

**KODIBOX**

PLATFORM THAT PROVIDES EFFECTIVE DEVELOPMENT,  
TESTING AND OPERATIONS OF INFORMATION SYSTEMS  
BUILT ON MICROSERVICE ARCHITECTURE

Your company has a system which development (project) is underway - one, two or more. It does not matter for whom and for what the system is being developed: for the company's own needs or to order.

There are 2 teams working on the project:

- ➔ 1 Feature 1 - Team 1
- ➔ 1 Feature 2 - Team 2

Feature 1 and Feature 2 are 2 components of one project. At the end of development, these components must be integrated.

In the process of developing Features 1 and 2, teams and team members face the following issues:

- How to deploy infrastructure?
- What is the development process?
- How and where to deliver the result of their work?
- Where is the project knowledge stored?

Each project has the following “knowledge”:

- Documentation;
- How-to;
- FAQ about the problems and their solutions.

At some point, the teams need to integrate with each other and they have the following questions:

- How to connect feature 1 to feature 2 build?
- How to lead joint software development?

- Where is the feature documentation located?
- How to monitor performance and debug work?
- How to test features?
- How not to break the environment within which the work is done?
- And hundreds more questions.

Thus, “parasitic” activity occurs for the team members, “crutches” and “rakes” appear in the project. The project ceases to be transparent. As a result, teams suffer and users suffers.

👉 With **kodiBox** team members no longer worry about anything other than their work. Thanks to isolated assemblies, they are independent of each other and can always start over. The platform itself takes care of all overhead costs: deployment, monitoring, logging, tracing, coordination, dependencies, cost accounting, etc.

👉 As a result, you simply register in the platform, form a declarative file with a description of the system. Press the button and do the assembly. As you need, where you need and when you need. Develop, test, collect logs, monitor recycling, integrate with the outside world. Everything becomes transparent.

The current situation in the world of software development shows that companies experience approximately the same problems in developing services:

- ★ Lack of expertise in building large multi-service or micro-service architectures;
- ★ Lack of standardization of development approaches, testing, monitoring and organization of environments;
- ★ Lack of intelligible automation;
- ★ Lack of understanding and control of support and development of services costs;
- ★ Lack of coordination in multi-team and multi-component development.

An analysis of potential demand showed that each company solves these problems in its own way, depending on the availability of a particular expertise.

At present time there is no formed and formulated understanding of the need for a new solution when developing a multi-component product in multi-team development on the market.

👉 **We see the necessity and prospects of such a solution. Therefore, we are creating a new product designed to consolidate accumulated world experience in such development and present it as a completely new solution.**

### Tasks Can Be Solved

- ➔ System Design
  - How the service being developed fits into the current architecture;
  - How the service being developed fits into the current infrastructure;
  - What standards need to be maintained;
  - What processes to launch to get started;
  - What teams will be connected to the development;
  - How teams will interact during development (whether the product will be transferred from team to team).
- ➔ Development
  - How to deploy infrastructure?
  - What resources do we have?
  - What is the development process (branches, gitflow, etc.)?
  - How to deliver the result of work to the environment?
  - ★ Pipeline Creation in CI
  - ★ Assembly
  - ★ Testing
  - ★ Security
- ➔ Documentation
  - Various documents;
  - Wiki.
  - ★ Standards

### Key Functionalities

- ☞ Unified knowledge base about services
- ☞ Support of various providers: GCE, AWS, DO, KVM, Azure
- ☞ Integration with languages: Go, Scala, Java, PHP, C++, Python, Ruby, NodeJS
- ☞ Integration / providing of monitoring services: cAdvisor, Prometheus Node Exporter
- ☞ Logging via the use of container-based infrastructure
- ☞ Kubernetes-based deployment
- ☞ Budget
- ☞ SSL
- ☞ Providing infrastructure for testing on the basis of k8s
- ☞ Support of hybrid infrastructure Kubernetes \ Docker \ Bare Metal
- ☞ Service Discovery: Consul, k8s

Consulting services are provided to clients - assistance on one-time issues, as well as a full cycle of implementation and support

### Target users

- ★ Developers;
- ★ Testers;
- ★ System administrators;
- ★ Product owners;
- ★ DevOps;
- ★ SRE: Site Reliability Engineering.

### Target Audience Segmentation

- ➔ Software development companies:
  - ★ Start-ups;
  - ★ Companies for which software development is a non-core asset, but such work is provided by internal IT resources (banks, stock exchanges, etc.).
- ➔ Engineers, individual users (both individually and internally):
  - ★ Developers;
  - ★ Testers;
  - ★ System administrators;
  - ★ Product Owners;
  - ★ DevOps;
  - ★ SRE

### Cloud Platform Services

- It is carried out in the form of providing access to the platform as a cloud service;
- Target audience: companies and individual users.

The user is given the opportunity to use the platform as a service delivered using cloud technologies.

### Consulting And Technical Support

- It is carried out in the form of contract support;
- Target audience: companies (main target audience) and individual users.

Providing the best combination of tools and expert recommendations to optimize performance, manage risk and control costs.

### Audit And Consulting Of Current Company Systems

- It is carried out in the form of a one-time service;
- Target audience: companies.

We offer Infrastructure Event Management (IEM), which includes architecture and scaling guidance, as well as operational support in preparing and conducting planned events such as holidays, product launches, or migration.

### Implementation Of The Platform On Client Side

- It is carried out in the form of a one-time project according to contract;
- Target audience: companies.

In case client has a circuit that is closed from external access (banks, state structures), the platform can be deployed directly in the client's data center at its facilities. Subject to technical availability.

The exact cost will depend on the specific configuration of the client's infrastructure.

#### The main services:

- ★ Cloud platform services;
- ★ Consulting and technical support;
- ★ Audit and consulting of current company systems;
- ★ Implementation of the platform on client side.

The company has a system - one, two or more.


The components of these systems are different: code, servers, processes, people (users, developers, managers, admins, etc.). They contain many points of interaction and connections, which often change a lot and interact with each other.

Each part individually can do its job well. But the weak point is the connection between them.

This leads to problems and conflicts in integration and interaction. Ultimately, the system gets out of your control. The user suffers, the business suffers.

We can work with different systems - simple and complex.

We provide a platform in which mechanisms of the best interaction practices are implemented, and most integration issues and reactions to changes within the system are addressed.



Using our platform, you regain control of your IT system