Team Administration/Logistics Interface

User Manual

# Table of Contents

[Table of Contents 2](#_Toc164419003)

[Team Administration/Logistics Interface 3](#_Toc164419004)

[Author Information 4](#_Toc164419005)

[Version 5](#_Toc164419006)

[Installation 6](#_Toc164419007)

[Documentation 8](#_Toc164419008)

[Tests 9](#_Toc164419009)

[Contributors 10](#_Toc164419010)

[License 11](#_Toc164419011)

[Index, Header, & Footer 12](#_Toc164419012)

[Index 12](#_Toc164419013)

[Header 12](#_Toc164419014)

[Footer 12](#_Toc164419015)

[Modules 13](#_Toc164419016)

[Admin Accounts 13](#_Toc164419017)

[Admin Permissions 14](#_Toc164419018)

[Pages & Content 15](#_Toc164419019)

[News Entries 17](#_Toc164419020)

[Home Slider 19](#_Toc164419021)

[Master History Report 20](#_Toc164419022)

[Software Versions 21](#_Toc164419023)

[Personnel 22](#_Toc164419024)

[Mailing List 25](#_Toc164419025)

# Team Administration/Logistics Interface

The Team Administration/Logistics Interface, or TALI, is a Content Management System intended to provide unique functionality while remaining user friendly and compatible. User content is managed by a number of Modules that act as Graphical User Interfaces (GUI) in order to add, update, and remove content stored on a MySQL database handled by the user. Additionally, TALI contains multiple features tailored to organization and team management.

This User Manual is intended to provide detailed information as to the purpose, function, FAQs, and minor troubleshooting for each of TALI’s Modules in addition to installation, uninstallation, and maintenance tips. For further information or specifics in the code, the user should read the comments to the code within TALI’s various files. Any further questions or comments can be directed to the author.

# Author Information

Kent C. Grimes is the author of the Team Administration/Logistics Interface (TALI), an online Content Management System (CMS) created in 2014 intended to support websites in a user-friendly manner while still leaving the door open for future expansion via the user’s unique ideas.

The point of this CMS is to rely a little more on functionality and less on interface limitations, which is a step in a different direction from most CMS’s. Although this CMS may call for more understanding of website design than mainstream CMS’s, it will provide a more functional and more rewarding experience. The key is TALI’s ability to work seamlessly with other major installations in order to take advantage of its modules and functionality without the concern for compatibility and front-end appearance. This ultimately gives the user more freedom while developing their website.

# Version

TALI V1.0 is stable on PHP 5.6 and MySQL 5.7.3

# Installation

For now, TALI is only hosted on github at: <https://github.com/kcgrimes/tali>

At this time, there is no “installer” for TALI, and it is instead a series of folders and files that require placement.

1. Obtain necessary files (available at <https://github.com/kcgrimes/tali>)
   1. “tali” folder
   2. config.ini.tmpl
   3. tali\_init.php.tmpl
   4. tali\_db\_template.sql
2. Place the “tali” folder and the tali\_init.php.tmpl file somewhere in the root directory of the website (/public\_html, /www, /dev, etc.)
   1. It does not have to be in the top level of the root, it could be in any folder in that root. Ex: /public\_html, /public\_html/cms, etc.
3. If you choose to use it, place config.ini.tmpl in your web directory, preferably outside of the root directory for security purposes
   1. Also rename it to config.ini
4. For any page that may call on TALI or in a file that is always executed (defines.php, head.php, global.php, etc.), place the following line of PHP code towards the beginning of the page’s execution. Ideally this is just after where your <head> tags occur (where <title> and <meta> are defined, shortly after session\_start(), etc.)

//Initialize Team Administration/Logistics Interface (TALI)

require "tali\_init.php";

Note: If the tali\_init.php file is anywhere but the root directory (or whatever level <head> is in) you will need to adjust the path above accordingly, along with make adjustments as directed in tali\_init.php.

