



Max Koldun

React Developer

CONTACT

koldun.maks@gmail.com

Linkedin

[koldunmaksym](#)

## SUMMARY

I'm a React developer with over 6 years of experience. I write a code following SOLID principles. Also I use OOP in React.js projects to bind my code with a subject area as much as possible. Also a list of patterns like Factory, Strategy and Singleton are always used in my apps. I like patterns and try to use them where it is possible. I have had experience in estimating IT long-term projects. My English level allows me to communicate with foreign clients efficiently. I develop projects mostly on React/Redux and also have experience with Node js (Express/Sequelize/MySQL). My code is tested by using test libraries like Jest, Enzyme and react-testing-library. I can make application deployments using Docker to set an app on a server. I am a team player and usually work by agile methodology in teams that include up to fifteen people.

## ARTICLES

[Guide of Testing React Components with Hooks & Mocks](#) by Max Koldun

03/2022

## CERTIFICATIONS

[Understanding TypeScript \(Udemy\)](#)

05/2025

[Next.js 15 & React - The Complete Guide](#)

09/2025

## WORK EXPERIENCE

WebbyLab — React Developer

06/2019 — 05/2025

### Safeguard

SafeGard Travel Medicine provides a full range of travel medicine services using an online travel review form that allows you to be evaluated by one of our travel medicine physicians and seen at one of our participating travel pharmacies.

*Technologies:* Next js (App Router), React.js, Redux, i18next, Material UI, Zendesk, Persona, Healthie.

*Responsibilities:*

- I developed a Medical travel form for US citizens. The medical form contains 4 different steps and a key feature that it renders by schema from js object. So updating form does not require changing UI components. All updates can be made through the schema.
- Setting CI/CD.
- Estimation.
- Provide Healthie service integration
- Provide Zendesk chat integration
- Provide Persona verification integration
- Set up OTP verification

### **Graberly**

Graberly provides a user with tools to collect more reviews, such as personalised QR codes and links as well as software that makes it easier to collect reviews. The graberly feedback user interface and the company dashboard are very intuitive and require no training. After submitting the graberly review, the user is prompted to write a review on the linked review portal, such as Google.

*Technologies:* Next js (pages Router), React.js, Redux, Stripe, i18next, antd ui library.

*Responsibilities:*

- It was a developed project but had a bad performance. My task was to improve the product and fix all the bugs that a previous team did.
- Setting CI/CD.
- Developing the Frontend part of the app.
- Estimation.
- Connecting payment system Stripe on UI

### **Radioholland Tur.ai Workflow Diagrams**

A workflow diagram that helps connect different services of the application through the UI nodes. For example a user could drag a node (Fetch Data) on a dashboard then input params in the node settings. Then connect the node with another service like (AI data Tune) set params for AI in the node and then the user can create a report in PDF file by dragging a special block for the task and connect it with a previous step. And it is not over a user can add more blocks to send this reports for external services, for example Zendesk CRM.

*Technologies:* react-flow, React.js, Zustand, Bootstrap, i18next, Chart.js, Bootstrap-daterangepicker, Nile Dashboard UI library.

*Responsibilities:*

- Setting up project.
- Developing the Frontend part of the app.
- Estimation.
- Implementation the diagram with React-flow library

### **Radioholland Tur.ai Orders Dashboard**

An admin panel that shows AI results of parsing orders of sea freight transportation. It shows how many orders were parsed correctly and orders that AI was not able to parse. Unparsed orders an admin can parse manually with the admin panel.

*Technologies:* React.js, Redux, Bootstrap, i18next, Chart.js, React-query, Bootstrap-daterangepicker, Nile Dashboard UI library.

*Responsibilities:*

- Setting up project.
- Developing the Frontend part of the app.
- Estimation.
- Implementation caching system through the React-query lib.

## **Hawk**

Web platform to hold webinars. Application consists of admin panel where lecturer creates an event and invites watchers by email and UI part where watchers joins webinar. The UI part also collects statistics about watcher attention based on video from a web camera.

*Technologies:* TypeScript, React, Redux, Socket.io, face-API.js, WebRTC.

*Responsibilities:*

- Fixing BE API.
- Mentoring other developers, review code.
- Developing FE part, creating integration with BE through web-sockets.

## **Pfizer BIM**

A dashboard of different brands of vaccines (Pfizer, Moderna, Novavax). It displays information about the vaccine, how much it costs, and what impact it has on the human body. For example, a percentage of myocarditis happens in patients when they consume one or another vaccine.

*Technologies:* React/Redux, Node.js.

*Responsibilities:*

- Developing frontend part.
- Creating mark up.
- Set up state management.
- Set up a connection to the API.
- Mentoring developers on the project.

## **SMTP onboarding**

An application that allows to set up, and verification of emails and domains. The set emails/domains are gonna be used to send messages by SMTP platform.

*Technologies:* Node.js, Express, Docker, Docker-compose, Confme, ESLint, LIVR.

*Responsibilities:*

- Developing BE API.
- Set up docker, docker-compose to start a project (FE + BE) by a single command.
- Developing auth system. The system had to keep an authenticated user on two platforms that are deployed on different servers with different domains.
- Project estimation.
- Hardening existing code.

