow if it is not found in the hash argument

- @hash => this must contain the hash with or without the pillow

Notes:

The # char is added by default if it is not found in the hash, additionally the dot is added to force to remove all chars before the # char in the document.location

4.2.4   Add hash

```
saltos.hash.add = hash
```

Function intended to add a hash in the current history, adds the pillow if it is not found in the hash argument

- @hash => this must contain the hash with or without the pillow

Notes:

The # char is added by default if it is not found in the hash, additionally the dot is added to force to remove all chars before the # char in the document.location

4.2.5 Change trigger

```
saltos.hash.trigger = ()
```

This function triggers the hashchange event to execute the onhashchange

4.2.6 Hash change management

```
saltos.hash.onhashchange = event
```

This function allow to SaltOS to update the contents when hash change

4.2.7 Hash change management

```
window.onhashchange = saltos.hash.onhashchange;
```

Attach the hash change management function to the window

4.2.8 Token helper object

```
saltos.token = {};
```

This object stores all token functions to get and set data using the localStorage

4.2.9 Get token function

```
saltos.token.get = ()
```

This function returns the token stored in the localStorage

4.2.10 Get expires function

```
saltos.token.get_expires = ()
```

This function returns the expires stored in the localStorage

4.2.11 Get autorenew function

```
saltos.token.get_autorenew = ()
```

This function returns the autorenew stored in the localStorage

4.2.12 Get autocheck function

```
saltos.token.get_autocheck = ()
```

This function returns the autocheck stored in the localStorage

4.2.13 Set token and expires

```
saltos.token.set = response
```

This function store the token and expires in the localStorage

- @response => the object that contains the follow parameters:
- @token => the token that you want to store in the localStorage
- @expires => the expires of the token that you want to store in the localStorage
- @autorenew => the autorenew of the token that you can use to force an autorenew
- @autocheck => the autocheck of the token that you can use to poll the autorenew

4.2.14 Unset token and expires

```
saltos.token.unset = ()
```

This function removes the token and expires in the localStorage

4.2.15 Authentication helper object

```
saltos.authenticate = {};
```

This object stores all authentication functions to get access, renew tokens to maintain the access and the deauthtoken to close the access

4.2.16 Authenticate token function

```
saltos.authenticate.authtoken = (user, pass)
```

This function uses the authtoken action to try to authenticate an user with the user/pass credentials passed by argument.

- @user => username used to the authentication process
- @pass => password used to the authentication process

4.2.17 Re-authenticate token function

```
saltos.authenticate.reauthtoken = ()
```

This function uses the reauthtoken action to try to re-authenticate an user with the token credentials.

4.2.18 De-authenticate token function

```
saltos.authenticate.deauthtoken = ()
```

This function uses the deauthtoken action to try to de-authenticate an user with the token credentials.

4.2.19 Check token function

```
saltos.authenticate.checktoken = ()
```

This function uses the checktoken action to check the validity of the current token.

4.2.20 Re-authenticate helper function

```
saltos.authenticate.checkrenew = ()
```

This function checks the reminder of the token's expires and if is needed, execute the renew token action.

4.2.21 Variable used to store the timer

```
saltos.authenticate.__autorenew_timer = null;
```

This variable must contains the timer of the auto renew token feature

4.2.22 Auto-renew helper function

```
saltos.authenticate.autorenew = on_off
```

This function allow to enable or disable the auto renew token feature, can receive an argument to specify if it must to enable or disable the feature, it is intended to be used when set or unset the token.

- @on_off => the parameter to indicates if you want to enable or disable the feature

4.2.23 Window communication helper object

```
saltos.window = {};
```

This object stores all windows communications functions to send and listen messages

4.2.24 Open window

```
saltos.window.open = url
```

This function is intended to open new tabs in the window, at the moment only is a wrapper to the window.open but in a future, can add more features

- @url => the url of the page to load

4.2.25 Close window

```
saltos.window.close = ()
```

This function is intended to close the current window

4.2.26 Listeners helper object

```
saltos.window.listeners = {};
```

This object stores all listeners added by the set_listeners and used by the onstorage

4.2.27 Set listener

```
saltos.window.set_listener = (name, fn)
```

This function allos to set a listener function to the named event

- @name => the name of the event that you want to suscribe
- @fn => callback executed when event named is triggered

4.2.28 Unset listener

```
saltos.window.unset_listener = (name)
```

This function removes a listener from the listeners object

- @name => the name of the event that you want to unsuscribe

4.2.29 Send message

```
saltos.window.send = (name, data, scope)
```

This function allow to send a message to all tabs using the name and data as event and argument of the callback executed.

- @name => the name of the event that you want to send
- @data => the arguments used by the callback function
- @scope => the scope where the event must to be triggered (me, all, other)

Notes:

The usage of the localStorage causes the execution of the onstorage function of the other tabs but not for the tab that send the message, to fix this we are dispatching an event in the current window, this allow that all tabs (including the source of the message sent) receives the notification and executes the listeners if needed

4.2.30 Storage management

```
saltos.window.onstorage = event
```

This function allow to SaltOS to receive the messages sended by other tabs using the localStorage.

4.2.31 Storage management

```
window.onstorage = saltos.window.onstorage;
```

Attach the storage management function to the window

4.3 Bootstrap helper module

```
web/js/bootstrap.js
```

This fie contains useful functions related to the bootstrap widgets, allow to create widgets and other plugins suck as plots or rich editors

4.3.1 Bootstrap helper object

```
saltos.bootstrap = {};
```

This object stores all bootstrap functions and data

4.3.2 Form fields constructor

```
saltos.bootstrap.field = field
```

This function and their helpers, allow the creation of the interface using the bootstrap widgets, the types that can be called are the follow:

- @div => id, class, style
- @container => id, class, style
- @row => id, class, style
- @col => id, class, style
- @text => id, class, PL, value, DS, RO, RQ, AF, AK, datalist, tooltip, label, color, onenter
- @hidden => id, class, PL, value, DS, RO, RQ, AF, AK, tooltip, color, onenter
- @integer => id, class, PL, value, DS, RO, RQ, AF, AK, tooltip, label, color, onenter
- @float => id, class, PL, value, DS, RO, RQ, AF, AK, tooltip, label, color, onenter
- @color => id, class, PL, value, DS, RO, RQ, AF, AK, tooltip, label, color, onenter
- @date => id, class, PL, value, DS, RO, RQ, AF, AK, tooltip, label, color, onenter
- @time => id, class, PL, value, DS, RO, RQ, AF, AK, tooltip, label, color, onenter
- @datetime => id, class, PL, value, DS, RO, RQ, AF, AK, tooltip, label, color, onenter
- @textarea => id, class, PL, value, DS, RO, RQ, AF, AK, rows, tooltip, label, color, height
- @ckeditor => id, class, PL, value, DS, RO, RQ, AF, AK, rows, label, color, height
- @codemirror => id, class, PL, value, DS, RO, RQ, AF, AK, rows, mode, label, color, height
- @iframe => id, class, src, srcdoc, height, label, color

