

responsive to user input. In addition, React is oriented to high performance in cases with huge animations and amounts of interactive nodes, there are many possibilities to optimize performance in such areas as data visualizations.

*D3.js (Data-Driven Documents)* is a JavaScript library for producing dynamic, interactive data visualizations in web browsers. It makes use of Scalable Vector Graphics (SVG), HTML5, and Cascading Style Sheets (CSS) standards to create huge graphical data visualizations.

*Three.js* is a cross-browser open-source JavaScript library and application programming interface (API) used to create and display animated 3D computer graphics in a web browser using WebGL (engine for working with 3D in web).

Three.js allows the creation of graphical processing unit (GPU)-accelerated 3D animations using the JavaScript language as part of a website without relying on proprietary browser plugins. This is possible due to the advent of WebGL.

*Npm* is a package manager for the JavaScript programming language. npm, Inc. is a subsidiary of GitHub, an American multinational corporation that provides hosting for software development and version control with the usage of Git. It is the default package manager for the JavaScript runtime environment. It consists of a command line client, also called npm, and an online database of public and paid-for private packages, called the npm registry, allowed to store software solutions as public code libraries.

**Maksim Yepifanav**

(Fr. Skorina GSU, Gomel)

Scientific advisor **Natallia Aksionava**, senior lecturer

## **JSC ITSUPPORTME DATA VISUALIZATION LIBRARY ARCHITECTURE AND DESIGN**

Implemented library was designed according to layered architecture. The specification is that application splits to the layers, each of them play its own role in software product. Layers – are the logical groupings of the software components that make up the application or service. Layered architecture provides patterns where the components organized in horizontal or hierarchical layers. This is the traditional method for designing such software as different libraries, which represent API (Application Programming

Interface) for developers. All the components are interconnected but do not depend on each other, which helps to design software using SOLID principles.

The library consists layers, which interact to each other and provide define functionality.

Layered architecture schema of the library showed at figure 1.

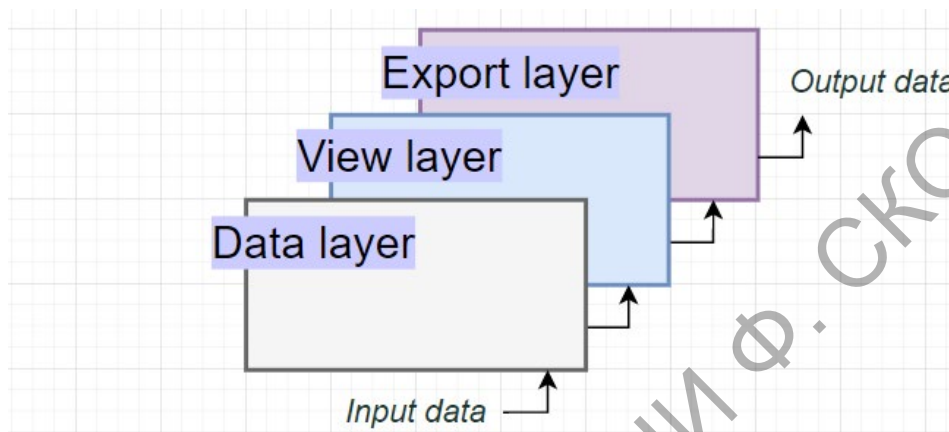


Figure 1 – Layered architecture schema

There are three layers in this architecture where each of them has a connection between modularity. From bottom to top, they are:

1 Data layer – consist logical part of flow that represented as module. Represented as React-hook functions, these handle input data, make everything ready for a rendering.

2 View layer – consist components of D3 and Three.js these after build into core component that ready to export. In fact, create output data.

3 Export layer – consist all of the components to make it easy to export by developers. Represented as root Typescript file in root part of sub-library directory.

The implementation of library was focused on the user roles and their interaction development. In the process, three sequence diagrams were created, each of them describes unique user case.

At all three user roles were identified: Developer-user (usual user of library's API), Contributor (side developer, who wants to make changes into the library, support it), Chief developer (same as Contributor, but with all possible access and permissions).

Project testing was in form of integrating a couple of library data visualizations into existence company's projects.

Based on all of that, product development and testing were carried out.