

# Kubernetes Empowerer via API (KEA)

Task 3

# Project statement

The product is a platform for deploying, managing, and scaling machine learning models in a production environment. It's primary purpose is to provide a flexible and secure environment for automating ML processes, including model versioning, request routing, and monitoring. The system integrates with Kubernetes, supports model containerization. The product is designed for developers, ML engineers, DevOps teams, and enterprises that require a stable, scalable, and resilient infrastructure for their ML projects.

Team K8C: Daniel Tsurkan; Gadji Dandamaev; Tsaturyan Konstantin; Smolkin Mikhail

Project repo: <https://github.com/fanglores/Advanced-Software-Design>

This report: [here](#)

# Roles

## ML Engineer

**Description:** This role joins professionals involved in the development, deployment, and monitoring of ML models. They want to simplify the deployment process, automate API documentation, and ensure efficient request validation and caching, ultimately enhancing their workflow and model performance.

## API Consumer

**Description:** This role includes all users interacting with APIs to integrate ML models into their applications. They want to access reliable and well-documented APIs, enabling seamless integration of ML models into their business applications and ensuring optimal performance and usability.

# Personas

## ML Engineer (Maria, 32 years old)

### Goals:

- Deploy and version ML models in Kubernetes.
- Automatic API documentation and request validation.
- Flexibility for different ML frameworks.

### Pain points:

- Manual API documentation.
- Difficulties in monitoring model performance.

## Backend Developer (Alexander, 28 years old)

### Goals:

- Use automatic OpenAPI schema generation.
- Easily add API endpoints with request validation and security.

### Pain points:

- Manual API documentation.
- Challenges with integrating authorization and managing access control.

# Personas

## **API Consumer (Sergey, 30 years old)**

### **Goals:**

- Get documentation for quick access to ML models.
- Work with reliable and validated APIs.

### **Pain points:**

- Incomplete or outdated documentation.
- API instability and delays.

## **Corporate Client (Yandex, Sber)**

### **Goals:**

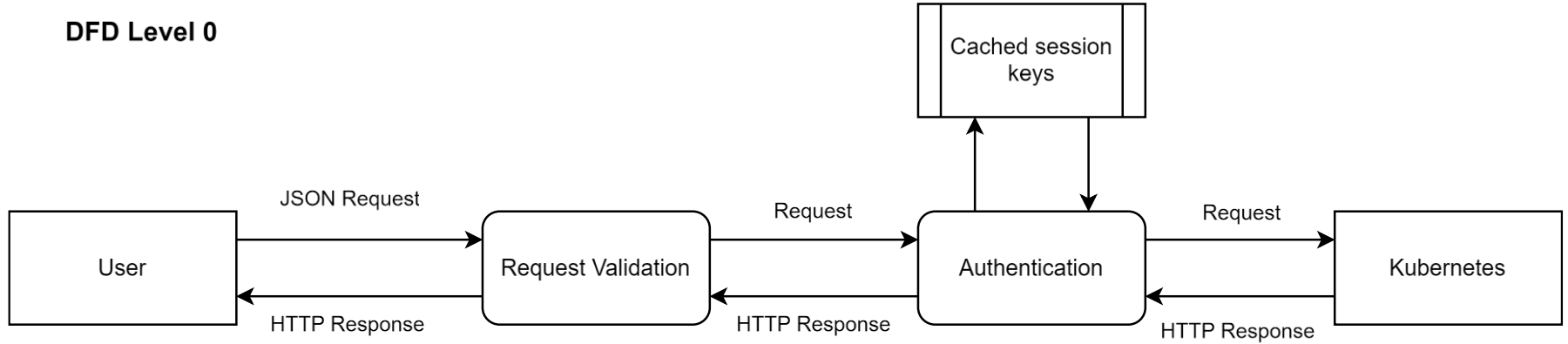
- Scalable and secure deployment of ML models.
- Integration of the API gateway into existing infrastructure.

### **Pain points:**

- Challenges with integration and corporate standards.

# DFD (Level 0)

DFD Level 0



# Story map