1. Using phpMyAdmin or similar MySQL Database GUI, import the TALI .sql file to the desired database so that the TALI tables will populate
2. Rename tali\_init.php.tmpl to tali\_init.php
3. Edit tali\_init.php to tailor TALI to your website’s configuration
   1. This file contains all of the various definitions, and small explanations for each one. Read carefully!
4. At this time, the user should not notice any changes to their website unless there are conflicting CSS definitions.
5. The user should however be able to access the “admin panel” TALI GUI at /tali/index.php (Ex. <https://www.domain.com/tali/index.php>) and begin utilizing the modules!
   1. The initial access login is temporary and should be changed to a permanent, unique login ASAP:
      1. Username: admin
      2. Password: password
   2. Although it is possible to set a hyperlink somewhere on the user’s website to access TALI, for security reasons it is suggested to instead setup a reasonably remembered redirect, such as /admin, pointing to /tali/index.php. This is done via the .htaccess file in the website’s root directory, or through a GUI such as cPanel, if made available by the website host.
   3. If a user is attempting to access TALI with a defined temporary password, they will be redirected to the Account page to set their own unique password.

# Documentation

A Word document is included with TALI. This User Manual is intended to provide detailed information as to the purpose, function, FAQs, and minor troubleshooting for each of TALI’s Modules in addition to installation, uninstallation, and maintenance tips. For further information or specifics in the code, the user should read the comments to the code within TALI’s various files. Any further questions or comments can be directed to the author.

# Tests

TALI is designed to exit upon critical failure and it will attempt to announce the problem in plain text. These types of failures are intended for development, and should never be encountered down the road if they were not encountered at launch, save for software updates. Upon setup or completion of modifications, it is recommended that the user, before launch, access at minimum the following pages:

* Home Page
* Some other front-end page that calls TALI
* Front-end page that uses a heavy TALI module (Roster, News, etc.)
* Back-end TALI Index
* Back-end page for any TALI module

# Contributors

Contributions are welcomed and encouraged. Please follow the below guidelines:

* Use the Pull Request feature
* Document any additional work
* Provide reasonable commit history comments
* Test all modifications locally and online

# License

MIT License

Copyright (c) 2014-2024 Kent C. Grimes. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

# Index, Header, & Footer

## Index

The TALI Index page is essentially a Landing Page for the GUI as it contains a series of Modules for the user to navigate. The visibility of Modules is determined by the user’s Admin Level and the Admin Permissions set for that Admin Level. Additionally, there are three non-module links:

* Log Out – Redirects user to login.php and logs them out of their session
* Access cPanel – Opens the website’s cPanel in a new tab (link is pre-defined)
* Return To Homepage – Redirects user to the index.php of the website (home page)

## Header

At the top of the Index and every other TALI page is the Header, consisting of multiple functions:

* Banner – Click the Banner to return to the TALI Index
* Go Back – Click to go up a level in the directory
* Account – Access the Account page for modification of one’s own account
* Log Out – Redirects user to login.php and logs them out of their session

## Footer

At the bottom of the Index and every other TALI page is the Footer, consisting of multiple functions:

* Terms and Conditions – Links to the Terms and Conditions verbiage
* Privacy Statement – Links to the Privacy Statement verbiage
* Declaration of copyright

# Modules

## Admin Accounts

This module allows you to Manage Admin Accounts, including Adding, Deleting, and Editing them. Admin Accounts consist of multiple components:

* ID – For reference purposes only and is not manually set or changed
* Level – Permission level, defined in Admin Permissions module
* Username – Username used to login (case-sensitive, can contain special characters)
* E-Mail – E-Mail, which can also be used to login
* Password – User-entered passwords are encryption (case-sensitive, can contain special characters)
* Associated Personnel ID – ID of the Personnel File associated to this Admin Account, if applicable

**Manage Admin Accounts**

To Update an Admin Account, use the ID to reference the Admin Account. Next, fill in the desired fields, or leave them blank to retain the current value. Click Update Account to submit the new value(s).

To Delete an Admin Account, use the ID to reference the Admin Account by putting it in the appropriate text input box. Then click the Delete Account button (Note: There is no way to recover deleted accounts from within TALI).

