Six steps to create an experiment in Optimizely Web

Skip Ahead

- Step 1: URL Targeting
- Step 2: Create a variation
- Step 3: Add audiences (optional)
- Step 4: Add metrics
- Step 5: Set traffic allocation
- Step 6: QA and publish

RELEVANT PRODUCTS:

- Optimizely Web Experimentation
- Optimizely Performance Edge

THIS ARTICLE WILL HELP YOU:

- Set up an A/B test with Optimizely Web Experimentation or Optimizely Performance Edge

Optimizely makes it easy to A/B test your site. Use the Editor to design variations, define where the experiment runs and who sees it, set events to measure success, and QA. When your experiment looks and works the way you'd like, publish it live to the world!

Follow the six steps in this article to create, configure, and launch a new experiment in Optimizely. If you want to make sure you've covered all your bases, this checklist helps you set up Optimizely.

Optimizely Performance Edge is a lightweight experimentation product that delivers significantly faster performance than other versions of Optimizely. It does this by relying on a streamlined "microsnippet" which limits the range of available features.

If you're using Optimizely Performance Edge, some of the features mentioned in this article may not be available to you.
Each section of this article contains links to other, more detailed articles on that section’s topic. Click through to those articles for a clear delineation of which features are available for Edge experiments, and which are not.

Want to dig a little deeper? Check out our Optimizely Academy course on how to build an experiment in Optimizely.

Begin by creating a new experiment. You'll add URL targeting, variations, audiences, and metrics to your experiment in later steps.

This experiment design template can help you document and share the details of your experiment.

1. Navigate to the Experiments dashboard.

2. Click Create New.

3. Select A/B Test from the dropdown.

4. Type a name and description for your new experiment.

---

**Step 1: URL Targeting**

The first step is to tell Optimizely where your experiment will run. You can target using a single URL or you can use pages for this instead.

In general, if you want to add a simple URL to run an experiment just once, target the single URL. If, on the other hand, you plan to run experiments on this set of URLs regularly, you'll want to use pages.

In Web experiments, you can use a combination of page triggers and conditions to tell Optimizely when to activate a page. These options are useful for single page applications (SPAs). In Performance Edge experiments, pages always activate immediately.

Be sure to see our Knowledge Base article on how to set up a new page.
Before you can run an experiment, you'll need to implement the snippet and create at least one page and metric in Optimizely Web. If your account uses custom snippets, you might see more than one snippet listed. In this case, choose the snippet you want to use for your experiment.

**Step 2: Create a variation**

From the *Experiments* dashboard, click the experiment name to open the experiment. To make changes to your variations. Select the variation you want to edit.

In the *Visual Editor*, you can make any changes to the layout and appearance of your site, including:

- Change the layout, including the element's visibility or position; or even rearrange the element on the page
• Modify or replace typography, images, background styles or borders

• Add inline CSS

• Fine-tune jQuery selectors

• Change the timing of the change from synchronous to asynchronous

---

Step 3: Add audiences (optional)

Audiences allow you to decide who will see your experiment.

The default audience is all visitors to your site (Everyone). Click (+) to add any existing audiences to your experiment. Or, click Create New Audience to build a new audience.

You can combine multiple audiences in your experiment using AND and OR conditions. To learn more, check out our article on setting up Audiences in Optimizely.
Step 4: Add metrics

Add the [metrics](#) that measure success for your experiment. If the changes in your variation are successful, what will you measure to show improvement?

Create a metric from an existing event or [create a new event](#) for the experiment. To add an existing event, click (+).

The first metric you add is the [primary metric](#) for the experiment. This metric determines whether your experiment "wins" or "loses," and should track an event that's directly affected by the changes you make in your experiment.

Optimizely's [Stats Engine](#) makes sure that the primary metric reaches [statistical significance](#) as quickly as possible. You can also add [secondary metrics](#) to measure the downstream effects of your experiment.

To complete the experiment setup, you must add at least one metric. Don't worry—you can always change, remove, or add new metrics to your experiment later. Click Save when you are finished adding metrics to your experiment.

Step 5: Set traffic allocation

Set traffic allocation to specify how [traffic will be split](#) between your variations. Optimizely randomly allocates traffic into different variations, including the original.

If you like, you can change the [percentage of traffic](#) that goes into each variation. Or, you can leave it as is. You can also change the [total traffic](#) that goes into the experiment, as a whole.
Step 6: QA and publish

After you’ve set up all components of your experiment, preview it to make sure it looks and works the way you intend.

Use the Preview button to view visual changes for your variation.

Not seeing your changes in Preview mode? Make sure your snippet is implemented on this page and configured to include Optimizely.

Check out our KB article for a comprehensive checklist of what you’ll need to consider before fully QAing your experiment, or go deeper and learn about advanced QA and troubleshooting.

When everything looks and works the way you’d like, click Start Experiment to set the experiment live for visitors.
And with that, you've created and launched an experiment in Optimizely Web Experimentation. Well done!