

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение высшего профессионального образования  
«Южно-Уральский государственный университет (национальный исследовательский университет)»  
Высшая школа электроники и компьютерных наук  
Кафедра системного программирования

# РАЗРАБОТКА СЕРВИСА ДЛЯ ГЕНЕРАЦИИ ИЗОБРАЖЕНИЙ ФАСАДОВ ЗДАНИЙ ПО ВЫДЕЛЕННОЙ МАСКЕ

Рецензент:

доцент кафедры ИАОУ «ЮУрГУ», к.т.н.  
А.А. Шинкарев

Научный руководитель:

доцент кафедры СП, к.т.н.  
М.В. Сухов

Автор:

студент группы КЭ-228  
М.А. Сипатов

Челябинск, 2024 г.

# АКТУАЛЬНОСТЬ

- Ускорение работы дизайнеров
- Улучшение архитектурного стиля города
- Очищение зданий от рекламных вывесок



# ЦЕЛЬ И ЗАДАЧИ ИССЛЕДОВАНИЯ

**Цель:** Разработка сервиса для генерации изображений фасадов зданий по выделенной маске

**Задачи:**

1. Провести анализ предметной области
2. Провести обзор научной литературы
3. Выполнить проектирование архитектуры приложения
4. Разработать инфраструктуру для приложения
5. Реализовать приложение
6. Протестировать приложение



# АНАЛИЗ ЛИТЕРАТУРЫ

Название	Набор данных	Модель	Метрика
<b>Denoising Diffusion Probabilistic Models</b>	CIFAR10	DDPM	FID: 9,46
<b>A Style-Based Generator Architecture for Generative Adversarial Networks</b>	CelebA-HQ	StyleGAN	FID: 5,17
<b>High-Resolution Image Synthesis with Latent Diffusion Models</b>	CelebA-HQ	LD	FID: 5,11



# СХЕМА ОБРАБОТКИ ИЗОБРАЖЕНИЙ

Текстовый  
запрос

single color, no text, neat  
facade, background, merges  
with the background, less  
boundaries

