

+48 534 750 636
bielczynski@gmail.com
linkedin.com/in/janusz-bielczynski/
github.com/xanoosh/
portfolio-react-vite-two.vercel.app/

Janusz Bielczyński - Frontend Developer

SUMMARY

Frontend Developer with 3-4 years of experience in creating and maintaining responsive web applications in React and TypeScript. Commercial skills include unit testing, working with UI libraries (Headless UI, Radix UI) and tools such as Vite and Vitest. Previously 2 years of experience in creating stores and websites in WordPress.

SKILLS

Frontend Technologies: Building React applications with custom hooks development, utilizing Headless UI and Radix UI components, consuming API data enhanced with Tanstack Query, React Router navigation and React Suspense.

Styling and User Interface: Tailwind CSS, building animations with Framer Motion, modern CSS3 layouts, Sass preprocessing, responsive design, mobile-first approach.

JavaScript and Core Technologies: JavaScript ES6+ programming, semantic HTML5 markup, TypeScript implementation, REST API integration, JSON data handling.

Validation and Data Management: Data types and schema validation with Zod, React Hook Form validation implementation, data fetching and caching, application state management with Redux or Zustand.

Dev Tools and Testing: Using Vite for frontend building, unit testing with Vitest and React Testing Library, Git version control system, component documentation in Storybook, Jira project management.

Additional Technologies: PHP backend integration, Smarty knowledge, WordPress knowledge, jQuery DOM manipulation. Using WordPress plugins CPT UI and ACF for WordPress template development.

Languages: Polish (native), English (B2 level - professional proficiency)

EXPERIENCE

HRLink — Frontend Developer - React

08.2022 – 02.2024

- development and maintenance work on ATS system and additional modules written in React
- system redesign - rewriting parts of ATS system to React using modern Frontend technologies
- unit test coverage for new ATS system parts using Vitest and React Testing Library
- building component documentation using Storybook

Gorillas.dev — Frontend Developer - React

09.2021 – 08.2025

- maintenance and development of web application handling printer servicing and rental, written in React
- using Redux for application state management

- utilizing Apollo GraphQL for comprehensive API data handling

DNS Group — Web Developer - Wordpress

06.2019 - 09.2020

- creating website templates in WordPress tailored to client requirements
- direct client contact and discussing project development direction
- search engine optimization of created websites (SEO)

Page Interactive — Web Developer - Wordpress

01.2019 - 06.2019

- involvement in website design and prototyping process
 - creating website templates in WordPress
 - using ACF plugin to build flexible website templates
 - utilizing CPT UI to build appropriate data structures for complex projects
-

PROJECTS

Portfolio:

A database of non-commercial projects with links to individual applications and their code. I wrote the project in React within the Vite environment using TypeScript. I implemented routing using Suspense (Lazy loading). For the project components, I wrote unit tests in React Testing Library integrated with documentation in the form of a component library using Storybook. For state management, I used Zustand. The project is styled with Tailwind.

<https://portfolio-react-vite-two.vercel.app/>

Investment Calculator:

One of the applications from my portfolio, an investment calculator, is based on animated charts implemented using Nivo Charts, which helps visualize the scale of investments when changing investment parameters (inflation, rate of return, costs, years). I wrote the form using React Hook Form, and handled subtle animations with the Framer Motion library.

<https://investment-calculator-red-five.vercel.app/>

Weather App:

Another application from the portfolio, it displays a 5-day weather forecast based on data consumed from an API. For API communication, I used custom React hooks with the help of Tanstack Query. Animations with Framer Motion. The components are headless, I used Headless UI and Radix UI.

<https://weather-app-blond-kappa.vercel.app/>