FileMaker with React

Bidirectional Communication

Adam Augustin, dotfmp 2024

agametis - Dr. Adam G. Augustin

- Self-employed developer from Munich area
- Developing custom applications for SMBs with FileMaker
  - and meanwhile also with ReactNative for iOS and Android
- Many presentations on dotfmp, FMK (the german FileMaker conference)
- Visited many times the DevCon in the states
- Examples of my projects can be found on my website www.agametis.de

info@agametis.de

Outlook

- “Simple”JavaScript Libraries vs. UI-Frameworks
    - Simple Libraries like Fullcalendar, ChartJS,…
    - UI-Frameworks like React, Vue, Angular, Svelte,…
- How to realise a bidirectional communication between FileMaker and WebViewer

Simple vs UI-Frameworks

Simple JS Libraries

- Libraries can be integrated and used directly in an HTML page with the <script> tag.
- All functions of the library are accessible directly in the window-context.
- With the script step Perform JavaScript in WebViewer and the JS function FileMaker.PerformScriptWithOption(), communication between the two worlds can be implemented directly.
- How does it work? See my presentations at the German FMK 2023 in Basel
    - Demos and presentations are available at https://github.com/agametis/

UI-Frameworks

Like React & Co.

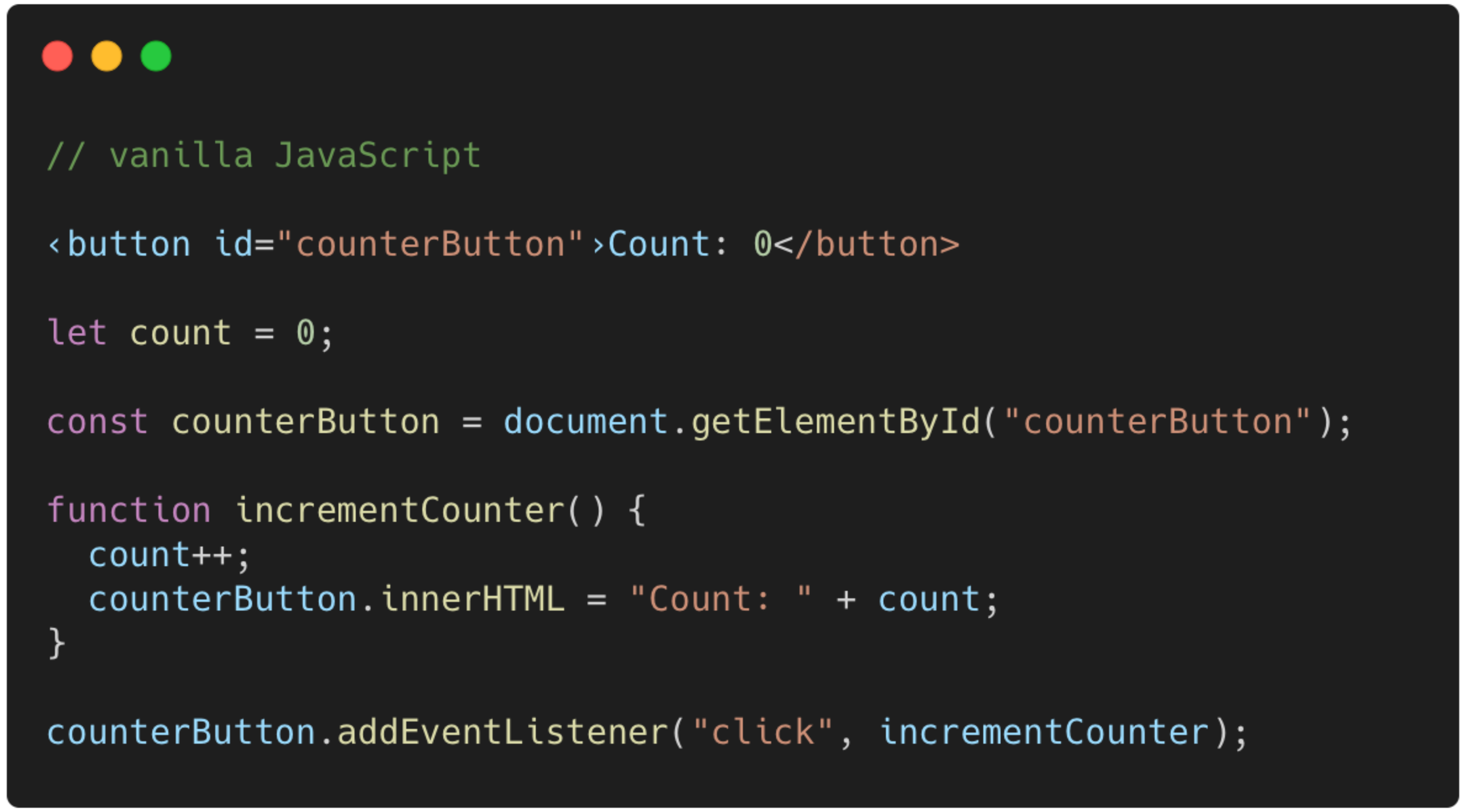
- Library cannot usually be integrated into an HTML page using the <script> tag.
- Programming with a UI framework is usually “supported” by a bundler.
- Advantage: You can use many helpers during development (e.g. TypeScript for programming, Tailwind for CSS) and the bundler takes care of the rest.
- You usually develop on a development server.
- The result of compilation by the bundler are highly optimized JS, CSS and HTML files that have been compiled down to the bare essentials.
- Disadvantage: The functions are not accessible directly. The data can no longer be executed as a “simple” website from the local disk or memory. They must be provided by a web server.