Исходное изображение



Выделение  
маски

Генерация  
изображения

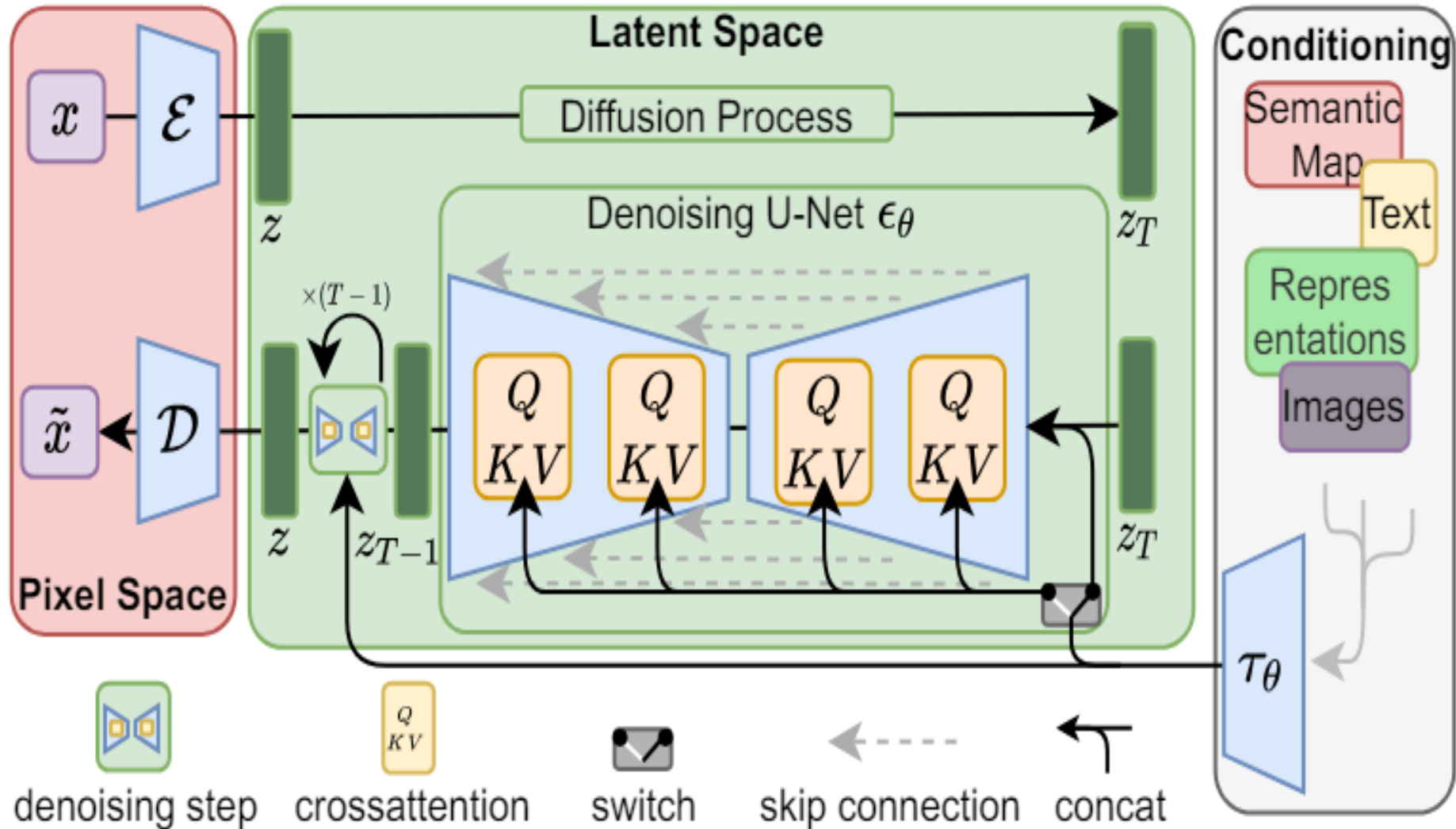
Обработанное изображение



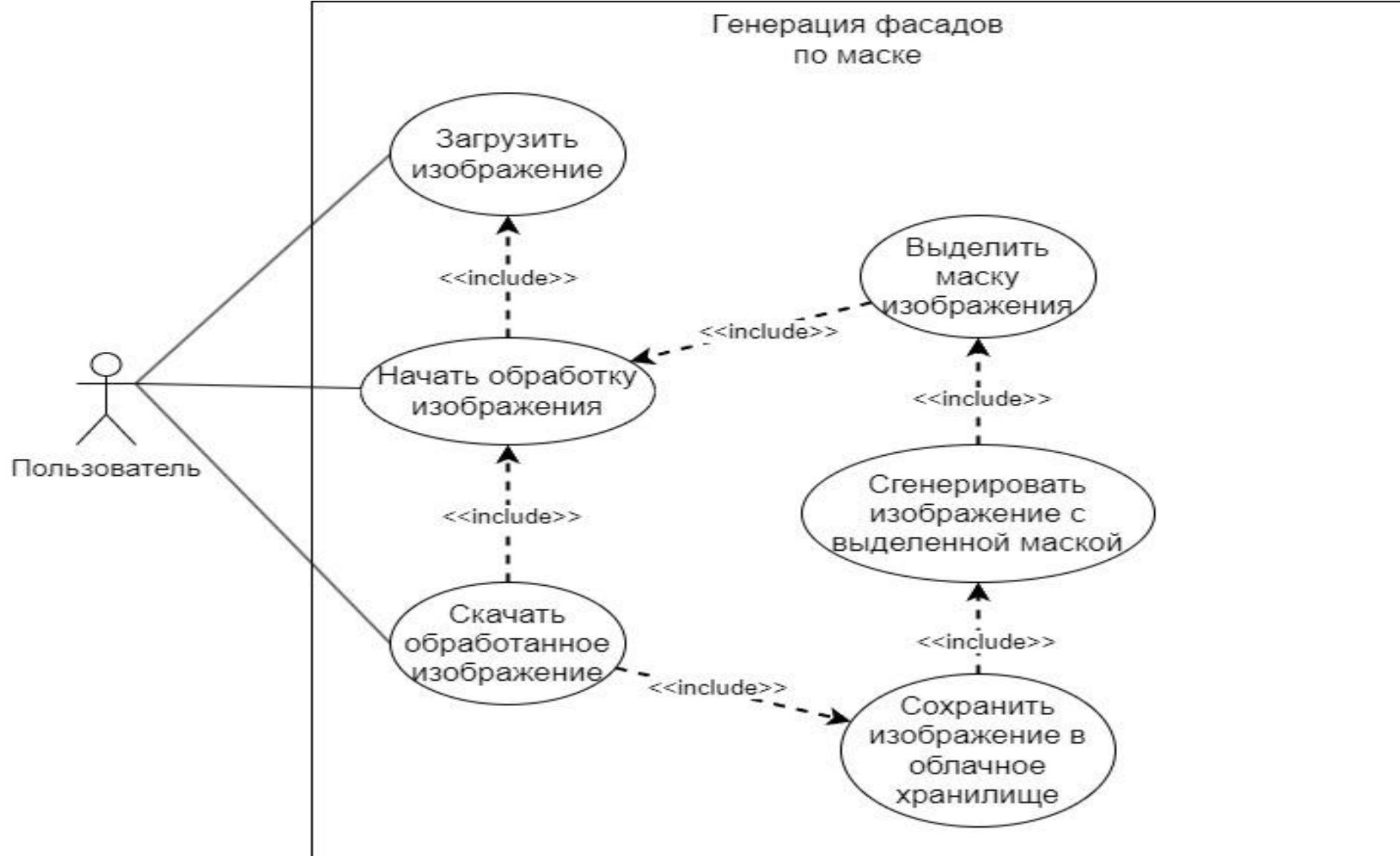
Маска изображения



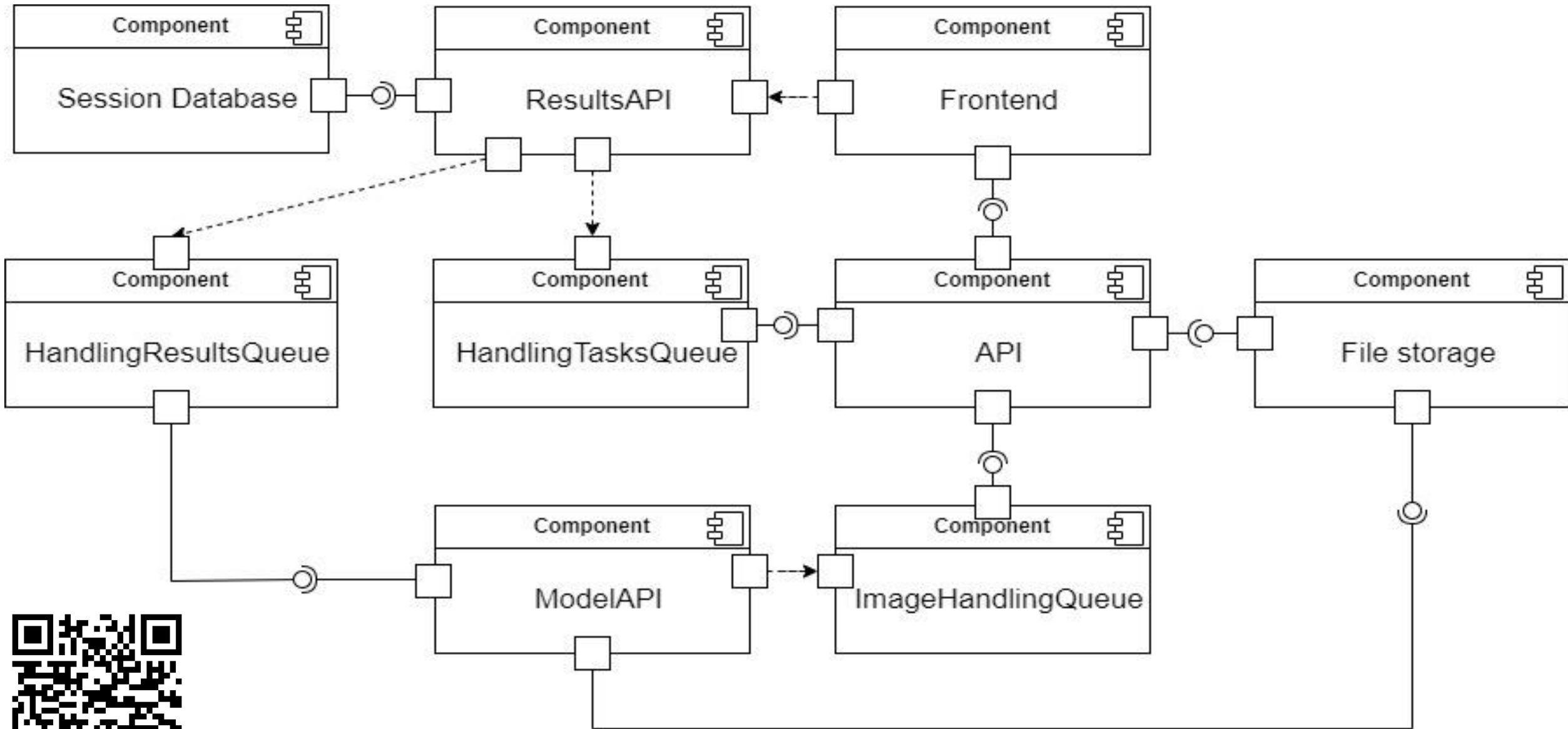
# АРХИТЕКТУРА МОДЕЛИ ДЛЯ ГЕНЕРАЦИИ ИЗОБРАЖЕНИЙ



# ДИАГРАММА ВАРИАНТОВ ИСПОЛЬЗОВАНИЯ



# ДИАГРАММА КОМПОНЕНТОВ





# СРЕДСТВА РАЗРАБОТКИ ИНФРАСТРУКТУРЫ

## Операционная система:

Ubuntu 22.04

## Пакеты:

opennebula 6.4.0

nginx 1.25.5

kubetl 1.29.1

helm 3.14.1

Docker 25.0.3

