



# Philip Balo

Senior full-stack JavaScript Developer

Email [philip.balo@getapp-group.com](mailto:philip.balo@getapp-group.com)

## SUMMARY

I am a full-stack JS software engineer experienced in creating single-page web applications, mobile apps, and web services using JavaScript, React, React Native and Node.js. Proficient in developing client-side applications based on the blockchain technology. Experienced in developing web applications for the currency exchange, healthcare, ecommerce, and travelling domains.

## AREAS OF EXPERTISE

Developing client-side applications using React and Next.js

Developing mobile-side applications using React Native

Developing server-side applications using Node.js/Express

Maintaining production applications and testing their performance

## EDUCATION

**Igor Sikorsky Kyiv Polytechnic Institute**

**Speciality** Software Engineering  
**Degree** Master

## Technical Proficiencies

### LANGUAGES

JavaScript, TypeScript

### BACKEND

Node.js, Express, Apollo, GraphQL, Koa

### CLIENT-SIDE

React, React Native, Next.js, Redux

### UI

Twitter Bootstrap, Material-UI, Antd, Styled Components, Sass/SCSS, Less

### TESTING TOOLS

Jest, Karma

### BUILD TOOLS

Webpack, Docker

### CLOUD SOLUTIONS

Amazon S3, Google Maps, Google Firebase, Amplify

### DATABASES

MySQL, PostgreSQL, DynamoDB

## Employment History

### GetApp-Group (September 2021 - Now)

Tech Lead in JavaScript department

- Leading professional growth of newcomers in JavaScript department
- Mentoring developers with hard solutions
- Performing R&D tasks on the projects to support incremental technology updates
- Developing client side of web and mobile apps with React, Next.js, React Native and Redux
- Developing server side web and mobile apps with Node.js/Express
- Implementing complex, responsive, and adaptive web interfaces

### SysGears (December 2018 - September 2021)

JavaScript Software Engineer

- Developing client side of web and mobile apps with React, React Native and Redux
- Developing server side web and mobile apps with Node.js/Express
- Implementing complex, responsive, and adaptive web interfaces
- Contributing to the Apollo Universal Starter Kit project

### S-PRO (May 2017 - December 2018)

#### JavaScript Software Engineer

- Implementing client-side applications with React, Next.js, and Redux
- Developing responsive user interfaces with Twitter Bootstrap and SCSS
- Developing the event-driven client-side applications using WebSockets
- Creating complex diagrams and tables for advanced analytics

### ProfIT Company (November 2016 - May 2017)

#### JavaScript Software Engineer

- Developing the event-driven client-side applications using the WebSocket API
- Creating responsive web interfaces using Styled Components and LESS
- Building client-side applications with React, Redux, and Saga

## Professional Experience

---

### | Services Marketplace

#### Description

B2C two-sides mobile application that allows publishing some kind of services that you require to be done from one hand and apply to execute the previously published work from the other hand. The platform was built for Finnish market.

#### Responsibilities

Adding new functionality on both server side using Google serverless architecture (Firebase, Functions, Auth) and client side using React Native for mobile application. Working with existing codebase to do refactoring, bug fixing.

#### Technology Stack

TypeScript, React Native, Expo, Redux, Redux-Saga, GCP services (Firebase, Functions, Auth)

### | Medical Orders System

#### Description

B2B application for pharmacies and medical hospital centers that allows authorized personnel to create and schedule automatic bulk orders for medications and other pharmacy products when it is suitable. The service is being used by large Finnish medical organisations.

Medical Orders System is a single page application and represented by a server side GraphQL API and a web client. The client side is implemented with React / Redux, while the API is running using Express / GraphQL.

#### Responsibilities

Adding new functionality on both server side using Node.js / Express and client side using React and Redux for web application. Updating code structure to the latest available versions of AUSK. Provide the client with refactoring and bug fixing.

## Technology Stack

JavaScript, Apollo v2, GraphQL, PostgreSQL, React, Redux, Express, SCSS, Webpack, Yarn, AntDesign

## | Apollo Universal Starter Kit

### Description

Apollo Universal Starter Kit ([github.com/sysgears/apollo-universal-starter-kit](https://github.com/sysgears/apollo-universal-starter-kit)) provides all the necessary tools to start developing universal JavaScript projects, and incorporates the newest approaches and technologies for building web and mobile applications.

The starter kit enables developers to implement universal components that work on every platform – web clients, web servers, and mobile applications. It also provides hot code reloading for each platform to speed up development, and allows developers to create native mobile applications that use native UI elements, just as if they were developed with Objective-C or Java.

