Templates

Twig is fully documented with a dedicated online book and full API documentation. For more information see:
Channey terem
SupportPro ships with a default set of templates:
 resources/templates/frontend/default
 resources/templates/operator/default
Unsupported - Periods within directory names Please ensure your template directory name ("my-new-template" in the example below) does not contain any periods.

For small template changes we highly recommend to investigate whether it's possible to use our plugin eco system. This appropriate is more beneficial than the following 2 methods because it means you no longer have to maintain the changes between SupportPro versions.

For example, below we're going to change the default selected Priority on the Submit Ticket page that's visible to users:

Create a plugin, see for more information.
 Define a for "frontend.footer" with the following JavaScript:
 script type="text/javascript"> var element =

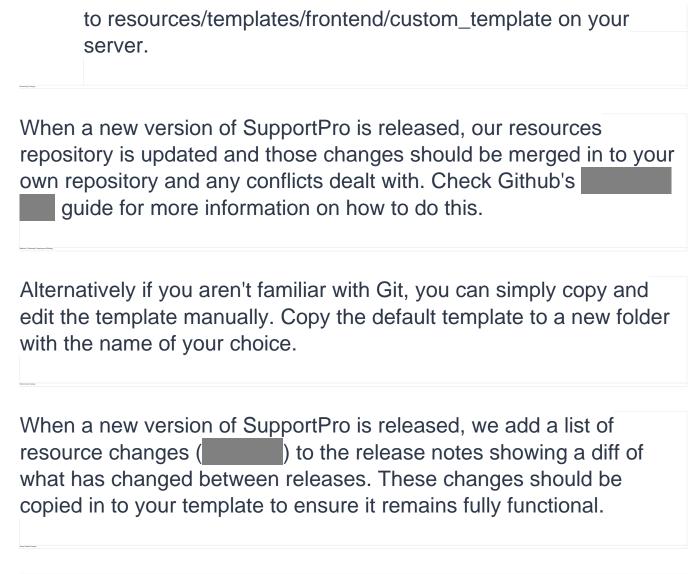
document.getElementById('priority'); if (element !== null) {

element.value = 1; }</script>

3. Above change 1 to the ID of the priority that you want to select

	are several other use cases that our possible so we encourage read through the plugin documentation.
Githuk have y tracke	re familiar with Git, we've made our templates available on and), allowing you to your own fork and build your templates in a way that can be d and automatically updated when new versions of SupportPro leased.
	Visit the relevant repository at Github (linked above), login and click the 'Fork' button near the top right. This will set up a fork of the repository in your account. Next clone the repository to your computer, you can find the clone link by clicking the 'Clone or download' button. For example for the frontend-template:
3.	You can now begin to make your changes, and can also commit them to your fork.
4.	When ready to deploy, login to your server via FTP or control panel.

- 5. Browse to your SupportPro installation directory, and create a new template directory in resources/templates/frontend for exa mple: resources/templates/frontend/custom_template where cu stom_template is the name of the new directory.
- 6. Upload all files from the repository



When working on your template, we recommend to keep the asset files (files under the resources/assets/ folder) the same, and instead create your own CSS or JS files that are loaded separately, making it easier to work with changes to assets files between releases.

- To preview the new template, use the 'template' parameter in URLs, such as https://www.yourdomain.com/support/admin/?te mplate=my-new-template
- 2. To activate the new template:
 - Go to Settings -> Core -> Brands and select the brand you wish to set it for.
 - 2. Click the "Website" tab.

	3. Change the Frontend or Operator Template to your new template.
	4. Click Save.
3.	If you're updating the operator template, you may need to also update your if you currently have it set to a specific template (not ' System Default'). The same can be done for other operators through the section.
Milyan	
	made some of our helper functions available for use in ites, below are a few that you might find useful in your template es.
humping milital.	
the HT	need to display any user submitted content, it is best to purify ML to remove the risk of XSS. This can be done with rifier.clean method.
{ { Pur	ifier.clean(html_content) }}
Turi a un	vill outomotically accompany LITML within a Twin ton. To otom
this an	will automatically escape any HTML within a Twig tag. To stop and display the code as it is, use the raw filter. Be careful to only on safe HTML that doesn't contain user submitted content.
{ { htm	nl_contentlraw }}
,	
	es a versioned URL for assets, so users download fresh assets tware updates.

```
link href="{{ asset rev('
resources/assets/operator/css/main.css') }}" rel="stylesheet" type="text/css" />
Checks if the user is logged in, works for both frontend and operator
panel.
{ % if auth_check() % }
Fetches the authenticated user (or operator). Returns null if the user is
not logged in.
   auth_user().formatted_name }}
Returns a formatted date for a timestamp based on the settings in the
help desk.
  formatDate(item.created_at) }}
Returns a relative date for a timestamp, such as '2 days ago'.
{ timeago(ticket.created_at) }}
```

By default, we only allow certain PHP functions to be used in the templates. You can add additional functions to the configuration if you'd like to use them in the Twig code, below is an example of adding the count() and substr() functions.

Important

When upgrading to a new version of SupportPro please ensure that any new configuration changes are updated in your production copy. If you do not migrate changes, array elements may be overwritten and the policy will be incorrect.

Create a new file twigbridge.php in the /config/production folder instead of updating the twigbridge.php found in the main /config folder, this will mean your configuration is not lost when you update your system.

```
<?phpreturn [ 'extensions' => [ 'functions' => [
    'count', 'substr' ], ],];
```

Online URL: https://docs.supportpro.vn/article/templates-251.html