## Характеристики сервера:

ОЗУ: 32 Гб

ЦПУ: AMD Ryzen 5 7500F

Графический ускоритель:

Nvidia Geforce RTX 2060

Видеопамять – 6 гб

Постоянная память – 1 тб



# ПАЙПЛАЙНЫ ДЛЯ НЕПРЕРЫВНОЙ ДОСТАВКИ И РАЗВЕРТЫВАНИЯ ПРИЛОЖЕНИЯ

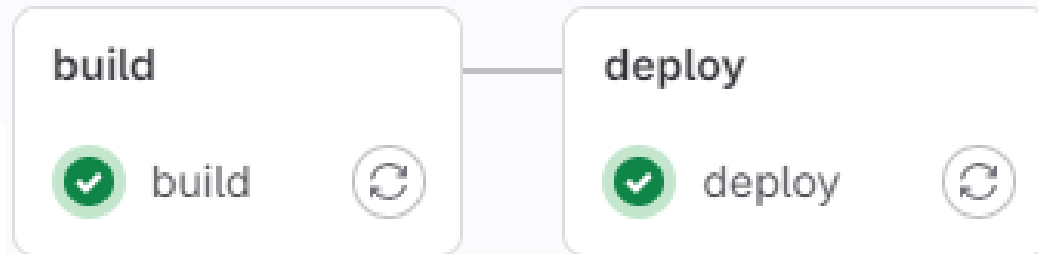
## ci: change ci to have cache build

✓ Passed MaxBitWise created pipeline for commit `f45700b9` 2 days ago, finished 2 days ago

For `develop`

2 jobs 38 seconds, queued for 1 seconds

Pipeline Needs Jobs 2 Tests 0



# ОКРУЖЕНИЕ ДЛЯ РАЗВЕРТЫВАНИЯ ПРИЛОЖЕНИЯ



# СРЕДСТВА РАЗРАБОТКИ ПРИЛОЖЕНИЯ

## Языки программирования:

Python 3.8

JavaScript ES 6

C# 12

## Фреймворки:

React 18

## Среда разработки:

PyCharm Community

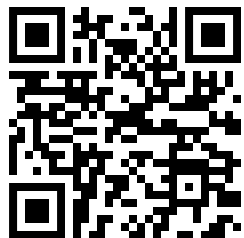
Visual Studio

Visual Studio Code



# REST API

Метод	URL	Описание
<b>GET</b>	/api/s3/download	Скачивание обработанного изображения
<b>GET</b>	/api/image-handling-results	Получение статуса обработки изображения
<b>POST</b>	/api/s3/upload	Загрузка изображения для обработки
<b>GET</b>	/api/gallery	Получение изображений для галереи



