

# React Router x Framer Motion - Page Transitions in React Router with Framer Motion - Animate transitions easily in React

*Written by Seth Corker on Benevolent Bytes*

A benefit of doing client-side routing is, as developers, we have more control over how we want to handle the transition from one page to the next. With React Router, we gain fine-grained control at a per-component level. Let's take advantage of this and animate the transition between different routes with the help of Framer Motion.

If you'd like to do page transitions in Next.js, I've got you covered too. You can use similar techniques to ensure pages animate smoothly between each other and also make use of advanced techniques like shared layout to achieve more native-like experiences.

## What we're creating

*Look at the Demo app to see what the animation looks like in action.*

The final animation, slide-in, scale up and fade-in from the left and slide-out, scale-up and fade-out to the right

The tutorial focuses on a step-by-step approach to animating a React app which is using React Router for routing. We'll start with a simple fade-in/fade-out animation by animating the opacity of the component. We'll then look at how we can animate the position to create a slide animation and finally we can leverage multiple variants (animation states) to have a different animation for mounting and unmounting the components when the route changes.

## The video tutorial

Video tutorial explaining how to animate page transitions with Framer Motion

**Animating mount and unmount** `AnimatePresence` is the key component for easily animating the route transitions. It's job is to determine whether a component is mounting or unmounting and play an animation on our `<motion />` components. The routes we want to animate are within the `<Switch/>` component will mount and unmount but we also need to tell `AnimatePresence` if we want to animate at all, this is where the key prop comes in. We don't want to trigger an animation when we navigate to the same route, using the `location.pathname` as the key results in an animation which only triggers when we change to a new route.

```
<AnimatePresence>
  <Switch location={location} key={location.pathname}>
    <Route path="/about" component={About} />
    <Route path="/settings" component={Settings} />
    <Route path="/" component={Home} />
  </Switch>
</AnimatePresence>
```

**Variants** Using variants in Framer Motion is an easy way to define the different animation states we want to transition between. Take the following example:

```
const pageVariants = {
  initial: {
    opacity: 0,
  },
  in: {
    opacity: 1,
  },
  out: {
    opacity: 0,
  },
}
```

We can now reuse the animations we defined in `pageVariants`, initial will be state which the component starts on. When the component mounts, it will animate from the `initial` state to the `in` state. `AnimatePresence` adds another state to our component which it expects to also be defined, `exit`. When the component unmounts, it will animate between the `in` and `out` defined in our `animate` and `exit` props respectively.

```
<motion.div
  initial="initial"
  animate="in"
  exit="out"
  variants={pageVariants}
>
```

We leverage this behaviour in React Router. As the route changes the `Switch` component determines which component to show. Each component is wrapped in a `<motion.div>` with animation states defined and thus the transition animations are played. It's as easy as that.

### Where to go from here

Framer Motion is easy to use and page transitions aren't too difficult to get working, I recommend experimenting with different transitions and getting comfortable with animations and variants. For more information, the Framer Motion documentation is a great place to start - it has simple examples to learn from. I've also written some more tutorials just like this one on how you can get started with Framer Motion.

### Resources

- Source code for the tutorial is available on [GitHub](#)
- [Framer Motion Documentation](#)
- My previous tutorial video - [Scroll-up Reveal animation in Framer Motion](#)

Updates: This article was updated on 04/06/2021 with the following changes:

Linked the video tutorial