

Redirects

TABLE OF CONTENTS

- [Redirect to an internal url](#)
- [Redirect in a form](#)
- [Redirect off-site \(to a third-party URL\)](#)
- [Redirect to an existing route with an anchor \(or fragment\)](#)
- [Redirect to a complex route](#)
- [Redirect in a controller](#)
- [Redirect user after login](#)
- [Redirect to the 403 or 404 page](#)
- [Redirect to a new page after node operation](#)
- [Redirect dynamically to wherever you came from](#)

views 72

Redirect to an internal url

In a controller when we return a `RedirectResponse` instead of a render array, Symfony redirects to the URL specified in the `RedirectResponse`.

For example:

```
use \Symfony\Component\HttpFoundation\RedirectResponse;

return new RedirectResponse('node/1');

// Or to redirect to the front page of the site.

$url = Url::fromRoute('<front>');
return new RedirectResponse($url->toString());
```

Redirect in a form

In a form, you can redirect to a route by its name:

```
$form_state->setRedirect('entity.bike_part.canonical', ['bike_part' => $entity->id()]);
```

Note. You can also redirect to a specific id (anchor) on the page by adding the fragment parameter

```
$form_state->setRedirect('tea_teks_admin.timeline_detail',
  ['node' => $parent_nid,
  ['fragment' => 'milestone-' . $nid.]
];
```

Or here in `submitForm()`:

```
$url = Url::fromRoute('user_account.user_register', [], ['query' => ['destination' => $shippingUrl]]);
$form_state->setRedirectUrl($url);
```

Redirect off-site (to a third-party URL)

From <https://drupal.stackexchange.com/questions/136641/how-do-i-redirect-to-an-external-url>

Redirect to a third party website with Redirect permanent status. The URL must have absolute path like <http://www.google.com>

```
return new RedirectResponse('https://google.com');
```

e.g.

```
Use Drupal\Core\Routing\TrustedRedirectResponse;

$absolute_url = 'https://google.com';
$response = new RedirectResponse($absolute_url, 301);
//OR
$response = new \Drupal\Core\Routing\TrustedRedirectResponse($absolute_url, 301);
$response->send();

exit(0);
```

Redirect to an existing route with an anchor (or fragment)

You can specify the route using the route name from your module.routing.yml file.

Note. You can also redirect to a specific id (anchor) on the page by adding the fragment parameter

```
$form_state->setRedirect('tra_teks_admin.timeline_detail',
  ['node'=>$parent_nid],
  ['fragment' => 'milestone-' . $nid,]
);
```

Redirect to a complex route

Here the route requires four arguments so they all need to be passed in.

```
$form_state->setRedirectUrl(Url::fromRoute('tra_teks_srp.confidential_voting', [
  'citadel_nid' => $citadel_nid,
  'performance_nid' => $performance_nid,
  'action' => 'vote',
  'type' => $type,
]));
```

Redirect in a controller

When you need to send the user to a different page on your site based on some logic, you might use code something like this. Lots of processing happens and if the conditions are met, instead of returning a render array, we return a RedirectResponse and the browser will load that page.

```

protected function reloadOperations(): ?RedirectResponse {
    $program_nid = $this->programNode->id();
    if (substr($this->action, 0, 6) === 'reload') {
        $new_type = substr($action, 7);
        $this->votingProcessor->buildVotingPath($this->programNode, $new_type);
        $valid_voting_path = $this->votingProcessor->loadYourVotingPathFromTempStore($program_nid);
        $empty_voting_path = $this->votingProcessor->isEmptyVotingPath();
        if ($valid_voting_path && $empty_voting_path) {
            // Redirect to kss overview.

            $message = "No eligible items found. Redirected to KSS overview";
            \Drupal::messenger()->addMessage($message);

            $url = Url::fromRoute('tea_teks_srp.kss_overview', [
                'program' => $program_nid,
                'type' => 'all',
                'userid' => $this->persona->getUserId(),
            ]);
            return new RedirectResponse($url->toString());
        }
        if ($valid_voting_path && !$empty_voting_path) {
            //redirect to the first item in the path.

            $item = $this->votingProcessor->getFirstVotingPathItem();
            $vote_type = 'narrative';
            if (empty($item['narrative'])) {
                $vote_type = 'activity';
            }
            $url = Url::fromRoute('tea_teks_srp.correlation_voting', [
                'program' => $program_nid,
                'expectation' => $item['expectation_nid'],
                'correlation' => $item['correlation_nid'],
                'action' => "vote-$new_type",
                'type' => $vote_type,
            ]);
            return new RedirectResponse($url->toString());
        }
    }
}