# ФОРМА ЗАГРУЗКИ ИЗОБРАЖЕНИЯ

*City beauty*

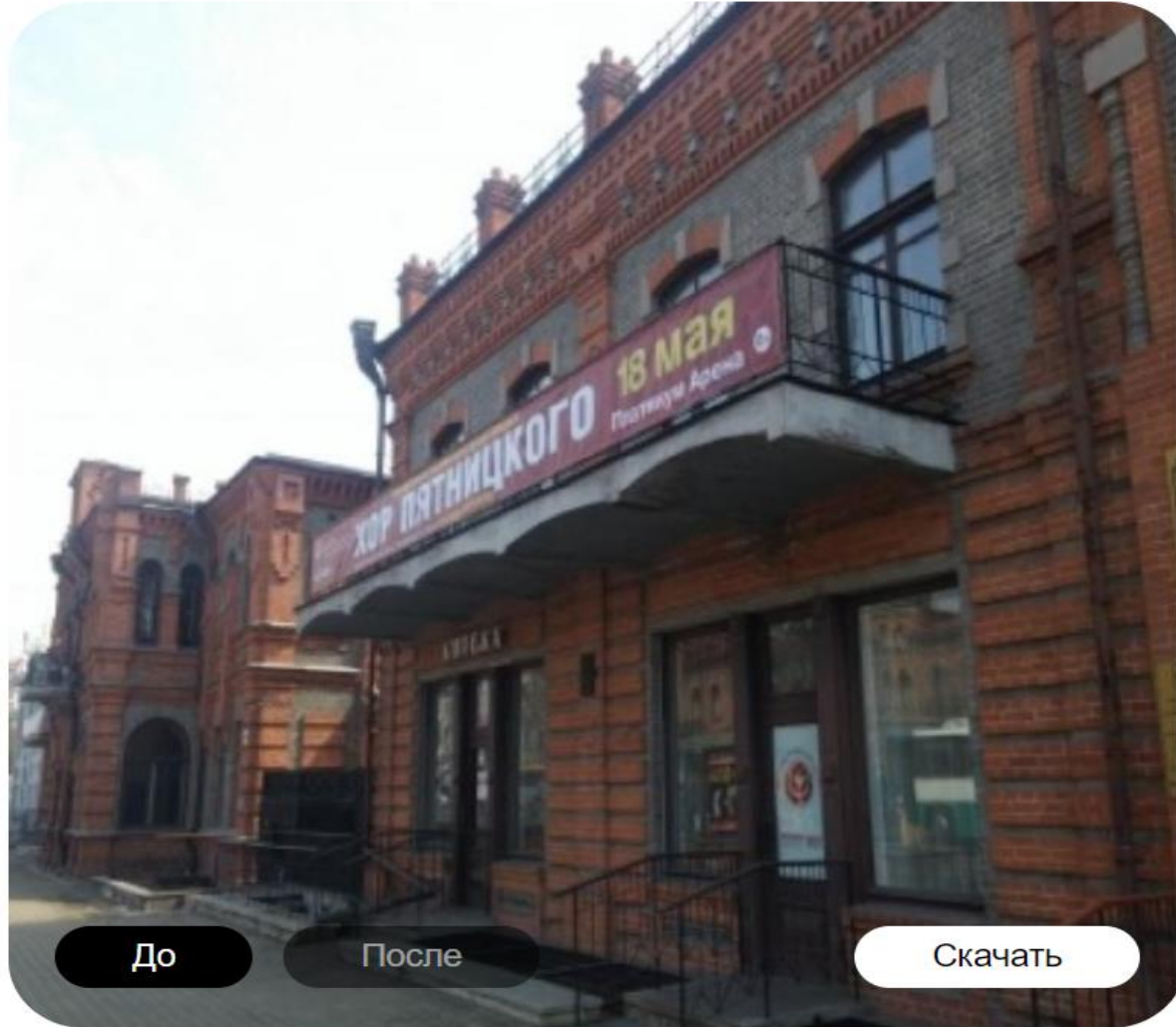


Выберите файл или  
перетащите его сюда

Обработать



# ФОРМА ИЗОБРАЖЕНИЯ ДО ОБРАБОТКИ



Добавить новое фото

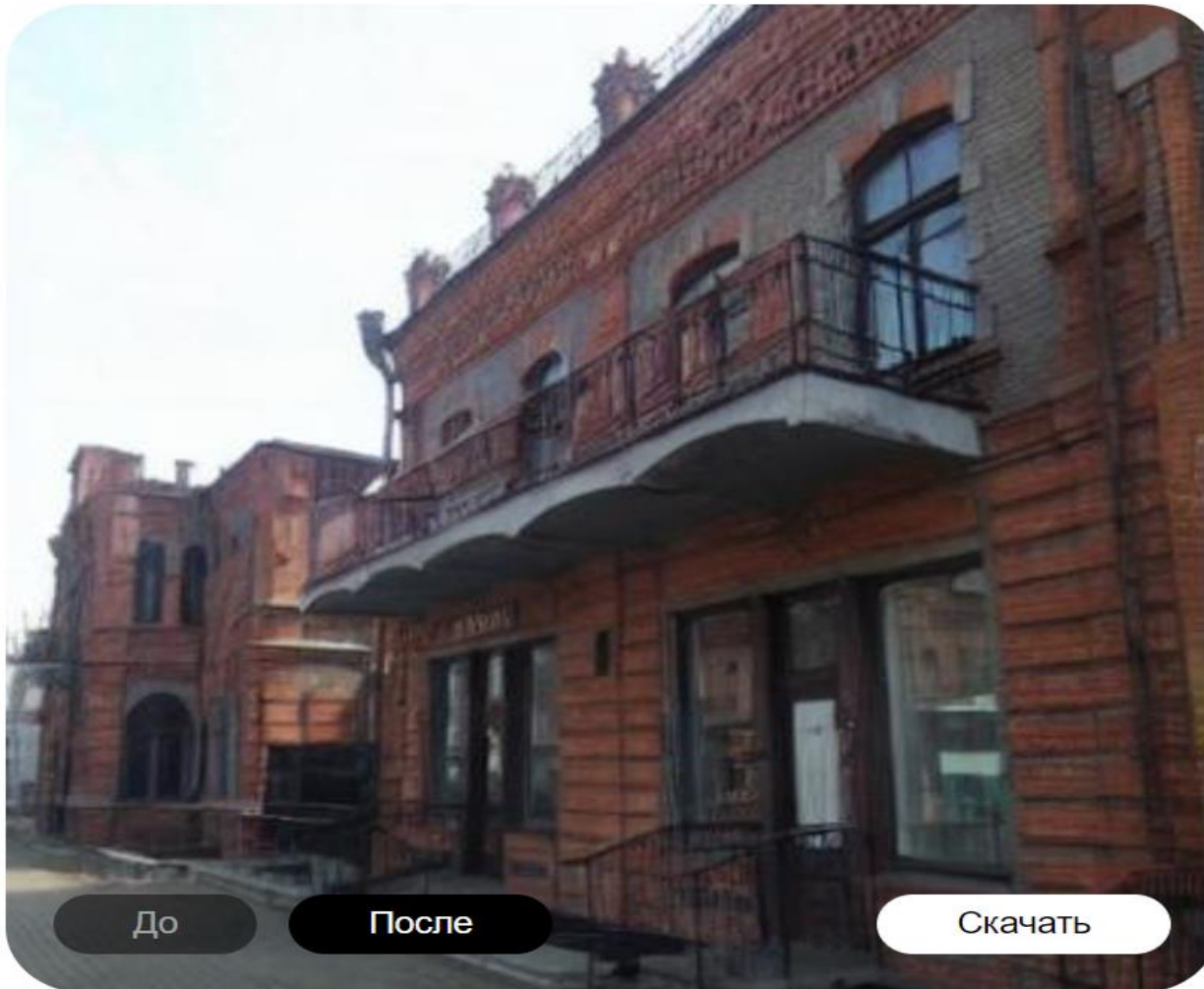
До

После

Скачать