**Add Admin Account**

To Add an Admin Account, enter the desired Level, Username, E-Mail, and Personnel ID (if applicable). Duplicate Usernames and duplicate E-Mails are not accepted. A random password is automatically generated, so the new user will simply reset their password via E-Mail when they first log in.

## Admin Permissions

This module contains multiple parts involving Admin Permissions. All settings are individually submitted, so only set one at a time.

**Manage Access Level Amount**

Enter a desired number of admin levels. Upon submitting, the page will reload and the section below this one will adjust accordingly.

**Manage Admin Access Settings**

For each Admin Level (one at a time), Check the modules that should be visible to users assigned to that Admin Level. Click Submit for each level.

## Pages & Content

This module puts the “CMS” in TALI. Here is where most front-end content on the website will be managed. The inputted data will be stored on a database that is queried whenever the page is loaded. The code of the page contains mostly formatting, and includes a line that calls a function to query the database for the real content. The idea behind this module is to have a user-friendly means of modifying content that may change more regularly than an average user would like to be accessing and modifying files directly.

**Create a Page**

To Create a Page, simply type a title (used for reference only) of the page and click Create New Page. This merely creates the database entry for a page; more work needs to be done.

Before any pages will work, TALI must first be executed by having tali\_init.php located and executed as recommended in the Installation section of this manual. Once TALI has been executed (via require, include, etc.), the following PHP code can be executed in order to output the body content stored in the database for the page matching the value for page title:

/pagename.php

$pageTitle = "Page Title Here";

TALI\_fetchDBPage ($pageTitle);

As an example for generic pages with a header and footer, the following file formatting is the recommended foundation and can be edited to suit the user’s needs:

Create includes/global/head.php

//Uniform way of making each page - header, body, footer

function bodyContent ($bodyContentDir) {

//Body Start

echo " <body>";

//Header

require "includes/global/header.php";

//Main - $bodyContentDir is directory to unique file for page's Main

require $bodyContentDir;

//Footer

require "includes/global/footer.php";

//Body End

echo " </body>";

}

//Initialize Team Administration/Logistics Interface (TALI)

require "tali\_init.php";

Create /pagename.php (in the root directory)

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<?php

require "includes/global/head.php";

/\*Body - Directory points to Body content to be displayed with standardized header and footer\*/

$bodyContentDir = “includes/pagename/pagename-main.php”;

bodyContent ($bodyContentDir);

?>

</html>

Create /includes/pagename/pagename-main.php

<?php

//pageTitle - Match name within quotes to name of page in TALI

$pageTitle = "Pagename";

echo "

<div class=\"container\">

";

fetchDBPage ($pageTitle);

echo "

</div>

";

?>

As an additional example, if you are converting a normal website page to one that is editable in TALI, simply cut from the file the content that you want editable and paste it in the TALI Pages module, and replace it in the file with the $pageTitle definition and TALI\_fetchDBPage ($pageTitle) function as mentioned before.

**Edit or Delete a Page**

To Edit or Delete a page, simply select the desired page from the dropdown and click the desired action.

**Editing a Page**

On the Edit page the user is able to edit the Title (used only for reference purposes) of the page in addition to the content of the page. The large textbox is MarkItUp!, a 3rd party and open-source product that, in this case, allows for the use of BB Code to easily format text. This is similar to what is used in e-mails, forums, and blogs. The top bar has a few clickable buttons for quick access to the mark-up language, and below are some examples and additional supported commands:

|  |  |
| --- | --- |
| [b]Bold text[/b] | **Bold text** |
| [i]Italicize this[/i] | *Italicize this* |
| [u]Underline this[/u] | Underline text |
| [img][www.somesite.com/image.jpg[/img](http://www.somesite.com/image.jpg%5b/img)] | (image here) |
| [email][mail@site.com[/email](mailto:mail@site.com[/email)] | <mailto:mail@site.com> (the mailto: is not displayed) |
| [size=200]bigger text[/size] (value differs) | Bigger text |
| [color=red]Color[/color] (color names used) | Color |
| [\*]List item | * List item |
| [list][\*]List item[/list] | Properly formats list items. |

HTML markup can still be used in this text field.