- @select => id, class, DS, RQ, AF, AK, rows, multiple, size, value, tooltip, label, color
- @multiselect => id, class, DS, RQ, AF, AK, rows, multiple, size, value, multiple, tooltip, label, color
- @checkbox => id, class, DS, RO, AK, label, value, tooltip, color
- @switch => id, class, DS, RO, AK, label, value, tooltip, color
- @button => id, class, DS, AK, value, onclick, tooltip, color
- @password => id, class, PL, value, DS, RO, RQ, AF, AK, tooltip, label, color, onenter
- @file => id, class, DS, RQ, AF, AK, multiple, tooltip, label, color
- @link => id, DS, AK, value, onclick, tooltip, label, color
- @label => id, class, label, tooltip, value
- @image => id, class, value, alt, tooltip, width, height, label, color
- @excel => id, class, data, rowHeaders, colHeaders, minSpareRows, contextMenu, rowHeaderWidth, colWidths, label, color
- @pdfjs => id, class, value, label, color
- @table => id, class, header, data, footer, value, label, color
- @alert => id, class, title, text, body, value, label, color
- @card => id, image, alt, header, footer, title, text, body, value, label, color
- @chartjs => id, mode, data, value, label, color
- @tags => id, class, PL, value, DS, RO, RQ, AF, AK, datalist, tooltip, label, color
- @gallery => id, class, label, images, color
- @placeholder => id, color

Notes:

To do more small the previous list, we have used the follow abreviations:

- @PL => placeholder
- @DS => disabled
- @RO => readonly
- @RQ => required
- @AF => autofocus
- @AK => accesskey

The saltos.bootstrap.__field object is part of this constructor and act with the constructor as a helper, the idea is that the user must to call the constructor and the helpers are only for internal use.

4.3.3 Form_field constructor helper object

```
saltos.bootstrap.__field = {};
```

This object allow to the constructor to use a rational structure for a quick access of each helper

4.3.4 Div constructor helper

```
saltos.bootstrap.__field.div = field
```

This function returns an object of the type class by default, you can pass the class argument in the field object to specify what kind of class do you want to use.

- @id => the id used by the object

- @class => the class used in the div object

- @style => the style used in the div object

4.3.5 Container constructor helper

```
saltos.bootstrap.__field.container = field
```

This function returns an object of the container-fluid class by default, you can pass the class argument in the field object to specify what kind of container do you want to do.

- @id => the id used by the object

- @class => the class used in the div object

- @style => the style used in the div object

4.3.6 Row constructor helper

```
saltos.bootstrap.__field.row = field
```

This function returns an object of the row class by default, you can pass the class argument in the field object to specify what kind of row do you want to do.

- @id => the id used by the object

- @class => the class used in the div object

- @style => the style used in the div object

4.3.7 Col constructor helper

```
saltos.bootstrap.__field.col = field
```

This function returns an object of the col class by default, you can pass the class argument in the field object to specify what kind of col do you want to do.

- @id => the id used by the object

- @class => the class used in the div object

- @style => the style used in the div object

4.3.8 Text constructor helper

```
saltos.bootstrap.__field.text = field
```

This function returns an input object of type text, you can pass the same arguments that for the input object of type text

- @id => the id used by the object
- @class => allow to add more classes to the default form-control
- @placeholder => the text used as placeholder parameter
- @value => the value used as value parameter
- @disabled => this parameter raise the disabled flag
- @readonly => this parameter raise the readonly flag
- @required => this parameter raise the required flag
- @autofocus => this parameter raise the autofocus flag
- @tooltip => this parameter raise the title flag
- @accesskey => the key used as accesskey parameter
- @label => this parameter is used as text for the label
- @datalist => array with options for the datalist, used as autocomplete for the text input
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)
- @onenter => the function executed when enter key is pressed

4.3.9 Hidden constructor helper

```
saltos.bootstrap.__field.hidden = field
```

This function returns an input object of type hidden, you can pass the same arguments that for the input object of type text

- @id => the id used by the object
- @class => allow to add more classes to the default form-control
- @placeholder => the text used as placeholder parameter
- @value => the value used as value parameter
- @disabled => this parameter raise the disabled flag
- @readonly => this parameter raise the readonly flag
- @required => this parameter raise the required flag
- @autofocus => this parameter raise the autofocus flag
- @tooltip => this parameter raise the title flag
- @accesskey => the key used as accesskey parameter
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)
- @onenter => the function executed when enter key is pressed

Notes:

This function allow the previous parameters but for hidden inputs, only id and value are usually used, in some cases can be interesting to use the class to identify a group of hidden input

4.3.10 Integer constructor helper

```
saltos.bootstrap.__field.integer = field
```

This function returns an input object of type integer, you can pass the same arguments that for the input object of type text

- @id => the id used by the object
- @class => allow to add more classes to the default form-control
- @placeholder => the text used as placeholder parameter
- @value => the value used as value parameter
- @disabled => this parameter raise the disabled flag
- @readonly => this parameter raise the readonly flag
- @required => this parameter raise the required flag
- @autofocus => this parameter raise the autofocus flag
- @tooltip => this parameter raise the title flag
- @accesskey => the key used as accesskey parameter
- @label => this parameter is used as text for the label
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)
- @onenter => the function executed when enter key is pressed

Notes:

This widget requires the imask library and can be loaded automatically using the require feature:

- @lib/imaskjs/imask.min.js

4.3.11 Float constructor helper

```
saltos.bootstrap.__field.float = field
```

This function returns an input object of type float, you can pass the same arguments that for the input object of type text

- @id => the id used by the object
- @class => allow to add more classes to the default form-control
- @placeholder => the text used as placeholder parameter
- @value => the value used as value parameter
- @disabled => this parameter raise the disabled flag
- @readonly => this parameter raise the readonly flag
- @required => this parameter raise the required flag
- @autofocus => this parameter raise the autofocus flag
- @tooltip => this parameter raise the title flag
- @accesskey => the key used as accesskey parameter