- Developing FE part, creating integration with BE.

## PIM

PIM-system — an application that allows to gather, manage, enrich product information and distribute it to eCommerce channels.

*Technologies:* Node.js, Express, Sequelize, MySQL, MSSQL, Bull, Lodash, LIVR, Faker, Multer, Swagger, ESLint, Docker, Docker-compose.

*Responsibilities:*

- Developing BE features.
- Developing a system of merging Databases.
- Developing a system of a filling database with a large amount of records (over 100000 products).
- Project estimation.
- Hardening existing code.
- Mentoring.

## FDA Reports

The project is an online service that helps a user to collect information about newly released drugs that are released on the website <https://www.fda.gov> and shows a list of user contacts (from LinkedIn) that can be interested in the released drugs. User can send a message to his contacts through the service. He needs to choose the contact, select a message template and click on a button 'Message'. The project contains two parts. 1 FE (UI that was implemented on React/Redux) and BE implemented on Node js, database Mysql. Projects were being automatically deployed on Amazon using gitlab CI and EC2 service on every MR in the main branch.

*Technologies:* i18next, React-i18next, React-router, Material UI, Redux, Redux-persist, Redux-toolkit, Notisnack, React-text-truncate, WebbyLab-config, React-query, Docker, Nginx, CRA, Dayjs, LIVR, Node.js, Sequelize, MySQL, Docker, Docker-dompose.

*Responsibilities:*

- Communicating with clients.
- Developing UI part.
- Developing BE part.
- Project estimation.
- Code Reviews of other devs in a team.
- Mentoring.
- Set-up CI/CD system.

## Wellana

The project is an online vegetarian restaurant where users can order food and admins can update information about dishes through the admin panel. It contains three parts. 2 FE (Admin panel and UI that were implemented on React) and BE implemented on Node js, database Mysql. The unusual part of the app is that it supports RTL as one of the requirements was to add Hebrew translation for the app on the UI and Admin parts. UI also supports xml tags like glossary and highlights which allows admins to create more rich content through wysiwyg editors. Admin panel was built mostly on JSON schemas and all schemas were rendered through one react

component. The admin panel supports different roles for admins (content-manager, superadmin, viewer, etc) and gives the ability to read/write all content of the app. The admin panel has a preview mode for the content. Projects were being automatically deployed on Amazon using gitlab CI and ECS service on every MR in the main branch.

*Technologies:* Next.js (pages router), i18next, React-i18next, React-router, Tailwindcss, Classnames, SCSS, Redux, Redux-persist, Redux-toolkit, React-modal, React-toastify, WebbyLab-config, React-query, Docker, Nginx, CRA, React-draft-wysiwyg, React-markdown, Moment, LIVR, Docker, Docker-compose.

*Responsibilities:*

- Communicating with clients, providing demo.
- Developing Admin part.
- Project estimation.
- Developing UI part.
- Fixing BE part.
- Hardening existing code.
- Code Reviews of other devs in a team.
- Set-up CI/CD system.
- Set-up View/Admin/Default modes of the application.
- Set-up PWA.
- Set up Google analytics.

### **Realife-tech Easel**

A British project is an admin panel to manage content displayed to users via mobile applications or web browsers. Content means things like News, Video, Galleries, Banners, Buttons, etc. The FE has the following tools to organize and manage content:

- Content management system, the feature is represented as a list of tables and forms. It supports CRUD operations, translation, sorting, and filtering for every entity. All entities are displayed in a table component. The table component is very flexible and has various widgets to display fetched information inside of the table. Create and update forms are different for every entity.
- Canvas, one of the primary workspaces in Easel, is a tool that automates the creation and management of personalized screens, primarily for a display on smartphones. It allows system users to treat every screen of an app or web app as a blank “canvas” which can be populated automatically with the most relevant content for each end-user based on business rules controlled by the system user.
- Campaigns, a tool that is used by venue marketing teams. With this tool they turn the data available to them into web components or push notifications. These web components and push notifications are used with the goal of driving people to perform a certain action, like buying a product or attending an event. The FE provides a modal window to create and update these templates of web components and notifications.

*Technologies:* React, Redux, Reduxjs/toolkit, React-beautiful-dnd, Redux-persist, Lexorank, Jest, React-testing-library, GraphQL.

*Responsibilities:*

- Communicating with foreign clients.
- Developing an architecture of the CMS, Canvas and Campaigns.
- Project estimation.

- Creating new project features.
- Covering code by tests (jest, react-testing-library).
- Hardening existing code.
- Creating tickets in JIRA for developers and QA.
- Code Reviews of other devs in a team.

### **Realife-tech Web Ordering Journey**

An ordering journey solution to clients with existing mobile apps, and potential clients who do not want apps in the future. The WOJ (Web Ordering Journey) is a tool that allows you to buy products online at different events (football matches, concerts, etc) that are fully customizable by styles, localized and can be used as an independent web app from the browser and as built-in Web View in the existing mobile app. To manage remote data we used apollo client with GraphQL.