# ФОРМА ИЗОБРАЖЕНИЯ ПОСЛЕ ОБРАБОТКИ

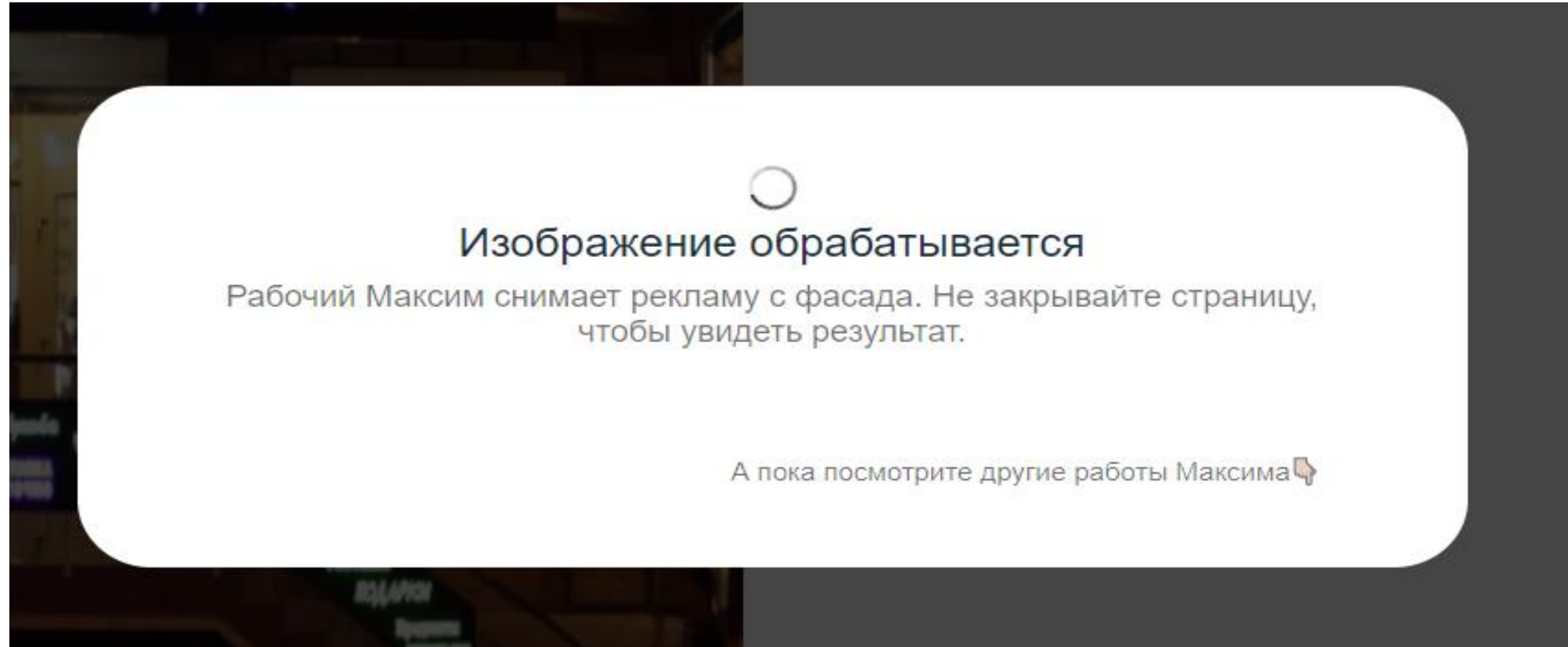


Добавить новое фото





# ВСПЛЫВАЮЩЕЕ ОКНО ОЖИДАНИЯ ОТВЕТА



# ПРИМЕРЫ СГЕНЕРИРОВАННЫХ ИЗОБРАЖЕНИЙ



# ФУНКЦИОНАЛЬНОЕ ТЕСТИРОВАНИЕ ПРИЛОЖЕНИЯ

Тестирование основных функций приложения:

- Загрузка изображений поддерживаемых форматов
- Загрузка изображений размером до 100 Мб
- Уведомление об ошибках обработки изображения
- Уведомление о невозможности загрузки файлов неподдерживаемых форматов
- Скачивание обработанного изображения