Click Submit to apply changes. This will store the content in the database, to be referenced by a formatted file that calls on it.

## News Entries

This module allows the insertion of articles into the database. From the Admin Panel, articles can be added, modified, and deleted. From the front-end, there are two different styles of viewing.

**Adding an Article**

To add an entry, simply input an author, title, and body. The BB Code editor MarkitUp!, a 3rd party plugin, is used to allow BB code markup of the articles (more info on this in the Pages module). The date and time of entry will be automatically assigned upon submission. All entries have their own local history to track changes.

**Editing/Deleting an Article**

To edit an entry, navigate to the desired entry and click the Edit button. From this page you can modify the author, title, and body of the entry, or Delete it altogether. Below the entry is the local history report for that particular entry. The submission time will not change.

**Front-End Recent News**

One of the ways to view the database entries is through Recent News, which is formatted to display the desired number of articles in their own boxes. To do this, execute the TALI\_Module\_News\_Recent() function in a file anywhere after TALI has been initialized, and set the number of articles to be displayed.

Create /includes/index/index-main.php

<?php

TALI\_Module\_News\_Recent (3);

?>

In the above example, the function will execute in “index-main.php” and will display the three latest entries.

**Front-End News Page**

The second way to view the News module on the front end is as a News Page that includes multiple functions. Upon first access, it will display a designated latest number of articles, like with Recent News, in addition to a link to view an archive of articles. Any listed article can be clicked to be viewed. To do this, execute the TALI\_Module\_News () function in a file anywhere after TALI has been initialized, and set the number of articles to be displayed.

Create /includes/news/news-main.php

<?php

TALI\_Module\_News (5);

?>

In the above example, the function will execute in “news-main.php” and will display the five latest entries in addition to the user being able to access the archive of entries.

## Home Slider

This module allows the insertion of images and associated headers into the database to be viewed as a box with the headers below it, where upon highlighting the header the associated image will display.

**Adding an Image**

To add an image, choose the file and then assign it a title/header and weight (to determine display order) and click Upload. The image will then be uploaded to the FTP and visible on the front-end upon execution of the Home Slider function. Entries cannot be modified, only added and deleted.

**Deleting an Image**

To delete an image, just select it from the dropdown and click Delete and the image will be removed from the FTP along with the database entry being deleted.

**Front-End Home Slider**

The Home Slider module is visible on the front-end through a simple execution. To do this, execute the TALI\_Module\_HomeSlider () function in a file anywhere after TALI has been initialized, and set the number of images (with associated text) to be displayed.

Create /includes/index/index-main.php

<?php

TALI\_Module\_HomeSlider (3);

?>

In the above example, the function will execute in “index-main.php” and will display the three most weighted images with their associated text.

## Master History Report

This module contains a listing of all recorded events that have occurred in the TALI Admin Panel. Within the TALI code there is a function that is called after most actions that affect the database (inserts, updates, and deletes) are processed. This function stores numerous data points as a single entry in the Master History Report module for accountability purposes. Items recorded include:

* Time – Time of record
* User ID – User ID of Admin Account that executed action
* Module – Name of involved module
* Item ID – Unique ID of involved entry (each module is independent)
* Event – Plain language version of the record, usually indicating the plain-text name of the affected entity

## Software Versions

This module is strictly informative and simply displays the detected versions of software running on the website. This is useful for debugging compatibility issues.

## Personnel

This module allows management of Personnel Files including the use of designations, ranks, statuses, roles, awards, drill reports, various analyses, and more, in addition to the front-end display of some of this data. Although there are many components built into this module, not all of them have to be used.

**Configuration**

Before the user can set up Personnel Files, the module must be configured to the desired setting. This is made simple on the Configuration page. Only one item can be configured at a time. All entities can be added, modified, and deleted as desired.

