Calling from a Script

Reportico has a nice interface that allows users to select a report to run, then fill out the criteria and run it. However, in some cases, we want more control. This is done programmatically with a a simple call to the reportico URL.

For more control, you can instantiate a copy of the reportico class or namespace. I will not cover namespace here since that is pretty well documented. Instead, I will cover setting up a class. The documentation is not available on this, but you can read the code in run.php, which is well documented.

The following PHP script will accept a report name (file name, sans the .xml) and show the report screen. I've left the comments from the original code, and added a few myself.

reports_run.php

reports_run.php

```
<?php

   $page_title = 'Reports';

   // get the report to execute, minus the .xml
   $thisReport = $_REQUEST['reportname'];

   // taken from start.php
   require_once( "/path/to/reportico/vendor/autoload.php" );
   // set error reporting level
   error_reporting(E_ALL);

   // Turn on logging ot browser console
   //Reportico\Engine\ReporticoLog::activeDebugMode();

   // Set the timezone according to system defaults
   date_default_timezone_set(@date_default_timezone_get());

   // Reserver 100Mb for running
   ini_set("memory_limit","100M");

   // Allow a good time for long reports to run. Set to 0 to allow
unlimited time
   ini_set("max_execution_time","90");

   // Only turn on output buffering if necessary, normally leave this
commented
   //ob_start();
```

```
   // Instantiate Reportico
   $reportico = new Reportico\Engine\Reportico();

   // Force reset of session handling, so you always start from scratch
   $reportico->clear_reportico_session = true;
   // Start user in specific project
   $reportico->initial_project = 'reporticoProjectName';
   // If starting user in specific project then project passweord is
required if one exists
   // and you dont want user to have to type it in
   $reportico->initial_project_password = 'reportico project password';
   // Specify a report to start user in specify the xml report file in
the specified project folder
   $reportico->initial_report = "$thisReport.xml";
   // Specify whether user is started in administration page, project
menu, report criteria entry,
   // report output or report design mode, use respectively ( "ADMIN",
"MENU", "PREPARE", "EXECUTE", "MAINTAIN")
   // default is "ADMIN"
   $reportico->initial_execute_mode = "PREPARE";
   // Specify access mode to limit what user can do, one of :-
   // FULL - the default, allows user to log in under admin/design mode
and design reports
   // ALLPROJECTS - allows entry to admin page to select project  but
no ability to logon in admin/designer mode
   // ONEPROJECT - allows entry to a single project and no access to
the admin page
   // ONEREPORT - limits user to single report, crtieria entry and
report execution ( requires initial project/report )
   // REPORTOUTPUT - executes a report and allows to "Return" button to
crtieria entry ( requires initial project/report )
   //$reportico->access_mode = "<MODE>";
   $reportico->access_mode = "ONEREPORT";
   // For passing external user parameters, can be referenced in SQL
with {USER_PARAM,parameter_name}
   // and can be referenced in custom SQL with $this->user_parameters
   $reportico->user_parameters["userid"] = $_SESSION['user-id'];
   // Show or hide various report elements
   $reportico->output_template_parameters["show_hide_navigation_menu"]
= "hide";
   $reportico->output_template_parameters["show_hide_dropdown_menu"] =
"hide";
//$reportico->output_template_parameters["show_hide_report_output_title
"] = "show";
//$reportico->output_template_parameters["show_hide_prepare_section_box
es"] = "hide";
//$reportico->output_template_parameters["show_hide_prepare_pdf_button"
] = "show";
   // the expand HTML button in second window
$reportico->output_template_parameters["show_hide_prepare_html_button"]
= "hide";
```

```
   // the HTML icon
$reportico->output_template_parameters["show_hide_prepare_print_html_button"] = "hide";
   if ( $thisReport == 'merchandise_received') {
$reportico->output_template_parameters["show_hide_prepare_csv_button"] = "hide";
   }
//$reportico->output_template_parameters["show_hide_prepare_page_style"] = "show";
   // the "Go" button
$reportico->output_template_parameters["show_hide_prepare_go_buttons"] = "hide";
//$reportico->output_template_parameters["show_hide_prepare_reset_buttons"] = "hide";

   $reportico->pdfDownloadMethod = 'inline';

   // Set a theme
   // ======================
   // Use the specified folder under the themes folder to identify which templates, stylesheets and js to use for the instance
   //$reportico->theme = "default";
   $reportico->theme = "bootstrap4";
   //$reportico->url_path_to_templates = "themes";

   // Set this to true to allow changes to edits to theme templates to be reflected immediately, otherwise
   // you will need to clear out the themes/cache folder to register any changes
   $reportico->disableThemeCaching = true;

   // Setup SESSION
Reportico\Engine\ReporticoSession::setUpReporticoSession($reportico->session_namespace);

   // Run the report
   $reportico->execute();
?>
```

Note that this script assigns the SESSION value of user-id to the external use parameter 'userid'. This can be referenced within the report or criteria as {USERPARAM, 'userid'}

From:
https://wiki.linuxservertech.com/ - **Unix Server Tech Knowledge Base**

Permanent link:
https://wiki.linuxservertech.com/software/reportico/calling

Last update: **2023/07/04 00:09**