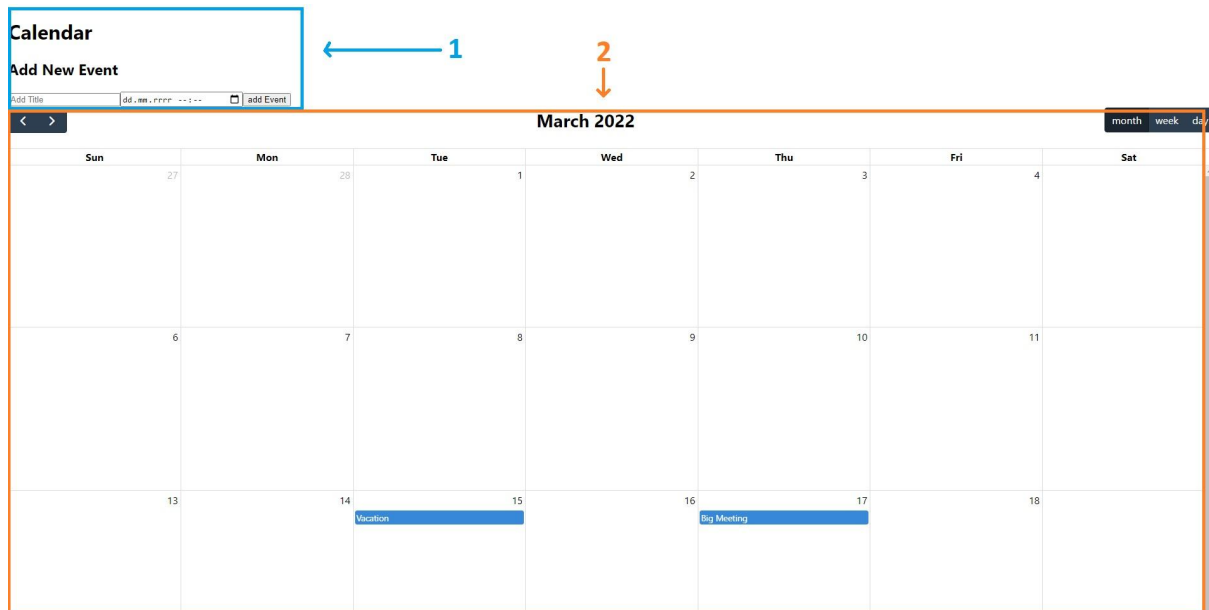


Krok 1: Podział interfejsu



1. Formularz, który pobiera tytuł datę i czas od użytkownika
2. Gotowy kalendarz z biblioteki

Krok 2: Wersja statyczna

```
import React from "react";

import FullCalendar from "@fullcalendar/react";
import dayGridPlugin from "@fullcalendar/daygrid";
import timeGridPlugin from "@fullcalendar/timegrid";

function App() {
  return (
    <div>
      <h1>Calendar</h1>
      <h2>Add New Event</h2>
      <div>
        <form>
          <input
            type="text"
            placeholder="Add Title"
          />
        </form>
      </div>
    </div>
  );
}
```

```

        />
        <input
            type="datetime-local"
            placeholder="Start Date"
        />
        <input
            type="submit"
            value="add Event"
        />
    </form>
</div>
<FullCalendar
    plugins={[dayGridPlugin, timeGridPlugin]}
    initialView="dayGridMonth"
    headerToolbar={{
        left: "prev,next",
        center: "title",
        right: "dayGridMonth,timeGridWeek,timeGridDay",
    }}
/>
</div>
);
}

export default App;

```

Krok 3: Minimalne odwzorowania stanu interfejsu użytkownika

- Tytuł
- Data i czas

Krok 4: Miejsce stanu

```

import React, { useState } from "react";

import FullCalendar from "@fullcalendar/react";
import dayGridPlugin from "@fullcalendar/daygrid";
import timeGridPlugin from "@fullcalendar/timegrid";

```

```
const DUMMY_EVENTS = [
  {
    title: "Big Meeting",
    start: "2022-03-17",
  },
  {
    title: "Vacation",
    start: "2022-03-15",
  },
  {
    title: "Conference",
    start: "2022-03-24",
  },
];

function App() {
  const [newEvent, setNewEvent] = useState({ title: "", start: "" });
  const [allEvents, setAllEvents] = useState(DUMMY_EVENTS);

  return (
    <div>
      <h1>Calendar</h1>
      <h2>Add New Event</h2>
      <div>
        <form>
          <input
            type="text"
            placeholder="Add Title"
            value={newEvent.title}
          />
          <input
            type="datetime-local"
            placeholder="Start Date"
            value={newEvent.start}
          />
          <input type="submit" value="add Event"/>
        </form>
      </div>
      <FullCalendar
        plugins={[dayGridPlugin, timeGridPlugin]}
        initialView="dayGridMonth"
        headerToolbar={{
```

```

        left: "prev,next",
        center: "title",
        right: "dayGridMonth,timeGridWeek,timeGridDay",
      })
      events={allEvents}
    />
  </div>
);
}

export default App;

```

Krok 5: Przepływ danych w drugą stronę

```

import React, { useState } from "react";

import FullCalendar from "@fullcalendar/react";
import dayGridPlugin from "@fullcalendar/daygrid";
import timeGridPlugin from "@fullcalendar/timegrid";

const DUMMY_EVENTS = [
  {
    title: "Big Meeting",
    start: "2022-03-17",
  },
  {
    title: "Vacation",
    start: "2022-03-15",
  },
  {
    title: "Conference",
    start: "2022-03-24",
  },
];

function App() {
  const [newEvent, setNewEvent] = useState({ title: "", start: "" });

```

```

const [allEvents, setAllEvents] = useState(DUMMY_EVENTS);

const eventTitleHandler = (e) => {
  setNewEvent({ ...newEvent, title: e.target.value });
};

const eventDateHandler = (e) => {
  setNewEvent({ ...newEvent, start: e.target.value });
};

function handleAddEvent(e) {
  e.preventDefault();
  setAllEvents([...allEvents, newEvent]);
}

return (
  <div className="App">
    <h1>Calendar</h1>
    <h2>Add New Event</h2>
    <div>
      <form>
        <input
          type="text"
          placeholder="Add Title"
          value={newEvent.title}
          onChange={eventTitleHandler}
        />
        <input
          type="datetime-local"
          placeholder="Start Date"
          value={newEvent.start}
          onChange={eventDateHandler}
        />
        <input type="submit" value="add Event"
onClick={handleAddEvent} />
      </form>
    </div>
    <FullCalendar
      plugins={[dayGridPlugin, timeGridPlugin]}
      initialView="dayGridMonth"
      headerToolbar={{
        left: "prev,next",
        center: "title",

```

```
        right: "dayGridMonth,timeGridWeek,timeGridDay",
      }}
      events={allEvents}
    />
  </div>
);
}

export default App;
```