JavaScript is not equal JavaScript

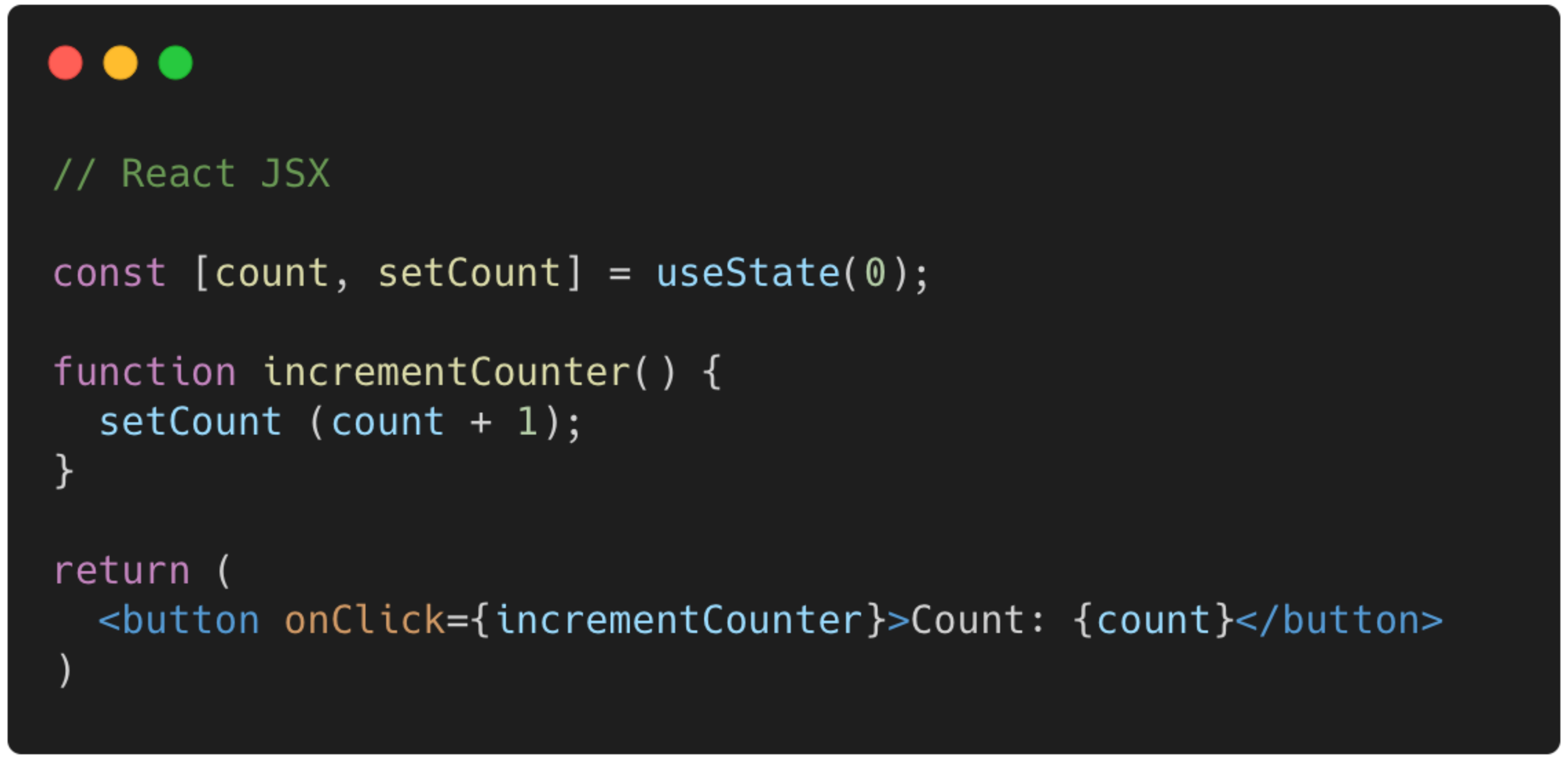
🤔

Vanilla JavaScript vs UI-Frameworks

Example: Increase counter with a button



```
// vanilla JavaScript

<button id="counterButton">Count: 0</button>

let count = 0;

const counterButton = document.getElementById("counterButton");

function incrementCounter() {
  count++;
  counterButton.innerHTML = "Count: " + count;
}

counterButton.addEventListener("click", incrementCounter);
```

