

Alen Mandić

alenmandic1@gmail.com | <https://alenmandic.pages.dev/> | [GitHub](#) | [LinkedIn](#)

Summary

Junior Web Developer proficient in HTML, CSS, JavaScript, React, NodeJS, Express, MongoDB, PostgreSQL Experienced in responsive design and testing/debugging. Problem-solving abilities, creative thinking with excellent communication skills. Committed to staying up-to-date with the latest web development trends. Deployed multiple projects to the internet using many different services.

Technical Skills

| | |
|------------------------------------|---|
| Programming Languages: | JavaScript/TypeScript, HTML, CSS |
| Libraries & Frameworks: | React, NodeJS, Material-UI , Cypress.io, Express |
| Tools: | Git, MongoDB, Mongoose, PostgreSQL, ESLint, Vite, Jest, NPM |

Projects

Full-Stack Blog-App Backend | NodeJS, Express, MongoDB, Mongoose, bcrypt, JSON web token, REST api

- Backend repository for my fullstack blog app following REST-api development standards.
- Repository follows folder structure best practices for easy scalability and readability. Full user authentication and protected routes are enabled using bcrypt and JSON web token.
- App has been additionally secured with a rate limiter 'express-rate-limiter' and the 'helmet' package, for increased security and protection against many kinds of attacks, including XSS script injections.
- End-to-end testing has been done using Cypress.io, and integration testing has been implemented using JEST and supertest

Full-Stack Blog-App Frontend | JavaScript, React, React-Router, Axios, Material-UI

- Frontend repository for my fullstack blog app, made using React.js for a robust and performant user interface.
- Axios was used to handle services and communication with our backend repository.
- React Router was used to separate the website into relevant pages and to enable link sharing for a better user experience.
- End-to-end testing has been implemented using Cypress.io
- Vite's 'npm run build' command was used to create a production ready minified version of our frontend code.

Weather Application | HTML, CSS, JavaScript, Serverless functions

- I used Netlify's serverless functions to handle backend functionalities.
- Live search bar dropdown menu using debounced fetching and type detection.
- Set any city/location as your default homepage, or open that location on a map.
- GPS button which gets your current device location and displays weather data for it
- Smart app refresh system using recursion

Many other projects and work are on my [GitHub](#) / [Portfolio websites](#) linked above 

EDUCATION

Srednja Škola Jure Kaštelan, Omiš

High school diploma; Electrical Engineering Technician | July 2017

Full-stack Open course /computer science curriculum, University of Helsinki

Core course completed and earned 5 ECTS university credits, passing with grade 4.

[LINK TO CERTIFICATE](#)