IBIMA Publishing
Communications of International Proceedings
https://ibimapublishing.com/p-articles/47AI/2026/4717926/
Vol. 2026 (12), Article ID 4717926

AI In Web Applications for Calendar and Event Management*

Aleksandra TEJSZERSKA-RZEZNIK and Adam KOSTULAK

Institute of Informatics, University of Gdansk
Gdansk, Poland

Correspondence should be addressed to: Aleksandra TEJSZERSKA-RZEZNIK, aleksandra.tejszerska@ug.edu.pl

*Presented at the 47th IBIMA International Conference, 29-30 June 2026, Madrid, Spain

Abstract

Currently, there are many JavaScript libraries on the market for calendar and event management, but most are quite simple and lack flexibility in terms of the elements we can place on the calendar. Most offer a single event type and do not always allow data to be passed between events, or they do so in a rather simplified form. This article presents an analysis of current libraries for creating web applications dedicated to managing calendar-based events. It presents the requirements that web developers place on existing tools and identifies the potential for using AI in the functionalities of such applications. A concept for a new library implementing these functions is proposed.

Introduction

By far the most popular and one of the most comprehensive front-end libraries is FullCalendar. It offers extensive configuration options, support for various views (monthly, weekly, daily, etc.), integration with various data sources, support for user interaction (dragging, resizing), and availability for various frameworks (React, Angular, Vue). Unfortunately, despite constantly updated and improved documentation, it remains unintuitive. Furthermore, not all of its features are available in the free version. While the documentation has been improved in recent months, in most cases, we can find the types that functions operate on. However, in older versions (v4 and lower), many types are still undefined.

The current documentation (from v5 and v6) has undergone significant changes over recent years. Its format remains unchanged, but it is no longer hand-written and is largely generated from source code. By using the TSDoc standard and native type definitions (.d.ts), the descriptions of parameters, methods, and events visible on the official documentation page are identical to those the developer sees in their IDE (e.g., VS Code) via IntelliSense. This eliminates errors resulting from discrepancies between the documentation and the actual behavior of the library and eliminates the need to manually check the code in the node_modules folder.

The library also lacks the ability to define general configurations for calendars placed in the application. Currently, each calendar placed in the application must be configured individually, and the developer must remember to include this configuration when creating a new calendar. The calendar itself is table-based (in basic views), making it inflexible and difficult to style independently. The injected configuration only allows the use of predefined Bootstrap-based themes (v4 and v5). Since v6, the library has moved away from rigid Bootstrap dependencies in favor of CSS variables.

The library itself is clearly written and incorporates numerous design patterns, including:
• Composite – FullCalendar is built as a component that can be easily integrated into applications using various frameworks; the calendar itself is composed of smaller components.
• Observer – used to listen for various events (date change, view change, etc.)

Cite this Article as: Aleksandra TEJSZERSKA-RZEZNIK and Adam KOSTULAK Vol. 2026 (12) "AI In Web Applications for Calendar and Event Management " Communications of International Proceedings, Vol. 2026 (12), Article ID 4717926, https://doi.org/10.5171/2026.4717926

• Factory – used to create views based on the provided configuration
• Decorator – used to customize and extend functions related to user interactions, calendar views, and event handling
• Strategy – each view (Month, DayGrid, Timeline) is a separate data rendering strategy
• Adapter – allows for integration with various data sources, such as Google Calendar

By rewriting the core to Preact and introducing a tree-shaking mechanism, the library has become significantly lighter. The introduction of virtual rendering means that even with thousands of events, FullCalendar is as fluid as minimalist solutions.

Popularly used libraries

React-big-calendar

Despite significant optimization in recent years, FullCalendar remains a very complex library due to its vast functionality. However, it has inspired new libraries that are becoming alternatives better suited to simpler, specific scenarios.

Inspired by FullCalendar, this calendar library for React-based applications offers support for various time views, user interactions, and interface customization. It has fewer features than FullCalendar, which can be an advantage for simple projects. It offers fewer calendar views and a simpler and more limited appearance but is easier to extend with custom functionality. It is based on flexbox, making it much more responsive.

Unlike FullCalendar, which strives to be a "one-size-fits-all" solution, React-big-calendar gives the developer complete control over every element rendered by React.

Event Calendar

A modern, extremely lightweight alternative (approx. 30-40 KB) that visually and functionally mimics FullCalendar while forgoing its most advanced features. It is ideal for projects where page loading speed is key. In recent years, this library has become FullCalendar's main competitor in projects where performance budgets are key.

Kendo UI Scheduler

This library offers a comprehensive scheduler with a user interface reminiscent of Google Calendar or Microsoft Outlook. It supports various time views, interface customization, integration with various data sources, and is available for Angular, React, and Vue.

Kendo UI is a strictly commercial product, requiring the purchase of a paid license regardless of the project. The lack of a free, open-source version means this library will not be considered.

TOAST UI Calendar

Definitely one of the most comprehensive calendars available for free. It has the most configurable views, including multi-week, not necessarily monthly, views.