- @label => this parameter is used as text for the label
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)
- @onenter => the function executed when enter key is pressed

Notes:

This widget requires the imask library and can be loaded automatically using the require feature:

- @lib/imaskjs/imask.min.js

4.3.12 Color constructor helper

```
saltos.bootstrap.__field.color = field
```

This function returns an input object of type color, you can pass the same arguments that for the input object of type text

- @id => the id used by the object
- @class => allow to add more classes to the default form-control
- @placeholder => the text used as placeholder parameter
- @value => the value used as value parameter
- @disabled => this parameter raise the disabled flag
- @readonly => this parameter raise the readonly flag
- @required => this parameter raise the required flag
- @autofocus => this parameter raise the autofocus flag
- @tooltip => this parameter raise the title flag
- @accesskey => the key used as accesskey parameter
- @label => this parameter is used as text for the label
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)
- @onenter => the function executed when enter key is pressed

Notes:

Ths color input launch a warning if value is not in the format #rrggbb, for this reason it is set to #000000 if value is void

4.3.13 Date constructor helper

```
saltos.bootstrap.__field.date = field
```

This function returns an input object of type date, you can pass the same arguments that for the input object of type text

- @id => the id used by the object
- @class => allow to add more classes to the default form-control
- @placeholder => the text used as placeholder parameter

- @value => the value used as value parameter

- @disabled => this parameter raise the disabled flag

- @readonly => this parameter raise the readonly flag

- @required => this parameter raise the required flag

- @autofocus => this parameter raise the autofocus flag

- @tooltip => this parameter raise the title flag

- @accesskey => the key used as accesskey parameter

- @label => this parameter is used as text for the label

- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

- @onenter => the function executed when enter key is pressed

4.3.14   Time constructor helper

```
saltos.bootstrap.__field.time = field
```

This function returns an input object of type time, you can pass the same arguments that for the input object of type text

- @id => the id used by the object

- @class => allow to add more classes to the default form-control

- @placeholder => the text used as placeholder parameter

- @value => the value used as value parameter

- @disabled => this parameter raise the disabled flag

- @readonly => this parameter raise the readonly flag

- @required => this parameter raise the required flag

- @autofocus => this parameter raise the autofocus flag

- @tooltip => this parameter raise the title flag

- @accesskey => the key used as accesskey parameter

- @label => this parameter is used as text for the label

- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

- @onenter => the function executed when enter key is pressed

4.3.15   Datetime constructor helper

```
saltos.bootstrap.__field.datetime = field
```

This function returns an input object of type datetime, you can pass the same arguments that for the input object of type text

- @id => the id used by the object

- @class => allow to add more classes to the default form-control

- @placeholder => the text used as placeholder parameter
- @value => the value used as value parameter
- @disabled => this parameter raise the disabled flag
- @readonly => this parameter raise the readonly flag
- @required => this parameter raise the required flag
- @autofocus => this parameter raise the autofocus flag
- @tooltip => this parameter raise the title flag
- @accesskey => the key used as accesskey parameter
- @label => this parameter is used as text for the label
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)
- @onenter => the function executed when enter key is pressed

4.3.16 Textarea constructor helper

```
saltos.bootstrap.__field.textarea = field
```

This function returns a textarea object with the autogrow plugin enabled

- @id => the id used by the object
- @class => allow to add more classes to the default form-control
- @placeholder => the text used as placeholder parameter
- @value => the value used as value parameter
- @disabled => this parameter raise the disabled flag
- @readonly => this parameter raise the readonly flag
- @required => this parameter raise the required flag
- @autofocus => this parameter raise the autofocus flag
- @tooltip => this parameter raise the title flag
- @accesskey => the key used as accesskey parameter
- @label => this parameter is used as text for the label
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)
- @height => the height used as style.minHeight parameter

Notes:

This widget requires the autoheight library and can be loaded automatically using the require feature:

- @lib/autoheight/autoheight.min.js

4.3.17 Ckeditor constructor helper

```
saltos.bootstrap.__field.ckeditor = field
```

This function returns a textarea object with the ckeditor plugin enabled

- @id => the id used by the object
- @class => allow to add more classes to the default form-control
- @placeholder => the text used as placeholder parameter
- @value => the value used as value parameter
- @disabled => this parameter raise the disabled flag
- @readonly => this parameter raise the readonly flag
- @required => this parameter raise the required flag
- @autofocus => this parameter raise the autofocus flag
- @tooltip => this parameter raise the title flag
- @accesskey => the key used as accesskey parameter
- @label => this parameter is used as text for the label
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)
- @height => the height used as style.minHeight parameter

Notes:

This widget requires the ckeditor library and can be loaded automatically using the require feature:

- @lib/ckeditor/ckeditor.min.js

4.3.18 Codemirror constructor helper

```
saltos.bootstrap.__field.codemirror = field
```

This function returns a textarea object with the codemirror plugin enabled

- @mode => used to define the mode parameter of the codemirror
- @id => the id used by the object
- @class => allow to add more classes to the default form-control
- @placeholder => the text used as placeholder parameter
- @value => the value used as value parameter
- @disabled => this parameter raise the disabled flag
- @readonly => this parameter raise the readonly flag
- @required => this parameter raise the required flag
- @autofocus => this parameter raise the autofocus flag
- @tooltip => this parameter raise the title flag
- @accesskey => the key used as accesskey parameter
- @label => this parameter is used as text for the label
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)
- @height => the height used as style.minHeight parameter

Notes:

This widget requires the codemirror library and can be loaded automatically using the require feature:

- @lib/codemirror/codemirror.min.css
- @lib/codemirror/codemirror.min.js

4.3.19 Iframe constructor helper

```
saltos.bootstrap.__field.iframe = field
```

This function returns an iframe object, you can pass the follow arguments:

- @id => the id used by the object
- @src => the value used as src parameter
- @srcdoc => the value used as srcdoc parameter
- @class => allow to add more classes to the default form-control
- @height => the height used as style.minHeight parameter
- @label => this parameter is used as text for the label
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

4.3.20 Select constructor helper

```
saltos.bootstrap.__field.select = field
```

This function returns a select object, you can pass the follow arguments:

- @id => the id used by the object
- @class => allow to add more classes to the default form-select
- @disabled => this parameter raise the disabled flag
- @required => this parameter raise the required flag
- @autofocus => this parameter raise the autofocus flag
- @multiple => this parameter enables the multiple selection feature of the select
- @size => this parameter allow to see the options list opened with n (size) entries
- @value => the value used to detect the selected option
- @tooltip => this parameter raise the title flag
- @accesskey => the key used as accesskey parameter
- @rows => this parameter contains the list of options, each option must be an object with label and value entries
- @label => this parameter is used as text for the label
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

4.3.21   Multiselect constructor helper

```
saltos.bootstrap.__field.multiselect = field
```

This function returns a multiselect object, you can pass the follow arguments:

- @id => the id used by the object
- @class => allow to add more classes to the default form-select
- @disabled => this parameter raise the disabled flag
- @size => this parameter allow to see the options list opened with n (size) entries
- @value => the value used as src parameter
- @tooltip => this parameter raise the title flag
- @accesskey => the key used as accesskey parameter
- @rows => this parameter contains the list of options, each option must be an object with label and value entries
- @label => this parameter is used as text for the label
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

Notes:

This widget is created joinin 2 selects and 2 buttons, the user must get the value using the hidden input that is builded using the original id passed by argument.

TODO: detected a bug with this widget in chrome in mobile browsers

4.3.22   Checkbox constructor helper

```
saltos.bootstrap.__field.checkbox = field
```

This function returns a checkbox object, you can pass the follow arguments:

- @id => the id used by the object
- @class => allow to add more classes to the default form-check
- @disabled => this parameter raise the disabled flag
- @readonly => this parameter raise the readonly flag
- @label => this parameter is used as label for the checkbox
- @value => this parameter is used to check or unckeck the checkbox, the value must contain a number that raise as true or false in the if condition
- @tooltip => this parameter raise the title flag
- @accesskey => the key used as accesskey parameter
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

Notes:

This widget returns their value by setting a zero or one (0/1) value on the value of the input.

4.3.23 Switch constructor helper

```
saltos.bootstrap.__field.switch = field
```

This function returns a switch object, you can pass the follow arguments:

- @id => the id used by the object
- @class => allow to add more classes to the default form-check and form-switch
- @disabled => this parameter raise the disabled flag
- @readonly => this parameter raise the readonly flag
- @label => this parameter is used as label for the switch
- @value => this parameter is used to check or unckeck the switch, the value must contain a number that raise as true or false in the if condition
- @tooltip => this parameter raise the title flag
- @accesskey => the key used as accesskey parameter
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

Notes:

This widget uses the checkbox constructor

4.3.24 Button constructor helper

```
saltos.bootstrap.__field.button = field
```

This function returns a button object, you can pass the follow arguments:

- @id => the id used by the object
- @class => allow to add more classes to the default form-select
- @disabled => this parameter raise the disabled flag
- @autofocus => this parameter raise the autofocus flag
- @value => value to be used as text in the contents of the buttons
- @onclick => callback function that is executed when the button is pressed
- @tooltip => this parameter raise the title flag
- @accesskey => the key used as accesskey parameter
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

Notes:

The buttons adds the focus-ring classes to use this new feature that solves issues suck as hidden focus when you try to focus a button inside a modal, for example.

4.3.25 Password constructor helper

```
saltos.bootstrap.__field.password = field
```

This function returns an input object of type password, you can pass some arguments as:

- @id => the id used by the object

- @class => allow to add more classes to the default form-control

- @placeholder => the text used as placeholder parameter

- @value => the value used as value parameter

- @disabled => this parameter raise the disabled flag

- @readonly => this parameter raise the readonly flag

- @required => this parameter raise the required flag

- @autofocus => this parameter raise the autofocus flag

- @tooltip => this parameter raise the title flag

- @accesskey => the key used as accesskey parameter

- @label => this parameter is used as text for the label

- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

Notes:

This widget add an icon to the end of the widget with an slashed eye, this allow to see the entered password to verify it, in reality, this button swaps the input between password and text type, allowing to do visible or not the contents of the input

I'm using previousElementSibling instead of previousSibling because between the input and the button, exists a new line that is identified as another previousSibling, but not as an element

4.3.26   File constructor helper

```
saltos.bootstrap.__field.file = field
```

This function returns an input object of type password, you can pass some arguments as:

- @id => the id used by the object

- @class => allow to add more classes to the default form-control

- @disabled => this parameter raise the disabled flag

- @required => this parameter raise the required flag

- @autofocus => this parameter raise the autofocus flag

- @multiple => this parameter raise the multiple flag, intended to select more files at time

- @tooltip => this parameter raise the title flag

- @accesskey => the key used as accesskey parameter

- @label => this parameter is used as text for the label

- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

Notes:

This control allow to select file from the tradicional input, and automatically, send it to the server using the addfiles action, add a row in the widget's table to show information about the new file and allow too to delete

it using the trash button.

To get the data, the controls store each file information in each added row of the table and in addition, too join all information in a data structure of the input of type file.

The difference between this control and the older controls is that they send the files to the server and store the information related to the file on the server to be processed after the real upload action.

4.3.27 Link constructor helper

```
saltos.bootstrap.__field.link = field
```

This function creates a field similar of text but with the appearance of a link using a button, the object can receive the follow arguments:

- @label => this parameter is used as text for the label
- @id => the id used by the object
- @disabled => this parameter raise the disabled flag
- @value => the value is conveted as label to be used in the button with the appearance of a link
- @onclick => callback function that is executed when the button is pressed
- @accesskey => the key used as accesskey parameter
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

Notes:

This object is not a real link, it's a button that uses the btn-link class to get the link appearance

4.3.28 Label constructor helper

```
saltos.bootstrap.__field.label = field
```

This function returns a label object, you can pass some arguments as:

- @id => the id used to set the reference for to the object
- @class => allow to add more classes to the default form-label
- @label => this parameter is used as text for the label
- @tooltip => this parameter raise the title flag
- @value => this parameter is used as label when label is void

4.3.29 Image constructor helper

```
saltos.bootstrap.__field.image = field
```

This function returns an image object, you can pass some arguments as:

- @id => the id used to set the reference for to the object
- @class => allow to add more classes to the default img-fluid

- @value => the value used as src parameter

- @alt => this parameter is used as text for the alt parameter

- @tooltip => this parameter raise the title flag

- @label => this parameter is used as text for the label

- @width => this parameter is used as width for the image

- @height => this parameter is used as height for the image

- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

4.3.30   Excel constructor helper

```
saltos.bootstrap.__field.excel = field
```

This function creates and returns an excel object, to do this they use the handsontable library, currently this library uses a privative license, by this reason, we are using the version 6.2.2 that is the latest release published using the MIT license.

