**Документация, содержащая информацию, необходимую для эксплуатации экземпляра программного обеспечения**

**Модуль «Персонализация контента»**

**г. Москва 2025**

[1. Введение 3](#_un7wrqx0ex4k)

[2. Установка программы 3](#_dqbrvbxvv46i)

[2.1. Требования к окружению 3](#_qm3yxywaajr6)

[2.2. Процесс установки 3](#_ewo2njynkzzl)

[3. Начало работы 4](#_mq2rpwi42fs)

[3.1. Запуск системы 4](#_gab522y7ieuk)

[Подготовка окружения 4](#_g5taxgqk4ftw)

[3.2. Развертывание инфраструктуры в Yandex Cloud 4](#_vm34phj1l07c)

[3.3. Развертывание базы данных 5](#_z03x43lwh0vj)

[3.4. Развертывание приложения 5](#_a9abwp5pnxju)

[4. Руководство пользователя 5](#_66mbfsidzxdi)

[4.1. Вход в систему и работа с интерфейсом 5](#_x8x77stwmdn5)

[4.2. Получение персонализированных рекомендаций 5](#_nmz104829dfr)

[4.3. Управление профилем 6](#_ghsmapp6tubv)

[4.4. Работа с типами рекомендаций 6](#_9nti9zj68v8t)

[4.5. Фильтрация контента 6](#_eu0h7oahybf5)

[5. Функциональные возможности 6](#_p8x02ueyyag)

[5.1. Генерация персонализированных рекомендаций 6](#_xb8a01816wxp)

[5.2. Поддержка различных типов рекомендаций 6](#_bd3deceqguzp)

[5.3. Модели машинного обучения 6](#_bcj7mkahg0ls)

[6. Обслуживание 7](#_cpaxti8dn3dk)

[6.1. Резервное копирование 7](#_py7k9f1088n)

[6.2. Мониторинг системы 7](#_amqcvp2i0z2e)

[7. Безопасность 7](#_t1inrvo0k63e)

[8. Обновление системы 7](#_gjk3sxehqp7n)

[9. Завершение работы 7](#_xzdl1inaniw0)

[10. Заключение 7](#_hh7n6gb2qqmn)

# 1. Введение

Программное обеспечение модуль "Персонализация контента" предназначено для обработки пользовательских данных и формирования рекомендаций видеоконтента на основе предпочтений и истории просмотров. Основная цель системы – предоставление пользователям персонализированного опыта потребления медиаконтента в онлайн платформе.

Данный документ содержит информацию, необходимую для установки, настройки и эксплуатации системы. Он предназначен для технических специалистов, администраторов системы, инженеров DevOps и других экспертов, осуществляющих проверку и развертывание системы в рабочем окружении. Также в документе приведены инструкции по управлению системой, работе с API, мониторингу состояния и обеспечению безопасности.

Система основана на технологиях машинного обучения и использует алгоритмы коллаборативной и контентной фильтрации. Основная цель - формировать рекомендации, учитывая как предпочтения отдельных пользователей, так и глобальные тренды среди всех зрителей платформы.

Документ включает описание архитектуры системы, требований к программному и аппаратному обеспечению, инструкцию по установке и эксплуатации, а также руководство пользователя, описывающее основные функции системы.

## **2. Установка программы**

### 2.1. Требования к окружению

Перед установкой ПО необходимо убедиться, что система соответствует следующим требованиям:

Операционная система: Ubuntu (рекомендуется 20.04 и выше)

Виртуализация: Yandex Cloud, Docker, Kubernetes

База данных: PostgreSQL, Redis, Qdrant

Облачное хранилище: Yandex Object Storage

Программные компоненты: FastAPI, gRPC, PyTorch, CatBoost, scikit-learn

***Минимальные аппаратные требования:***

* 4 vCPU
* 8GB RAM
* 100GB SSD

***Рекомендуемые аппаратные требования:***

* 8 vCPU
* 16GB RAM
* 500GB SSD

### 2.2. Процесс установки

1. **Подготовка инфраструктуры:**
   * Развертывание облачного окружения в Yandex Cloud.
   * Настройка контейнеризации с использованием Docker и Kubernetes.
   * Конфигурация базы данных PostgreSQL и кеширования Redis.
2. **Развертывание приложения:**
   * Запуск контейнеров с использованием Helm:  
     helm install personal-recommendation ./charts
   * Подключение к облачному хранилищу для обработки данных.
3. **Настройка API и взаимодействия с внешними сервисами:**
   * Обеспечение связи с каталогами контента онлайн-кинотеатра Premier.
   * Настройка взаимодействия через API с внешними системами.

## **3. Начало работы**

### 3.1. Запуск системы

#### Подготовка окружения

**Обновите пакеты и установите зависимости:**

sudo apt update && sudo apt upgrade -y

sudo apt install -y curl wget git unzip

**Установите Docker и Docker Compose:**sudo apt install -y docker.io

sudo systemctl enable --now docker

sudo usermod -aG docker $USER  
Перезагрузите систему или выполните newgrp docker для активации прав.