### Responsibilities

Refactoring Apollo Universal Starter Kit modules to use the latest functionality provided by Apollo. Re-working the user module implementation into two separate modules to separate the concerns.

### Technology Stack

JavaScript, TypeScript, React, Redux, Expo, Apollo, GraphQL, Node.js, Express, SCSS, webpack

## | Delivery Service Platform

(name is under NDA)

### Description

The Delivery Service Platform is a custom marketplace designed to help carriers earn money by delivering things, and help users save money on ordering deliveries. The platform includes a communication portal where people can browse delivery options, place orders, and talk to delivery companies. Users can specify various data about their package such as the size, weight, and other, while carriers can choose the packages to deliver from the list of user requests. The application automatically calculates the delivery rates and charges the users to transfer the money to a linked account of a delivery company or an individual carrier.

The Delivery Service Platform client side was created with React and Redux and consists of the user and carrier components. The platform's back-end consists of a Node.js proxy server that communicates with the client side and also partially acts like a proxy by sending queries to the PHP-based back-end API.

### Responsibilities

Developing the user account module with the functionality to enable users to manage their personal and payment data. Implementing the internationalization solution for the client-side application using i18next. Creating the Google Maps-based module to let the users view the route of their items.

### Technology Stack

Node.js, Koa, React, React-Router, Redux, Saga, i18n, Google Maps, Styled Components

## | Cryptocurrency Exchange

(name is under NDA)

### Description

The Cryptocurrency Exchange platform allows users to track the live exchange rate, currency fluctuation, comparative value, and potential growth of various cryptocurrencies and earn money by investing into the chosen cryptocurrencies. The exchange platform uses advanced analytics to build diagrams and tables to show the fluctuations for each currency in real time in order to help users track the current exchange rates and purchase the cryptocurrency at the best possible time.

The back-end Node.js and Express application continually aggregates all the data about each registered cryptocurrency and sends it to the client-side application. The client side application, built with React and Redux, was integrated with MetaMask, a Google Chrome plugin, and web3.js to enable users to trade a cryptocurrency.

### Responsibilities

Implementing real time update functionality using WebSockets to get the latest data about cryptocurrencies from the server-side Express application. Building the client-side charts with the ECharts.js library to continually re-render according to the data received from the server. Integrating the application with the Google Chrome plugin to provide the purchasing functionality of the cryptocurrency using the blockchain technology. Developing the client-side user authentication and authorization modules.

### Technology Stack

Node.js, Express, React, React-Router, Redux, WebSocket, SCSS, web3js, MetaMask, Google Captcha

## | Initial Coin Offering Platform

(name is under NDA)

### Description

Initial Coin Offering Platform helps companies to organize campaigns to sell cryptocurrency and raise funds. A company can organize an ICO for their own cryptocurrency in a bid to raise money for expansion, while users can purchase cryptocurrency at various stages of the ICO, view how much money they spent and the daily, weekly, and monthly predictions about cryptocurrency fluctuations.

ICO Platform is a blockchain-based solution that consists of the back end application for calculating and analyzing the current state of the ICO, and the front-end application that enables companies, user, and ICO Platform administrators to handle their data and watch various ICO statistics and predictions presented in tables and diagrams. The platform also includes a dedicated service letting users move out money from the ICO.

### Responsibilities

Implementing the advanced analytics system and dashboard to enable users manage their data. Developing the administration module to enable platform admins manage users and user data. Implementing authentication and authorization modules for the client side.

### Technology Stack

Node.js, Express, React, React-Router, Redux, WebSocket, SCSS, Blockchain (Solidite), Ethereum

# | Commodities Exchange Platform

(name is under NDA)

## Description

Commodities Exchange Platform is a blockchain-based solution to help companies sell their products and to allow users to participate in trading in order to earn money. The platform provides advanced analytics for companies and users to let them view and calculate the growth or decline of the product's cost, while administrators can manage trading, control the access levels for companies via licencing, and help users stop trading in case of failing of a campaign.

## Responsibilities

Building a dedicated access module with fine-grained permissions to control how companies and users can access certain application features. Creating the data analytics module to let the users, companies, and platforms' administrators to view live statistics of trading sessions. Refactoring the codebase and re-implementing application features after the users' feedback. Developing the client-side authentication and authorization modules.

## Technology Stack

Node.js, Express, React, React-Router, Redux, WebSocket, Styled Components, SCSS, AWS EC2