This widget can receive the following arguments:

- @id => the id used to set the reference for to the object

- @class => allow to set the class to the div object used to allocate the widget

- @data => this parameter must contain a 2D matrix with the data that you want to show in the sheet

- @rowHeaders => can be an array with the headers that you want to use instead the def numbers

- @colHeaders => can be an array with the headers that you want to use instead the def letters

- @minSpareRows => can be a number with the void rows at the end of the sheet

- @contextMenu => can be a boolean with the desired value to allow or not the provided context menu of the widget

- @rowHeaderWidth => can be a number with the width of the headers rows

- @colWidths => can be an array with the widths of the headers cols

- @label => this parameter is used as text for the label

- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

Notes:

You can get the values after to do changes by accessing to the data of the div used to create the widget.

This widget requires the handsontable library and can be loaded automatically using the require feature:

- @lib/handsontable/handsontable.full.min.css

- @lib/handsontable/handsontable.full.min.js

4.3.31   Pdfjs constructor helper

```
saltos.bootstrap.__field.pdfjs = field
```

This function creates and returns a pdfviewer object, to do this they use the pdf.js library.

- @id => the id used to set the reference for to the object

- @class => allow to set the class to the div object used to allocate the widget

- @src => the file that contains the pdf document

- @srcdoc => the data that contains the pdf document

- @label => this parameter is used as text for the label

- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

Notes:

This widget requires the pdfjs library and can be loaded automatically using the require feature:

- @lib/pdfjs/pdf_viewer.min.css,

- @lib/pdfjs/pdf.min.mjs

- @lib/pdfjs/pdf_viewer.min.mjs

- @lib/pdfjs/pdf.worker.min.mjs

The last file (the worker) is loaded by the library and not by SaltOS, is for this reason that this file not appear in the next requires

4.3.32   Table constructor helper

```
saltos.bootstrap.__field.table = field
```

Returns a table using the follow params:

- @id => the id used to set the reference for to the object

- @class => allow to add more classes to the default table table-striped table-hover

- @header => array with the header to use

- @data => 2D array with the data used to mount the body table

- @footer => array with the footer to use

- @checkbox => add a checkbox in the first cell of each row, for mono or multi selection

- @dropdown => a boolean value to force the usage of the dropdown feature, void for auto detection

- @label => this parameter is used as text for the label

- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

- @actions => each row of the data can contain an array with the actions of each row

Each action can contain:

- @app => the application that must be used to check the permissions

- @action => the accion that must to be used to check the permissions

- @value => the text used as label in the button of the action

- @icon => the icon used in the button of the action

- @tooltip => the tooltip used in the button of the action

- @onclick => the onclick function that receives as argument the url to access the action

Notes:

This function defines the yellow color used for the hover and active rows.

The header field must be an object with the labels, types, aligns, ..., of each field, if the header is ommited, then the data will be painted using the default order of the data without filters, the recomendation is to use header to specify which fields must to be painted, the order, the type and the alignment.

The actions will be added using a dropdown menu if more than one action appear in the the row data, the idea of this feature is to prevent that the icons uses lot of space of the row data, and for this reason, it will define the dropdown variable that enables or not the contraction feature

The elements of the data cells can contains an object with the field specification used to the saltos.bootstrap.field constructor, it is usefull to convert some fields to inputs instead of simple text, too is able to use the type attribute in the header specification to identify if you want to use a column with some special type as for example, the icons

4.3.33 Alert constructor helper

```
saltos.bootstrap.__field.alert = field
```

This component allow to set boxes type alert in the contents, only requires:

- @id => the id used to set the reference for to the object
- @class => allow to add more classes to the default alert
- @title => title used in the body of the card, not used if void
- @text => text used in the body of the card, not used if void
- @body => this option allow to specify an specific html to the body of the card, intended to personalize the body's card
- @close => boolean to specify if you want to add the dismissible option to the alert
- @label => this parameter is used as text for the label
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

Note:

I have added the dismissible option using the close attribute, too I have added a modification for the style to allow the content to use the original size of the alert, in a future, I don't know if I maintain this or I remove it, but at the moment, this is added by default

4.3.34 Card constructor helper

```
saltos.bootstrap.__field.card = field
```

This functions creates a card with a lot of options:

- @id => the id used to set the reference for to the object
- @image => image used as top image in the card, not used if void
- @alt => alt text used in the top image if you specify an image
- @header => text used in the header, not used if void
- @footer => text used in the footer, not used if void

135

- @title => title used in the body of the card, not used if void

- @text => text used in the body of the card, not used if void

- @body => this option allow to specify an specific html to the body of the card, intended to personalize the body's card

- @label => this parameter is used as text for the label

- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

4.3.35   Chart.js constructor helper

```
saltos.bootstrap.__field.chartjs = field
```

This function creates a chart using the chart.js library, to do this requires de follow arguments:

- @id => the id used by the object

- @mode => to specify what kind of plot do you want to do: can be bar, line, doughnut, pie

- @data => the data used to plot the graph, see the data argument used by the graph.js library

- @label => this parameter is used as text for the label

Notes:

To be more practice and for stetic reasons, I'm adding to all datasets the borderWidth = 1

This widget requires the chartjs library and can be loaded automatically using the require feature:

- @lib/chartjs/chart.umd.min.js

4.3.36   Tags constructor helper

```
saltos.bootstrap.__field.tags = field
```

This function creates a text input that allow to manage tags, each tag is paint as a badge and each tag can be deleted, the result is stored in a text using a comma separated values

- @id => the id used by the object

- @value => comma separated values

- @datalist => array with options for the datalist, used as autocomplete for the text input

- @label => this parameter is used as text for the label

- @class => allow to add more classes to the default form-control

- @placeholder => the text used as placeholder parameter

- @disabled => this parameter raise the disabled flag

- @readonly => this parameter raise the readonly flag

- @required => this parameter raise the required flag

- @autofocus => this parameter raise the autofocus flag

- @tooltip => this parameter raise the title flag

- @accesskey => the key used as accesskey parameter

- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

Notes:

This object creates a hidden input, a text input with/without a datalist, and a badge for each value, and requires the arguments of the specific widgets used in this widget

4.3.37 Gallery constructor helper

```
saltos.bootstrap.__field.gallery = field
```

This function returns a gallery object, you can pass some arguments as:

- @id => the id used to set the reference for to the object
- @class => allow to add more classes to the default img-fluid
- @label => this parameter is used as text for the label
- @images => the array with images, each image can be an string or object
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

This widget requires venobox, masonry and imagesloaded

This widget requires the venobox, masonry and imagesloaded libraries and can be loaded automatically using the require feature:

- @lib/venobox/venobox.min.css
- @lib/venobox/venobox.min.js
- @lib/masonry/masonry.pkgd.min.js
- @lib/imagesloaded/imagesloaded.pkgd.min.js

4.3.38 Placeholder helper

```
saltos.bootstrap.__field.placeholder = field
```

This function returns a grey area that uses all space with the placeholder glow effect

- @id => id used in the original object, it must be replaced when the data will be available

4.3.39 Private text constructor helper

```
saltos.bootstrap.__text_helper = field
```

This function returns an input object of type text, you can pass some arguments as:

- @id => the id used by the object
- @class => allow to add more classes to the default form-control
- @placeholder => the text used as placeholder parameter
- @value => the value used as value parameter
- @disabled => this parameter raise the disabled flag

137

- @readonly => this parameter raise the readonly flag
- @required => this parameter raise the required flag
- @autofocus => this parameter raise the autofocus flag
- @tooltip => this parameter raise the title flag
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)
- @onenter => the function executed when enter key is pressed

Notes:

This function is intended to be used by other helpers of the form_field constructor

4.3.40 Private textarea constructor helper

```
saltos.bootstrap.__textarea_helper = field
```

This function returns a textarea object, you can pass the follow arguments:

- @id => the id used by the object
- @class => allow to add more classes to the default form-control
- @placeholder => the text used as placeholder parameter
- @value => the value used as value parameter
- @disabled => this parameter raise the disabled flag
- @readonly => this parameter raise the readonly flag
- @required => this parameter raise the required flag
- @autofocus => this parameter raise the autofocus flag
- @tooltip => this parameter raise the title flag
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

Notes:

This function is intended to be used by other helpers of the form_field constructor

4.3.41 Private tooltip constructor helper

```
saltos.bootstrap.__tooltip_helper = obj
```

This function is intended to enable the tooltip in the object, too it try to do some extra features: program that only show the tooltip when hover and hide when will get the focus or get the click event

- @obj => the object that you want to enable the tooltip feature

4.3.42 Label helper

```
saltos.bootstrap.__label_helper = field
```

This function is a helper for label field, it is intended to returns the label object or a void string, this is because if no label is present in the field argument, then an empty string is returned, in the reception of the result, generally this is added to an object and it is ignored because an empty string is not an element, this thing is used by the optimizer to removes the unnecessary envelopment

- @field => the field that contains the label to be added if needed

4.3.43 Label Combine

```
saltos.bootstrap.__label_combine = (field, old)
```

This function combine the label with the object, to do it, tries to create a new container object to put the label and the passed object, and then tries to optimize to detect if the label is void

- @field => the field that contains the label
- @old => the object

Notes:

This function acts as helper to add a label by the constructors that not implement any specific label container, in the other cases, each constructor must to implement their code because each case is different

4.3.44 Onclick helper

```
saltos.bootstrap.__onclick_helper = (obj, fn)
```

This function is a helper function that adds the onclick event listener to the obj using the correct way to do it, to do it, checks the type of fn.

- @obj => the object where you want to add the onclick event
- @fn => the function that must be executed when onclick

4.3.45 Onenter helper

```
saltos.bootstrap.__onenter_helper = (obj, fn)
```

This function adds the event and detects the enter key in order to execute fn

- @obj => the object that you want to enable the onenter feature
- @fn => the function executed when the onenter is raised

4.3.46 Menu constructor helper

```
saltos.bootstrap.menu = args
```

This function creates a menu intended to be used in navbar, nabs and tabs

- @class => the class used in the main ul element
- @menu => an array with the follow elements:
- @name => name of the menu

- @icon => icon of the menu

- @disabled => this boolean allow to disable this menu entry

- @active => this boolean marks the option as active

- @onclick => the callback used when the user select the menu

- @dropdown_menu_end => this trick allow to open the dropdown menu from the end to start

- @menu => with this option, you can specify an array with the contents of the dropdown menu

- @name => name of the menu

- @icon => icon of the menu

- @disabled => this boolean allow to disable this menu entry

- @active => this boolean marks the option as active

- @onclick => the callback used when the user select the menu

- @divider => you can set this boolean to true to convert the element into a divider

4.3.47   Navbar constructor helper

```
saltos.bootstrap.navbar = args
```

This component creates a navbar intended to be used as header

- @id => the id used by the object

- @space => boolean to indicate if you want to add the space div

- @brand => contains an object with the name, logo, width and height to be used

- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

- @name => text used in the brand

- @logo => filename of the brand image

- @width => width of the brand image

- @height => height of the brand image

- @items => contains an array with the objects that will be added to the collapse

4.3.48   Modal constructor helper object

```
saltos.bootstrap.__modal = {};
```

This object is used to store the element and the instance of the modal

4.3.49   Modal constructor helper

```
saltos.bootstrap.modal = args
```

This function creates a bootstrap modal and open it, offers two ways of usage:

1) you can pass an string to get a quick action

- @close => this string close the current modal

- @isopen => this string is used to check if some modal is open at the moment

2) you can pass an object with the follow items, intended to open a new modal

- @id => the id used by the object

- @class => allow to add more classes to the default dialog

- @title => title used by the modal

- @close => text used in the close button for aria purposes

- @body => the content used in the modal's body

- @footer => the content used in the modal's footer

- @static => forces the modal to be static (prevent close by clicking outside the modal or by pressing the escape key)

- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

Returns a boolean that indicates if the modal can be open or not

Notes:

This modal will be destroyed (instance and element) when it closes, too is important to undestand that only one modal is allowed at each moment.

4.3.50   Offcanvas constructor helper object

```
saltos.bootstrap.__offcanvas = {};
```

This object is used to store the element and the instance of the offcanvas

4.3.51   Offcanvas constructor helper

```
saltos.bootstrap.offcanvas = args
```

This function creates a bootstrap offcanvas and open it, offers two ways of usage:

1) you can pass an string to get a quick action

- @close => this string close the current modal

- @isopen => this string is used to check if some modal is open at the moment

2) you can pass an object with the follow items, intended to open a new modal

- @id => the id used by the object

- @class => allow to add more classes to the default offcanvas

- @title => title used by the offcanvas

- @close => text used in the close button for aria purposes

- @body => the content used in the offcanvas's body

- @static => forces the offcanvas to be static (prevent close by clicking outside the offcanvas or by pressing the escape key)

- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

Returns a boolean that indicates if the offcanvas can be open or not

Notes:

This offcanvas will be destroyed (instance and element) when it closes, too is important to undestand that only one offcanvas is allowed at each moment.

