

# More fun 🤩

- No need to worry about stale data
- Easily add small things that really boost UI/UX
- Control of the data flow
- (Better) Types for API responses

```
useEffect(() => {  
  const interval = setInterval(() => {  
    // only poll if we have submitted steps,  
    // otherwise we need to take action first on the partner side  
    if (revalidator.state === "idle" && hasSubmittedSteps) {  
      revalidator.revalidate();  
    }  
  }, POLLING_INTERVAL);  
  return () => clearInterval(interval);  
}, [revalidator, hasSubmittedSteps]);
```

# Less code 🧹

- No Redux 🐱
- State handling **is** network request handling
- Useful hooks from react-router
- `<form>`s! HTML helping us with what it does best 😍

```
<Form method="post">
  <input type="hidden" value={JSON.stringify(step)} />
  <input type="hidden" value={LQV_ACTION_INTENT.SUBMIT} />
  <Button
    variant="filled-accent"
    className="mx-auto inline-block w-full max-w-sm"
    disabled={!stepIsCompleted || isSubmitting}
    type="submit"
  >
    {T.translate(isSubmitting ? "lqv.submitting" : "lqv.submit")}
  </Button>
</Form>
```