*Technologies:* React.js, Material-ui, @apollo/client, GraphQL, i18next, Stripe, Jest, Enzyme.

*Responsibilities:*

- Setting up payment system using Stripe, 3d-secure.
- Covering code by tests (jest, Enzyme).
- Providing app localization using i18next lib.
- Project estimation.
- Creating an app automatic authentication system without straight auth form but using configs which are located on the Firebase store.
- Creating a system of styles customization of the app based on the config admin set in firebase with material ui.

### **Realife-tech Order-Dashboard**

A British project is an admin panel to manage concessions. The system shows a user's interaction with concessions, particularly what users buy, the type of order they choose and how long clients wait for orders.

*Technologies:* React.js, Redux.

*Responsibilities:*

- Communication with foreign clients.
- Project estimation.
- Project planning.
- Making front end part of the project on React, Redux.

### **Access-control**

The system allows you to manage checkpoints to buildings using RFID and NFC marks. The system allows managing time when users can be allowed to buildings. A user's interaction with marks displays to the admin and calls alert if there are any illegal intrusions.

*Technologies:* React, Redux, Docker, CI/CD.

*Responsibilities:*

- Setup CI/CD using Docker.
- Project estimation.
- Making front end part of the project on React, Redux.

## Dtek

The system allows you to create and validate tenders, you have six roles, seller, customer, admin, initiator, lawyer, commission. They can collaborate with each other via chats, the initiator and the seller set a tender and the admin checks the validity of the tender and if everything is good, the seller can start the auction, the customer can make a proposal, if the customer wins, the lawyer helps him with paperwork.

*Technologies:* React.js (Material UI).

*Responsibilities:*

- Making the front end part of the project on React.
- Taking participation in weekly meetings.

## NDA UI Framework

It is an IoT project. We develop an application to manage real devices from your smartphone. With a few examples of usage, you can set the position of a blind (open, close), regulate brightness and power on your lamps, listen to music and monitor air quality in the house. Application is managed via nva queries, nva is developed with a language which allows you to collaborate with a whole system via core (central broker that manages all devices by MQTT protocol).

*Technologies:* React.js, React Native, Nodejs, MongoDB, Postgresql, Amazon S3, Docker.

*Responsibilities:*

- Developing a CLI for saving, downloading, monitoring and deleting bundles on Amazon S3 service written on Node.js.
- Taking participation in weekly meetings.
- Project Estimation.
- Developing Video Player on React Native.
- Making the front end part of the project on React.

Upwork

11/2018 — 06/2019

## Athena Tools

*Technologies:* React.js, Node.js, Zuora api, Firebase.

*Responsibilities:*

- Front end part.
- Communication with foreign clients.
- Project Estimation.
- Creating a proxy server on Node.js for communication with Zuora API.
- Integrating Zuora API as a payment system on the project.
- Deploying the project on Firebase (Front end) and Heroku (Back end).

## ShoSho

*Technologies:* React.js, MySQL.

*Responsibilities:*

- Front end part.
- Communication with foreign clients.
- Project Estimation.
- Integrating API of languagetool.

- Integrating thesaurus api.
- Making text on the page editable with using Editor.js library.

AVA.codes — Front End Developer

10/2018 — 05/2019

### **Emmet.ai**

*Technologies:* React.js, Stripe, Firebase, MySQL.

*Responsibilities:*

- Developing frontend on React.
- Deploying on the project on Firebase.
- Integrating Stripe as a payment system on Front end.
- Project Estimation.
- Communication with foreign clients.

### **Bookapp**

*Technologies:* React.js, Node.js, MySQL, MSSQL, Access.

*Responsibilities:*

- Developing frontend on React.
- Deploying on the project on Heroku.
- Project transferring from Access on the WEB.
- Communication with foreign clients.
- Creating API for getting data from MySQL.

Binary Studio — JavaScript Developer

05/2018 — 09/2018

*Technologies:* React.js, Redux, Node.js, MongoDB.

*Responsibilities:*

- Developing projects on React, Node.js.
- Taking participation in weekly meetings.
- Tasks estimates.
- Suggestions about improving and developing features in a project.
- Learning new technologies if it's needed.

## **QUALIFICATIONS**

### **Technologies**

- *Technologies:* React.js, Next.js (app and pages routers), Redux, Redux-thunk, Wysiwyg editors, dnd-lbs, React-query, Redux-toolkit, Jest, Babel, Material UI, Stripe, Enzyme, React-testing-library, Node.js, Express, Sequelize, MySQL.
- *Infrastructure:* Firebase, Heroku, GitHub, BitBucket CI/CD, Gitlab, Gitlab CI/CD, Docker, Docker Compose.
- *Tools:* Jira, Confluence, Trello.

### **Languages**

- English — upper-intermediate.
- Ukrainian — native.



**EDUCATION and COURSES**

National Technical University of Ukraine 'Kyiv Polytechnic Institute',  
Automation and computer-integrated technologies, Bachelor's Degree.

09/2015 — 06/2019

National Technical University of Ukraine 'Kyiv Polytechnic Institute',  
Automation and computer-integrated technologies, Master's Degree.

09/2019 — 12/2020