```

Redirect user after login

From: <https://www.drupal.org/forum/support/module-development-and-code-questions/2013-08-18/how-to-redirect-user-after-login-in> This example checks both the route and user role to conditionally redirect.

```

use Drupal\Core\Session\AccountInterface;
use Drupal\Core\Url;

/**
 * Implements hook_user_login().
 */
function greenacorn_user_login(AccountInterface $account) {
    $roles = $account->getRoles();
    $route_name = \Drupal::routeMatch()->getRouteName();
    if ($route_name != 'user.reset.login' && in_array('client', $roles))
    {
        $destination = Url::fromUserInput('/my-issues')->toString();
        \Drupal::service('request_stack')->getCurrentRequest()->query->set('destination', $destination);
    }
}

```

Redirect to the 403 or 404 page

```
// Redirect to the 403 page.
throw new \Symfony\Component\HttpKernel\Exception\AccessDeniedHttpException();

// Redirect to the 404 page.
throw new \Symfony\Component\HttpKernel\Exception\NotFoundHttpException();
```

Redirect to a new page after node operation

This was implemented in a module in a `hook_form_alter` call. Here we tell the form to call the special handler: `cn_submit_handler()` function in the form.

```
function obg_mods_form_alter(array &$form, FormStateInterface $form_state, $form_id) {
    $accountProxy = \Drupal::currentUser();
    $account = $accountProxy->getAccount();

    // Add special validation for anonymous users only.
    if (($accountProxy->isAnonymous() && ($form_id == 'node_catastrophe_notice_form')) {
        $form['#validate'][] = 'cn_form_validate';
    }

    // Submit handler to redirect to /thanks-your-submission.
    foreach (array_keys($form['actions']) as $action) {
        if ($action != 'preview' && isset($form['actions'][$action]['#type']) && $form['actions'][$action]['#type'] === 'submit') {
            $form['actions'][$action]['#submit'][] = 'cn_submit_handler';
        }
    }
}
```

And here is the submit handler where the work gets done and you are redirected to `/thanks-your-submission`.

```
/**
 * Submit handler to redirect user to thank-you page.
 *
 * @param $form
 * @param \Drupal\Core\Form\FormStateInterface $form_state
 */
function cn_submit_handler($form, FormStateInterface $form_state) {
    $url = Url::fromUri('internal:/thanks-your-submission');
    $form_state->setRedirectUrl($url);
}
```

More at <https://drupal.stackexchange.com/questions/163626/how-to-perform-a-redirect-to-custom-page-after-node-save-or-delete>

Redirect dynamically to wherever you came from

In this case, in a form, we grab the referrer url in the `buildForm` method using the following code:

```

public function buildForm(array $form, FormStateInterface $form_state, int $program_nid = 0, int $feedback_error_nid = 0, int $citation_nid = 0){
    $form['#theme'] = 'tea_teks_srp__vote_error';

    // Get the referer.
    $request = \Drupal::request();
    $referer = $request->headers->get('referer');
    // $base_url = Request::createFromGlobals()->getSchemeAndHttpHost();
    $base_url = \Drupal::request()->getSchemeAndHttpHost();
    $alias = "";
    if (!is_null($referer)) {
        $alias = substr($referer, strlen($base_url));
    }
    $form_state->set('referrer_alias', $alias);
    ...
}

```

Then in the submitForm() method, once we've completed the work we needed to do, we can redirect back to where we came from using the code below. We can optionally add a fragment which refers to an ID on the page.

```

$referrer_alias = $form_state->get('referrer_alias');

// Add the fragment so they drop back on the item they came from.
$url = Url::fromUri('internal:' . $referrer_alias, ['fragment' => "item_$feedback_error_nid"]);
$form_state->setRedirectUrl($url);

```

[Back to top](#)

Drupal at your fingertips by [Selwyn Politi](#) is licensed under [CC BY 4.0](#) 

Page last modified: Jun 22 2023.

[Edit this page on GitHub](#)

This site uses [Just the Docs](#), a documentation theme for Jekyll.