4.3.52 Toast constructor helper

```
saltos.bootstrap.toast = args
```

This function creates a bootstrap toast and show it, and can accept the follow params:

- @id => the id used by the object
- @class => allow to add more classes to the default toast
- @title => title used by the toast
- @subtitle => small text used by the toast
- @close => text used in the close button for aria purposes
- @body => the content used in the toast's body
- @color => the color of the widget (primary, secondary, success, danger, warning, info, none)

Returns a boolean that indicates if the toast can be created (see the hash note)

Notes:

The toast will be destroyed (instance and element) when it closes.

All toasts are added to a toast-container placed in the body of the document, this container is created automatically if it not exists when the first toast need it.

Each toast includes a hash to prevent the creation of repeated toasts.

This widget requires the md5 library and can be loaded automatically using the require feature:

- @lib/md5/md5.min.js

4.3.53 Accesskey listener

```
saltos.bootstrap.__accesskey_listener = event
```

This function is intended to improve the default accesskey in the object by adding features suck as combinations of keys like alt+f or alt+delete

- @obj => the object that you want to enable the accesskey feature

4.3.54 Accesskey listener

```
document.addEventListener('keydown', saltos.bootstrap.__accesskey_listener)
   ;
```

Attach the accesskey listener function to the keydown event of the document

4.4 Core helper module

```
web/js/core.js
```

This fie contains useful functions related to the core application, provides the low level features for manage errors, logs, manipulates html and DOM objects, manage ajax requests, and more things

4.4.1 Core helper object

```
saltos.core = {};
```

This object stores all core functions and data

4.4.2 Error management

```
saltos.core.onerror = (event, source, lineno, colno, error)
```

This function allow to SaltOS to log in server the javascript errors produced in the client's browser

4.4.3 Error management

```
window.onerror = saltos.core.onerror;
```

Attach the error management function to the window

4.4.4 Log management

```
saltos.core.addlog = msg
```

This function allow to send messages to the addlog of the server side, requires an argument:

- @msg => the message that do you want to log on the server log file

4.4.5 Check params

```
saltos.core.check_params = (obj, params, value)
```

This function allow to prepare parameters to be used by other functions, the main idea is that the other functions can access to properties of an object without getting errors caused by the nonexistence, to do this, checks for the existence of all params in the obj and if some param is not found, then define it using the default value passed:

- @obj => the object that contains the arguments, for example
- @params => an array with the arguments that must to exists
- @value => the default value used if an argument doesn't exists

4.4.6 UniqID

```
saltos.core.uniqid = ()
```

This function generates an unique id formed by the word 'id' and a number that can take values between 0 and 999999, useful when some widget requires an id and the user don't provide it to the widget constructor

4.4.7 When visible

```
saltos.core.when_visible = (obj, fn, args)
```

This function allow to execute some code when the object is visible, useful for third part widgets as ckeditor or codemirror that requires a rendered environemt to initialize their code and paint the widget correctly

- @obj => the object that do you want to monitorize the visibility
- @fn => the callback that you want to execute
- @args => the arguments passed to the callback when execute it

4.4.8 Get keycode

```
saltos.core.get_keycode = event
```

This function allow to get the keycode of a keyboard event detecting the browser

- @event => the event that contains the keyboard data

4.4.9 HTML builder

```
saltos.core.html = (...args)
```

This function allow to create an DOM fragment from a string that contains html code, can work with one or two arguments:

- @type => the type used when create the container element
- @html => contains the html code that you want to use as template

The main use is only using the html argument and omiting the type, in this case, the type used will be a div, but if you want to create a fragment of object, for example as tr or td, you need to specify that the coontainer type used to create the objects must to be a table or tr, is you don't specify the type, the div container creates a breaked portion of the element and they don't works as expected because the DOM builded is bad, you can see this problem in action when work with tables and try to create separate portions of the table as trs or tds.

4.4.10 AJAX

```
saltos.core.ajax = args
```

This function allow to use ajax using the same form that with jQuery without jQuery

- @url => url of the ajax call
- @data => data used in the body of the request

- @method => the method of the request (can be GET or POST, GET by default)

- @success => callback function for the success action (optional)

- @error => callback function for the error action (optional)

- @progress => callback function to monitorize the progress of the upload/download (optional)

- @async => boolean to use the ajax call asynchronously or not, by default is true

- @content_type => the content-type that you want to use in the transfer

- @headers => an object with the headers that you want to send

The main idea of this function is to abstract the usage of the XMLHttpRequest in a simple way as jQuery do but without using jQuery.

4.4.11 Fix key

```
saltos.core.fix_key = arg
```

This function is intended to fix the keys of the objects, this is caused because you can not have 2 repeated keys in an object, to have more entries with the same name, SaltOS add a suffix by adding #num, with this trick, SaltOS is able to process XML files with the same node name and convert it to an array structure, and when convert this to json, the same problem appear and for this reason, exists this function here

- @arg => can be an string or an array of strings and returns the same structure with the keys fixed

4.4.12 Copy object

```
saltos.core.copy_object = arg
```

This function is intended to do copies of objects using as intermediate a json file

- @arg => the object that you want to copy

4.4.13 Optimizer object

```
saltos.core.optimize = obj
```

This function checks an object to see if only contains one children and in this case, returns directly the children instead of the original object, otherwise nothing to do and returns the original object

- @obj => the object to check and optimize

4.4.14 Require helper array

```
saltos.core.__require = [];
```

This array allow to the require feature to control the loaded libraries

4.4.15 Require feature

```
saltos.core.require = file
```

This function allow the other functions to declare their requirements to previously load the desired file intead of create the object and throwing an error.

- @file => the file desired to be loaded

Notes:

This function is intended to load styles (css files) or javascript code (js files), in each case, they uses a different technique, for css the load is asynchronous and for javascript the load will be synchronous.

4.4.16   Eval Bool

```
saltos.core.eval_bool = arg
```

This function returns a boolean depending on the input evaluation, the main idea is to get an string, for example, and determine if must be considered true or false otherwise returns the original argument and send a log message to the console.

