



ANDRII P. Full-Stack Developer < ASAP / >

SUMMARY

I'm a proficient **full-stack developer** with a deep focus on modern web technologies. I bring extensive experience across both front-end and back-end development, with a specialization in [React.js](#) [Redux](#) [Next.js](#) [Node.js](#) [Express.js](#), [Nest.js](#), [Firebase](#) and [Supabase](#). I am confident with [AWS](#) (Lambda, Cognito, EC2, S3, etc.) [Docker](#); I also work with [GraphQL](#) for API design and management. My expertise spans multiple databases—[MongoDB](#), [PostgreSQL](#), [Pinecone](#) and [MySQL](#)—giving me a comprehensive understanding of data architecture and handling. In addition to my technical proficiency, I excel in English communication, enabling me to collaborate effectively with global teams. My strengths lie in problem-solving, full-stack application architecture, and delivering scalable solutions. I am passionate about contributing to innovative projects and continuously improving and refining existing applications to ensure optimal performance.

SKILLS

Programming Languages

JavaScript, TypeScript

Technologies

Node.js, Express.js,
Nest.js, MongoDB,
Mongoose, PostgreSQL,
MySQL, GraphQL,
Sequelize | TypeORM,
Twilio API, SendGrid,
Stripe, RabbitMQ,
Mocha | Jest, WebSockets
| Auth0 | JWT, Docker,
CI/CD, AWS | AWS S3,
AWS EC2, AWS
CodeCommit, AWS API
Gateway, AWS Lambda,
AWS DynamoDB, AWS IAM,
AWS CloudWatch, AWS
Serverless Framework,
Next JS, React JS,
React Query, Redux,
Redux-Toolkit, MaterialUI,
TailwindCSS,
styled-components,
SCSS/Less/Sass/HTML5/
CSS3, chart.js,
ApexCharts, Firebase,
Auth0, Jest, Azure
Storage, Azure AD B2C

PROJECTS

1. Front-End Developer, Sparkit

Sparkit is a fan-directed video collab network enabling creators to make authentic non-branded content ranked by fan engagement and real-time voting.

Technologies

Next.JS, React.js, Material UI(components, icons, data-grid), TypeScript, GraphQL, Apollo classnames formik yup, SASS/SCSS, reactour Redux, react-redux

Requirements

- Create flexible and reusable components
- Recycle previous components
- Create the necessary reaction hooks
- Develop and create the project architecture on nextJS13
- Fix many different bugs
- Develop data collection and processing using graphQL
- Improve and optimize previously written components and functions
- Create tables using the "mui data-grid" library
- Create a variety of forms with different input fields
- Create validation for forms

Results

The introduction of flexible and reusable components has significantly enhanced the project's scalability and maintainability. The implementation of reactive hooks adds a layer of dynamism, further elevating the user engagement.

The project's architecture, built on Next.js 13, reflects a forward-thinking approach, embracing the latest advancements in web development. The integration of GraphQL for data acquisition and processing demonstrates efficiency and modern data-handling practices. Tables, powered by the "mui data-grid" library, have been seamlessly incorporated, enhancing data presentation and user interaction.

EDUCATION

Bachelor's degree

Computer science

Sumy State University

LANGUAGES

English

- Spoken: Upper Intermediate
- Written: Upper Intermediate

2. Full-Stack Developer, [Lendistry](#)

Lendistry is an online platform that streamlines the loan application process, allowing business owners to apply for various financing options conveniently.

Technologies

[Nest.js](#) AWS Docker React Sass Next.js, react-modal, antd, ag-grid-react

Requirements

- Add draw down and transfer out flow for borrower
- Fix application flow for borrower
- Add features/fix bugs for underwriter portal related to ag-grid
- Add bash script to load valid credentials for authentication
- Develop duns UI to underwriter-portal
- Fix validation issues and added new validation features
- Refactor existing code and improve build
- Write unit tests
- Fix UI issues and bugs

Results

I successfully implemented the needed features, added the draw down, and transfer-out flow for borrowers, ensuring a seamless process for managing their transactions. I also fixed the application flow for borrowers, addressing any inconsistencies to improve the overall user experience. On the underwriter portal, I worked on features and bug fixes related to ag-Grid, optimizing data handling and display. Additionally, I created a bash script to load valid credentials for authentication, streamlining the login process and enhancing security. I developed the DUNS UI for the underwriter portal, ensuring a seamless user experience. I also fixed validation issues and introduced new validation features to enhance data accuracy. I also refactored existing code to improve maintainability and optimized the build for better performance. To ensure code reliability, I wrote unit tests, and addressed various UI issues and bugs, enhancing overall functionality and user satisfaction.

3. Full-Stack Developer, [Rentberry](#)

Rentberry is a Real Estate platform, where you can buy/sell/rent property.

Technologies

NodeJS Express, TypeScript, AWS, Stripe, Webpack, Google Sign-In, Facebook sign in, Apple sign in.

Requirements

- Write authentication/registration/password reset logic
- Integrate Stripe payment system
- Integrate auto-deployment on AWS
- Integrate ability to upload PDF documents (Rental history, Background info.)
- Perform server-side optimizations
- Troubleshoot and resolve bugs
- Collaborate with front-end team for API integrations

Results

I successfully implemented authentication, registration, and password reset logic using secure methods, and implemented Stripe payment integration to handle subscriptions, one-time payments, and webhook events, ensuring compliance with

security standards. Secondly, I set up auto-deployment on AWS, and integrated the ability to upload PDF documents, such as rental history and background info. I also optimized server-side performance by improving database queries, implementing caching, and enhancing security, ensuring system stability. During the project I worked closely with the front-end team, assisting at all steps of API integration.

4. Front-End Developer, [Obenan](#)

Obenan provides AI-powered Local SEO solutions to increase local businesses online visibility and automate online duties with unsupervised workflow.

Technologies

NextJS, React.js, MaterialUI, TailwindCSS, chart.js

Requirements

- Develop a responsive web app and landing page for SEO Seller.
- Use Next.js (React.js) as the framework.
- Implement MaterialUI and TailwindCSS for styling and responsive design.
- Integrate the Chart.js library to display analytics in the admin dashboard.
- Create a user-friendly interface for selecting SEO promotion plans.
- Ensure the app and landing page are optimized for performance.
- Ensure the app scales to support multiple users and high data loads.
- Prioritize responsiveness and accessibility in both the app and landing page.

Results

I successfully developed a responsive web app and landing page, using Next.js and React.js, implemented MaterialUI and TailwindCSS for styling to ensure a modern, responsive design, and integrated the Chart.js library to display analytics in the admin dashboard, providing users with valuable SEO insights. I also created a user-friendly interface for selecting SEO promotion plans. The app and landing page were optimized for performance and scalability, supporting multiple users and handling high data loads. Throughout the development process, I prioritized responsiveness and accessibility.