**Установите Kubernetes (kubectl) и Helm:**curl -fsSL https://get.k3s.io | sh -

curl -fsSL -o /usr/local/bin/helm https://get.helm.sh/helm-v3.9.0-linux-amd64.tar.gz

chmod +x /usr/local/bin/helm

**Убедитесь, что все компоненты установлены:**docker --version

kubectl version --client

helm version

#### 3.2. Развертывание инфраструктуры в Yandex Cloud

**Авторизуйтесь в Yandex Cloud:**yc init  
Следуйте инструкциям на экране для настройки аккаунта.

**Создайте кластер Kubernetes в Yandex Cloud:**yc managed-kubernetes cluster create --name my-cluster --zone ru-central1-a --network-id default --node-group memory-optimized

**Настройте контекст Kubernetes:**yc managed-kubernetes cluster get-credentials my-cluster --external

#### 3.3. Развертывание базы данных

**Создайте PostgreSQL-базу данных в Yandex Cloud:**yc managed-postgresql cluster create --name my-db --network-id default --host zone=ru-central1-a

**Настройте подключение к базе:**export DATABASE\_URL=postgresql://user:password@host:port/dbname

#### 3.4. Развертывание приложения

**Клонируйте репозиторий приложения:**git clone https://github.com/your-repo/personalization-service.git

cd personalization-service

**Настройте переменные окружения:**cp .env.example .env

nano .env  
Укажите API-ключи, параметры базы данных и другие конфигурации.

**Запустите приложение с помощью Helm:**helm install personal-recommendation ./charts

**Проверьте статус развертывания:**kubectl get pods -n recommendation-system

**Настройте балансировщик нагрузки:**kubectl expose deployment personal-recommendation --type=LoadBalancer --port=80

**Проверьте доступность сервиса:**curl -X GET "http://your-service-ip/recommendations?user\_id=1234"

**После успешного развертывания проверьте, что API сервиса работает корректно, а также выполните тестирование всех функциональных возможностей. Система готова к эксплуатации**

### 4. Руководство пользователя

#### 4.1. Вход в систему и работа с интерфейсом

Взаимодействие с системой происходит через API. Пользовательский интерфейс предоставляет доступ к персонализированным рекомендациям и настройке предпочтений.

#### 4.2. Получение персонализированных рекомендаций

Для получения рекомендаций пользователь должен войти в систему с использованием PassMedia ID.

Рекомендации доступны через API-запрос:

curl -X GET "https://api.premier-recsys.com/recommendations?user\_id=1234"

В ответ система вернет список подходящего контента.

#### 4.3. Управление профилем

Пользователь может изменить свои предпочтения для улучшения рекомендаций:

curl -X POST "https://api.premier-recsys.com/profile" -d '{"user\_id": "1234", "preferences": {"genre": "comedy"}}'

Возможные настройки профиля включают жанры, актеров, языки контента.

#### 4.4. Работа с типами рекомендаций

**Персонализированные рекомендации** – генерируются на основе истории просмотров пользователя.

**Трендовые рекомендации** – основываются на общем поведении пользователей.

**Жанровые подборки** – учитывают предпочтения пользователя по жанрам.

#### 4.5. Фильтрация контента

Система поддерживает фильтры, позволяющие исключать нежелательный контент по возрастным рейтингам, регионам или жанрам.

Для применения фильтрации необходимо передать параметры в API-запросе.

## **5. Функциональные возможности**

### 5.1. Генерация персонализированных рекомендаций

Система формирует индивидуальные подборки контента на основе истории просмотров и предпочтений пользователей.

### 5.2. Поддержка различных типов рекомендаций

Персонализированные рекомендации – на основе истории пользователя.

Общие трендовые рекомендации – на основе популярности контента среди всех пользователей.

### 5.3. Модели машинного обучения

Коллаборативная фильтрация – анализ предпочтений схожих пользователей.

Контентная фильтрация – анализ характеристик контента для подбора релевантных рекомендаций.

## **6. Обслуживание**

### 6.1. Резервное копирование

Автоматическое резервное копирование базы данных раз в сутки. Возможность восстановления данных в течение 1 часа.

### 6.2. Мониторинг системы

**Метрики производительности:**

Среднее время ответа API – не более 200 мс. Поддержка нагрузки до 1500 rps.

**Инструменты мониторинга:**

Prometheus – сбор метрик производительности.

Grafana – визуализация данных.

Loki – логирование и анализ ошибок.

## **7. Безопасность**

Аутентификация и авторизация пользователей через PassMedia ID. Защита API от SQL-инъекций и XSS-атак. Шифрование данных при передаче и хранении.

## **8. Обновление системы**

Обновления ПО разворачиваются через CI/CD. Обновления моделей машинного обучения выполняются по заранее определенному графику.

## **9. Завершение работы**

1. **Остановка обработки запросов API:**kubectl delete -f deployment.yaml
2. **Остановка контейнеров:**docker-compose down
3. **Выключение облачной инфраструктуры (при необходимости).**

## **10. Заключение**

Программное обеспечение "Персонализация контента" обеспечивает обработку запросов, расчет персонализированных рекомендаций и их передачу через API. Использование облачной инфраструктуры и машинного обучения позволяет достигать высокой точности рекомендаций, масштабируемости и надежности работы системы.