This is the only library that implements a built-in form for editing and deleting events. The components are fixed and accept specific data. Ideally, forms should be flexible and dynamically generated based on the created model and data type. FullCalendar doesn't have built-in forms, leaving this to the developer (so they can use their own Angular/React components). TOAST UI dictates its appearance. The functions responsible for creating and editing data accept predefined, simple data types. This limits the group of people who choose this library. You can set a "readonly" flag for events, defining whether events can be edited or deleted. Beyond this functionality, the calendar is ultimately less configurable than FullCalendar. While v2 has added more customization options via templates, overriding the default popup logic is still more difficult than building your own from scratch in another library.

The calendar is based partly on flexbox and partly on inline-block elements. This allows for greater flexibility when editing than a table, but due to its complex interface, it doesn't look good at lower resolutions, and it can be unreadable on mobile devices.

The library has undergone a major rebranding and refactoring to version v2. In the JS world, it is now known as @toast-ui/calendar. Importantly, since 2024, development of this library has slowed significantly compared to

FullCalendar and is considered a "maintenance mode" project, which can be a key consideration when choosing a technology for a new project.

Evo-calendar

Provides basic customization options that allow you to tailor the calendar's appearance to the project's needs and is much less complex than FullCalendar or TOAST UI Calendar. It is lightweight, which can be beneficial for projects where minimizing page load is crucial, but its simplicity may be insufficient. Only a monthly view with optional daily details is available. It lacks the ability to display events and their titles from an overview or a weekly view. It's a jQuery-based plugin, and when building a project in 2026, you avoid including jQuery just to support a single component. This increases the application's weight and breaks the architecture (direct DOM manipulation via jQuery conflicts with Angular's change detection mechanism). Furthermore, the last commit on GitHub was in 2020.

An Analysis of library properties

Table 1: Analysis of library properties

Library	Type of license	Advantages	Disadvanteges
FullCalendar	MIT / Commercial (Pro)	Industry standard, native TS, tons of features, CSS variable styling, Virtual Rendering.	Some advanced views (Timeline) require a paid license.
Vue-fullcalendar	-	-	Abandoned project. Unofficial wrapper that doesn't support modern Vue 3 standards.
React-big-calendar	MIT	Created for React, lightweight, uses Flexbox, no imposed styles (full control over the DOM).	No support for other frameworks (only React), fewer ready-made views than FullCalendar.
Event Calendar	MIT	Extremely lightweight (~30KB), no dependencies, free, very high performance.	A poorer list of features compared to the Pro version of FullCalendar.
Kendo UI	Commercial	Enterprise quality, dedicated support, very stable, native to Angular.	High price, no free version for commercial use.
TOAST UI Calendar	MIT	Ready-made popups for editing and deleting events "out of the box", unique multi-week views.	Difficult form customization, slow project development, poor responsiveness.
Evo-calendar	MIT	Very simple, light, specific design (calendar with a list underneath).	Requires jQuery, no week view, no TS support, project inactive.

Examples of flawed solutions

Other libraries not included in the above list due to their limited functionality, being designed for a single framework, or lack of development in recent years:

• Angular Calendar – although a high-quality library and regularly updated, its use is limited to the Angular framework.

• Calendar-js – extremely simplified scripts acting as static date widgets, last updated in 2019

• Caleandar – a basic library with limited functionality (similar to Calendar-js), last updated in 2016

• Vue Cal – A simple calendar library for Vue.js

• CalenStyle – a deprecated solution based on jQuery, discontinued in 2017

• Planby – limited to React applications

AI-Powered Calendar Apps

Calendar AI

Calendar AI is a Next.js application that leverages artificial intelligence to help manage schedule intelligently and efficiently. Calendar AI enables creating and managing events using natural language, automatically rescheduling events, and sharing a personal calendar with others.

Key Features:

• Create Events with Natural Language: Schedule events simply by typing or speaking in natural language.

• Complete Assistant with GPT Integration: Use ChatGPT integration to receive intelligent suggestions and automate meeting and event scheduling.

• Automatic Rescheduling: AI-powered features to automatically adjust calendar events when necessary.

• Calendar Sharing: Share the calendar with anyone, making it easier to coordinate schedules and commitments.

Reclaim

It is an intelligent AI-powered calendar assistant that automatically schedules tasks, habits, meetings, and focus time in Google Calendar or Outlook, reclaiming up to 40% of work time. It flexibly blocks time, reschedules events in case of conflicts, and integrates with tools like Slack.

Key Features:

• Smart Tasks and Habits: Automatically finds the best time for tasks and routines, flexibly adapting to your changing calendar.

• Focus Time Protection: Automatically blocks time for deep work, preventing the calendar from becoming overcrowded with meetings.

• Smart Meetings & Scheduling Links: Simplifies meeting booking and provides intelligent scheduling links that show true availability.

• Calendar Synchronization: Combines multiple calendars (e.g., personal and work) to prevent double-bookings.

• Prioritization: Allows to set tasks as "critical" (P1) to "low priority" (P4), allowing AI to automatically reschedule less important events.