Ranks – Include name, abbreviation, an uploaded image, and weight of the rank

Statuses – Include name and weight of the status

Roles – Include name and weight of the status

Designations – Include name, leader, weight, and superior of the designation (team, organization, etc.). These can be activated and deactivated as opposed to deleted, for the sake of preserving data.

**Roster**

The Roster is a view of all Active and Past Personnel, as desired. It will display their rank, name, and status, along with providing the options to edit the individual’s awards, service record, uniform, and profile. Personnel can be added by clicking the Add button, filling out the form, and clicking Submit.

When editing Awards, awards can be added along with a date and comment. This entry can be modified or deleted.

When editing Service Records, service record entries can be added along with a date of record. This entry can be modified or deleted.

Each personnel file has the opportunity to have a Display Uniform and Modifiable Uniform file associated to it. When editing Uniforms, you can either download the display (.png, .jpg, etc.) or modifiable (.psd, etc.) image, or upload new files for each. If there is no associated image file or the associated file could not be found, the respective default file, as defined in tali\_init.php, will be downloaded instead, assuming it can be located (Note: the mentioned “default files” are not present by default, and must be added by the user). The actual uniform files are located in /tali/talisupplement/personnel/uniforms and /uniforms-modifiable. The system is set up so that the user can organize their uniform files however they please as long as they are in their respective folder. When the system is searching for the appropriate file to download, it searches from the top most folder (in this directory) and from A to Z and selects the first one, allowing for the presence of older files. For instance, if a “Current” folder contains JDoe.png and an “Old” folder also contains JDoe.png, the “Current” one will be downloaded when the link is clicked, and the “Current” one will also be replaced upon upload of another JDoe.png file. Upon uploading a file by a different name, such as JADoe.png, the personnel file will re-associate to that file name and display it correctly, while leaving the file with the old name in place. This is also true when uploading uniforms for the first time to a personnel file. The Modifiable and Display association systems are separate, so the files could have different names.

When editing Profiles, the minimum amount of data must be retained. Files can be deleted, but this results in loss of data. It is preferred that files for members that are no longer present be Discharged, which is essentially a means of archiving the file for use in statistics, viewing, or future use.

**Awards**

Award Classes are used to categorize awards, and can be added by including a name and weight. These entries can then be modified or deleted.

Awards can be assigned to Personnel Files. Awards can be added by including a name, an image file to upload, an award class, a description of the award (such as a condition or standard), and a weight. These entries can then be modified or deleted.

**Drill Reports**

Drill Reports are used to document event attendance and provide comments. This data can be referenced later on for viewing purposes or statistics. To Create a Drill Report, enter the date of the drill, select a designation, check/uncheck personnel as appropriate for attendance tracking, and provide any comments in the MarkitUp! BB code editor (more on this in Pages module). Drill Reports can later be viewed by designation and modified if needed.

**Metrics**

On the Metrics page, numerous analyses can be conducted based on an individual or a designation. Additionally, various Personnel module statistics are also calculated.

**Front-End Personnel Module**

The front-end of the Personnel module includes tabular display of the roster (past and present), awards, and ranks. The roster itself is sorted by designation and rank. Individuals can be selected to view their profile which includes their uniform image, various statistics and profile information, and awards. The Personnel module is visible on the front-end through a simple execution. To do this, execute the TALI\_Module\_Roster () function in a file anywhere after TALI has been initialized.

Create /includes/roster/roster-main.php

<?php

TALI\_Module\_Roster ();

?>

In the above example, the function will execute in “roster-main.php” and will display the front-end of the Personnel module.

## Mailing List

This module allows the customization of mailing lists in addition to sending e-mails to individuals, custom mailing lists, or dynamic mailing lists based on member status.

Custom Mailing Lists are added by providing a name and list of e-mail addresses.

E-mails are sent by selecting the desired groups or entering individual e-mail addresses, along with entering a subject and body. The message body is edited in the MarkitUp! BB code editor (more on this in the Pages module) which also supports HTML.