

## 1. Назначение и условия применения программы

### 1.1 Назначение

Документ предназначен для программистов, занимающихся разработкой, сопровождением и тестированием рекомендательной системы в финансовом секторе.

Программа реализует:

- Прокси-сервер на базе Nginx с логированием в JSON.
- Рекомендательный сервис на Flask с безопасным поиском через ORM.
- Хранение данных в MySQL.
- Возможность интеграции с Fail2ban для защиты от DDoS (не настроен).

### 1.2 Условия применения

Для работы системы необходимы:

- Операционная система : Linux (Ubuntu 20.04+, Alpine), Windows (Docker Desktop), macOS.
- Docker Engine : версия 20+
- Docker Compose : версия 2.0+
- Python : 3.11+ (для Flask).
- MySQL : 8.0+
- Fail2ban : 1.1.0+

## 2. Характеристика программы

### 2.1 Технические средства

Система разработана с использованием:

- Docker : контейнеризация, изолированные сети
- Nginx : прокси, логирование в JSON, шифрование TLS.
- Flask : бэкэнд рекомендаций, безопасный поиск через SQLAlchemy.
- MySQL : хранение данных, инициализация через **init.sql**.
- Fail2ban : планируется интеграция для защиты от DDoS (не настроен).

### 2.2 Ограничения и особенности

- Fail2ban не поддерживает JSON-логи напрямую → требуется парсер (например, **jq**) или переход на строковый формат логов Nginx.
- JSON-логи Nginx : подходят для анализа в ELK, Grafana, но требуют обработки.
- Fail2ban не блокирует IP → структура позволяет настроить позже.
- SQL-инъекции и XSS : предотвращаются через ORM и логирование

## 3. Обращение к программе

### 3.1 Запуск системы

# Перейдите в папку проекта

```
cd ~/recommendation_system
```

# Соберите и запустите контейнеры

```
docker-compose up -d --build
```

### 3.2 Проверка состояния

# Проверьте, все ли контейнеры запущены

```
docker ps
```

# Проверьте логи Nginx

```
docker logs recommendation_system_proxy_1
```

# Проверьте данные в MySQL

```
docker exec -it db mysql -u root -padmin -e "SELECT * FROM  
recommendation_db.recommendations"
```

## 4. Входные и выходные данные

### 4.1 Входные данные

- Пользовательский запрос GET /search?q=... → параметр **q** для поиска рекомендаций
- IP-адрес клиента - Для мониторинга и блокировки через Fail2ban (не настроен)
- MySQL - Данные из таблицы recommendations

### 4.2 Выходные данные

- HTML-интерфейс - Карточки рекомендаций в формате Bootstrap
- JSON-логи Nginx - Строка вида: { "@timestamp": "2025-06-05T10:00:00+00:00",  
"remote\_addr": "172.21.0.1", "request": "GET /search?q=Card HTTP/1.1", "status": 200 }

## 5. Сообщения

### 5.1 Типовые сообщения системы

- 403 Forbidden - IP-адрес заблокирован за частые запросы (если Fail2ban настроен)
- curl: (6) Could not resolve host: localhost – Контейнер proxy не запущен
- docker: Error response from daemon: Conflict. - Конфликт портов → перезапустите систему

### 5.2 Логи

- Nginx : **./nginx-proxy/logs/access.log**
- Flask : **docker logs recommendation\_system\_recommendation\_system\_1**

- MySQL : **docker logs recommendation\_system\_db\_1**
- Fail2ban : **docker logs recommendation\_system\_fail2ban\_1**