AI-Powered Interactive Event Calendar Creator

Effortlessly create stunning, interactive event calendars using AI technology. Organize and visualize events with ease.

The AI-Powered Interactive Event Calendar Creator is a web application that utilizes artificial intelligence to generate dynamic, interactive event calendars. AI will create a fully responsive, customizable calendar of upcoming events. Application is dedicated for organizations, event planners, and individuals looking to present their events in an engaging, easy-to-navigate format.

Morgen

Morgan is a multi-calendar aggregator (Google, Outlook, Apple) with a built-in AI planner that helps optimize weekly schedules. It offers the following features: Unified Calendar Management, Built-in Tasks and Time-Blocking, AI Planner, Scheduling Links, and Quick Join.

Toki

Toki is an intelligent mobile application (iOS/Android) designed for automated schedule management using AI. It allows instantly creating and organizing events using voice, text, or photos (e.g., screenshots/posters), integrating with Google, Apple, and Outlook calendars for easy scheduling. It features intelligent scheduling, multimodal data entry, contextual awareness, and a proactive assistant. It works with major calendars (Google, Apple, Outlook) to manage all plans in one place.

Conclusions

Currently, there are many useful libraries for creating web applications with calendar functions. Web application developers have access to both licensed solutions and a wide range of open-source projects. Simplicity of design and ease of implementation are paramount when assessing usability. An analysis of existing event management applications on the market clearly indicates the need to develop libraries to provide features related to the automation and intuitiveness of operations implemented using AI mechanisms.

References

- Angular Calendar, *GitHub*, https://github.com/mattlewis92/angular-calendar [Access: 20.03.2026]
- Bryntum Scheduler Documentation, https://bryntum.com/products/schedulerpro/docs/guide/SchedulerPro/basics/calendars [Access: 24.03.2026]
- Calendar AI, *GitHub*, https://github.com/typper-io/calendar-ai [Access: 28.03.2026]
- Calendar-js, *GitHub*, https://github.com/igor-ribeiro/calendar-js [Access: 23.03.2026]
- Caleandar, *GitHub*, https://github.com/jackducasse/caleandar [Access: 23.03.2026]
- Event Calendar, *GitHub*, https://github.com/vkurko/calendar [Access: 20.03.2026]
- Evo Calendar Documentation https://edlynvillegas.github.io/evo-calendar/ [Access: 20.03.2026]
- Evo Calendar, *GitHub*, https://github.com/edlynvillegas/evo-calendar [Access: 20.03.2026]
- FullCalendar Documentation, https://fullcalendar.io/ [Access: 20.03.2026].
- FullCalendar Documentation, *TypeScript Integration*, https://fullcalendar.io/docs/typescript [Access: 20.03.2026].
- FullCalendar Documentation, *CSS Custom Properties (Variables)*, https://fullcalendar.io/docs/css-custom-properties [Access: 20.03.2026].
- FullCalendar Documentation, *themeSystem* https://fullcalendar.io/docs/themeSystem [Access: 14.12.2023]
- FullCalendar Documentation, *CSS Customization* https://fullcalendar.io/docs/css-customization [Access 20.03.2026].
- FullCalendar Documentation, *Roadmap* https://fullcalendar.io/roadmap [Access: 20.03.2026]
- Kendo UI Scheduler Documentation https://www.telerik.com/kendo-angular-ui/components/scheduler/ [Access: 14.12.2023]
- Kendo UI Scheduler, *License agreement* https://www.telerik.com/purchase/license-agreement/kendo-ui [Access: 20.03.2026]
- Medevel.com, *11 Best Open Source Free JavaScript Calendars*, https://medevel.com/os-js-calendar/ [Access: 14.12.2023]
- Morgen, https://www.morgen.so/ [Access: 28.03.2026]
- PlanBy, https://planby.app/ [Access: 24.03.2026]
- Reclaim, https://reclaim.ai/ [Access: 28.03.2026]
- React Big Calendar, *GitHub*, https://github.com/jquense/react-big-calendar [Access: 20.03.2026]
- Shaw A., *React & TypeScript Innovations in FullCalendar v5*, https://fullcalendar.io/blog/2020/05/react-ts-v5-beta [Access: 20.03.2026]
- Syncfusion, https://www.syncfusion.com/javascript-ui-controls/js-scheduler [Access: 23.03.2026]
- Toast UI Calendar, *GitHub*, https://github.com/nhn/tui.calendar/tree/main [Access: 20.03.2026]
- Toast UI Calendar, *Getting Started,* https://github.com/nhn/tui.calendar/blob/main/docs/en/guide/getting-started.md [Access: 14.12.2023]

- Toast UI Calendar, *isReadOnly*, https://github.com/nhn/tui.calendar/blob/main/docs/en/apis/event-object.md#isreadonly [Access: 14.12.2023]
- Toki, https://yestoki.com/ [Access: 28.03.2026]
- Vue Cal, *GitHub*, https://github.com/antoniandre/vue-cal [Access: 23.03.2026]