Lab 04
Formulare mit React-Hook-Form

1.
```
npm install react-hook-form
```

In Form.jsx:

```
import { useForm } from "react-hook-form";

export default function Form() {
  const { register, handleSubmit, watch,
formState: { errors } } = useForm();
  const onSubmit = data =>
console.log(data);

  console.log(watch("example")); // watch
input value by passing the name of it

  return (
    /* "handleSubmit" will validate your
inputs before invoking "onSubmit" */
    <form
onSubmit={handleSubmit(onSubmit)}>
      {/* register your input into the
hook by invoking the "register" function
*/}
      <input defaultValue="test"
{...register("example")} />

      {/* include validation with
required or other standard HTML
validation rules */}
```

```
      <input
{...register("exampleRequired", {
required: true })} />
      {/* errors will return when field
validation fails  */}
      {errors.exampleRequired &&
<span>This field is required</span>}

      <input type="submit" />
    </form>
  );
}
```

```
import { useForm } from "react-hook-
form";

export const Form = () => {
    const {
      register,
      handleSubmit,
      formState: { errors },
    } = useForm();
    const onSubmit = (data) => {
      console.log(data);
  };
    return (
      <form
onSubmit={handleSubmit(onSubmit)}>
        <input type="text"
{...register("textInput", { required:
true })} />
        {errors.textInput && <div>Die
Eingabe wird benötigt.</div>}
```

```
        <button
type="submit">Absenden</button>
  </form> );
  };
```

2. In Fluent-UI Form einbauen:

```
import { useForm, Controller } from "react-
hook-form";
import { TextField, Checkbox } from "@material-
ui/core";

function App() {
  const { handleSubmit, control, reset } =
useForm({
    defaultValues: {
      checkbox: false,
    }
  });
  const onSubmit = data => console.log(data);

  return (
    <form onSubmit={handleSubmit(onSubmit)}>
      <Controller
        name="checkbox"
        control={control}
        rules={{ required: true }}
        render={({ field }) => <Checkbox
{...field} />}
      />
      <input type="submit" />
    </form>
  );
}
```

```
import { useForm, Controller } from "react-
hook-form";
import { Checkbox, TextField } from
'@fluentui/react';
import { PrimaryButton } from
'@fluentui/react/lib/Button';

export function FormFluent() {
  const { handleSubmit, control, reset } =
useForm({
    defaultValues: {
      checkbox: false,
    }
  });
  const onSubmit = data => console.log(data);

  return (
    <form onSubmit={handleSubmit(onSubmit)}>
      <Controller
        name="checkbox"
        control={control}
        rules={{ required: true }}
        render={({ field }) => <TextField
{...field} />}
      />
      <PrimaryButton type="submit">Click
Me</PrimaryButton>
    </form>
  );
}
```

3. Checkbox, etc. andere UI-Komponenten ausprobieren (Doku befolgen!)