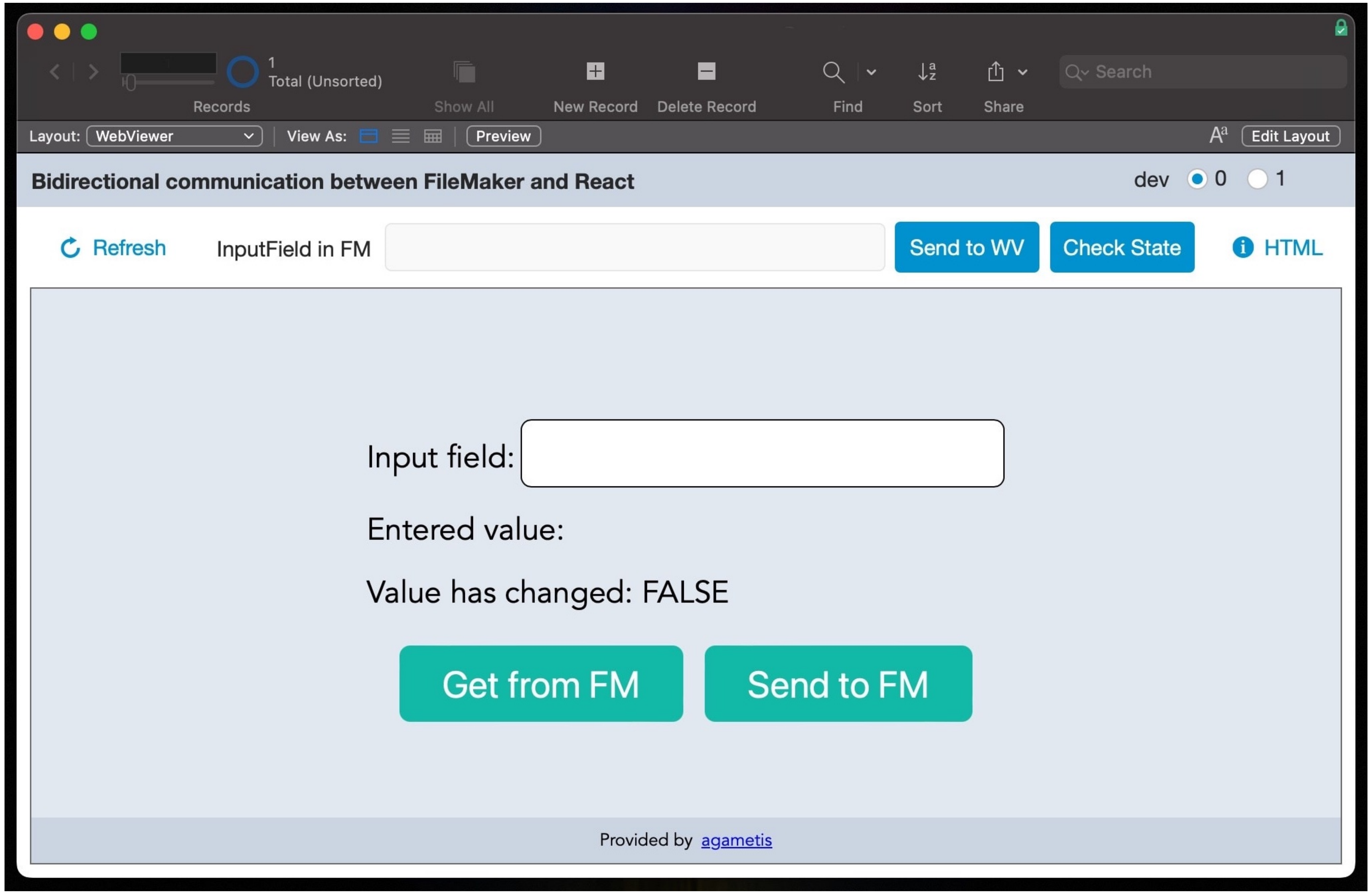


```
// React JSX

const [count, setCount] = useState(0);

function incrementCounter() {
  setCount (count + 1);
}

return (
  <button onClick={incrementCounter}>Count: {count}</button>
)
```

Okay, let’s dive into the demo

Demo

https://github.com/agametis/fm-react-demo



Bidirectional communication between FileMaker and React

dev [✓ 0] 1

Input field:

Entered value:

Value has changed: FALSE

Provided by agametis