The valid inputs are the strings one, zero, void, true, false, on, off, yes and no

- @arg => the value that do you want to evaluates as boolean

Notes:

This function is the same feature that the same function proviced by the backend by the php/autoload-/xml2array.php file with more javascript details as type detection.

4.4.17   toString function

```
saltos.core.toString = arg
```

This function tries to convert to string from any other formats as boolean, number, null, undefined or other type.

4.4.18   Is attr value

```
saltos.core.is_attr_value = data
```

This function return true if the data argument is an object with #attr and value

- @data => the data that wants to check

4.4.19   Join attr value

```
saltos.core.join_attr_value = data
```

This function return an object that contains all elements of the #attr and value

- @data => the data that wants to join

146

4.5 Object helper module

`web/js/object.js`

This file contains the main saltos object

4.5.1 Main object

`var saltos = {};`

This object contains all SaltOS code

4.6 Customers application

`apps/customers/js/app.js`

This application implements the tipical features associated to customers

4.6.1 Main object

`saltos.customers = {};`

This object contains all SaltOS code

4.6.2 TODO

`saltos.customers.initialize_update_list = ()`

TODO

4.6.3 TODO

`saltos.customers.initialize_update_view = ()`

TODO

4.6.4 TODO

`saltos.customers.search = ()`

TODO

4.6.5 TODO

`saltos.customers.reset = ()`

TODO

4.6.6 TODO

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

TODO

4.6.7 TODO

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

TODO

4.6.8 TODO

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

TODO

4.6.9 TODO

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

TODO

4.6.10 TODO

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

TODO

4.7 Dashboard application

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

This application implements the tipical features associated to dashboard

4.7.1 Main object

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

This object contains all SaltOS code

4.7.2 TODO

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

TODO

4.8 Email application

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

This application implements the tipical features associated to emails

4.8.1 Main object

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

This object contains all SaltOS code

4.8.2 Initialize emails app

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

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

4.8.3 TODO

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

TODO

4.8.4 TODO

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

TODO

4.8.5 TODO

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

TODO

4.8.6 TODO

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

TODO

4.8.7 TODO

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

TODO

4.8.8 TODO

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

TODO

4.8.9 TODO

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

TODO

4.8.10 TODO

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

TODO

4.8.11 TODO

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

TODO

Chapter 5

Emails

5.1 Get email library

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

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

5.1.1 Requires section

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

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

5.1.2 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.

5.1.3 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

5.1.4 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

5.1.5   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

5.1.6   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

5.1.7   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

5.1.8   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

5.1.9   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

5.1.10 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

5.1.11 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

5.1.12 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

5.1.13 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

5.1.14 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

5.1.15 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

5.1.16 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

5.1.17 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

5.1.18 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

5.1.19 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

5.1.20 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

5.1.21 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)

5.1.22 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

5.1.23 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

5.1.24 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

5.1.25 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

5.1.26　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

5.1.27　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

5.1.28　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

5.1.29　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

5.1.30　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

5.1.31　Receive

```
function getmail_receive()
```

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

5.1.32 Delete

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

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

- @ids => array with the emails id

5.1.33 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

5.1.34 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

5.1.35 TODO

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

TODO

5.2 Send email library

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

This library provides the necesary functions to send emails.

5.2.1 Used libraries

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

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

5.2.2 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

5.2.3 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

5.2.4 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

5.2.5 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

5.2.6 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

Chapter 6

Javascript

6.1 invoices application

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

This application implements the tipical features associated to invoices

6.1.1 Main object

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

This object contains all SaltOS code

6.1.2 TODO

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

TODO

6.1.3 TODO

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

TODO

6.1.4 TODO

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

TODO

6.1.5 TODO

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

TODO

6.1.6 TODO

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

TODO

6.1.7 TODO

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

TODO

6.1.8 TODO

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

TODO

6.1.9 TODO

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

TODO

6.1.10 TODO

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

TODO

6.1.11 TODO

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

TODO

6.1.12 TODO

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

TODO

6.1.13 TODO

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

TODO

6.1.14 TODO

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

TODO

6.1.15 TODO

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

TODO

6.2 Login application

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

This application implements the tipical features associated to login

6.2.1 Main object

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

This object contains all SaltOS code

6.2.2 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

6.2.3 Access denied

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

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

6.3 Tester application

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

This application implements the tipical features associated to tester

6.3.1 Tester object

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

This object stores all function used by this app

6.3.2 Campo 8

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

TODO

6.3.3 Campo 9

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

TODO

6.3.4 Campo 10

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

TODO

6.3.5 Campo 11

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

TODO

6.3.6 Campo 22

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

TODO

6.4 Tester application

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

This application implements the tipical features associated to tester

6.4.1 Main code

```
(function() {
```

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

Chapter 7

Tester

7.1 TODO

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

7.2 TODO

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

7.3 TODO

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

Chapter 8

Javascript

8.1 Types application

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

This application implements the tipical features associated to types

8.1.1 Main object

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

This object contains all SaltOS code

8.1.2 Initialize types

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

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

8.1.3 TODO

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

TODO

8.1.4 TODO

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

TODO

8.1.5 TODO

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

TODO

8.1.6 TODO

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

TODO

8.1.7 TODO

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

TODO

8.1.8 TODO

```
saltos.types.default = ()
```

TODO

8.1.9 TODO

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

TODO

8.1.10 TODO

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

TODO

8.1.11 TODO

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

TODO

8.2 types application

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

This application implements the tipical features associated to types

8.2.1 Main object

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

This object contains all SaltOS code

8.2.2 TODO

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

TODO

8.2.3 TODO

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

TODO

8.2.4 TODO

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

TODO

8.2.5 TODO

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

TODO

8.2.6 TODO

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

TODO

8.2.7 TODO

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

TODO

8.2.8 TODO

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

TODO

8.2.9 TODO

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

TODO

8.2.10 TODO

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

TODO

8.3 Types application

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

This application implements the tipical features associated to types

8.3.1 Main object

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

This object contains all SaltOS code

8.3.2 Initialize types

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

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

8.3.3 TODO

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

TODO

8.3.4 TODO

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

TODO

8.3.5 TODO

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

TODO

8.3.6 TODO

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

TODO

8.3.7 TODO

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

TODO

8.3.8 TODO

```
saltos.types.default = ()
```

TODO

8.3.9 TODO

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

TODO

8.3.10 TODO

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

TODO

8.3.11 TODO

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

TODO