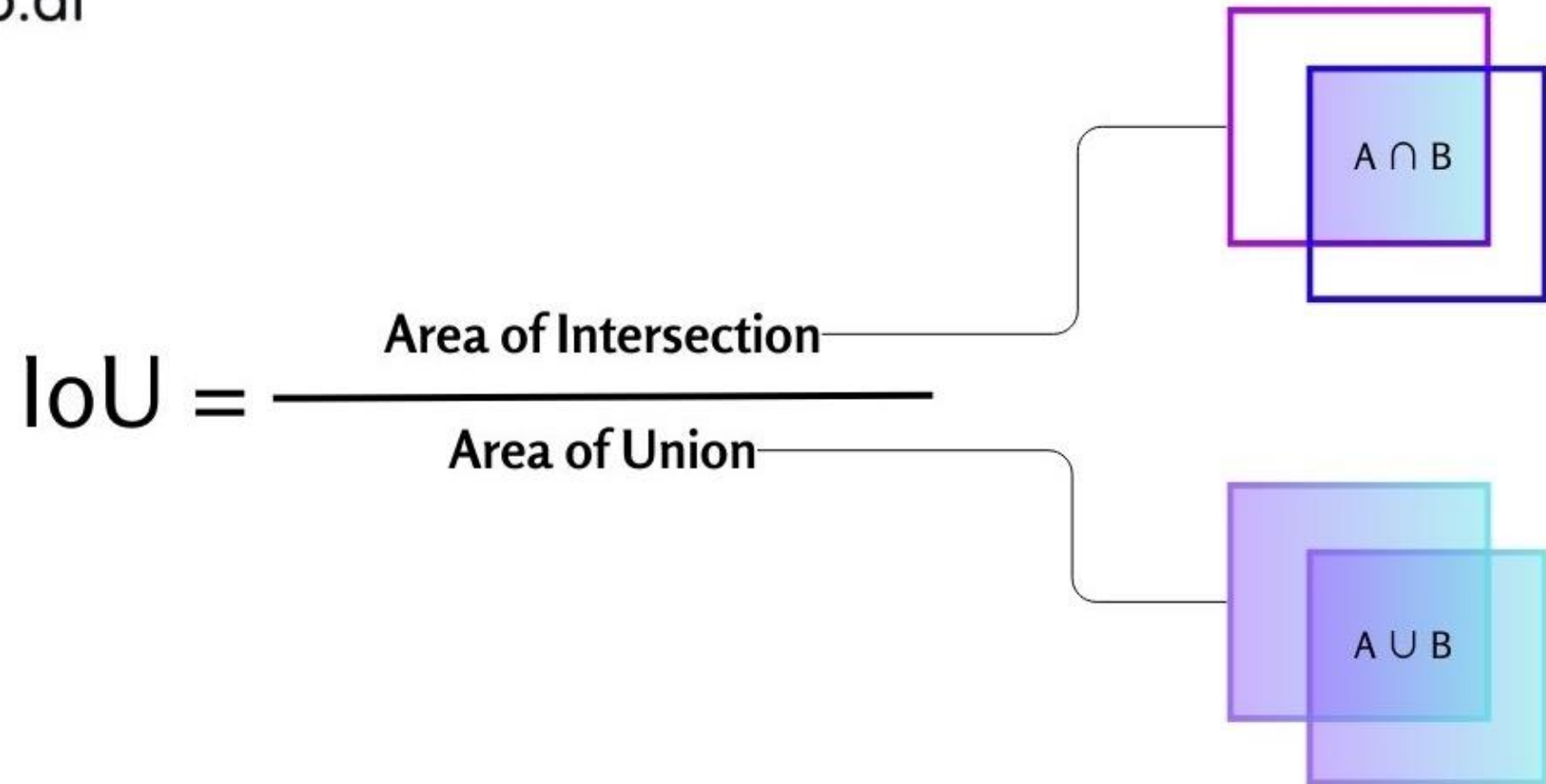
# ОСНОВНЫЕ РЕЗУЛЬТАТЫ

1. Проведен анализ предметной области
2. Проведен обзор научной литературы
3. Спроектирована архитектура приложения
4. Разработана инфраструктура для приложения
5. Разработано приложение для генерации изображений с использованием диффузионных нейронных сетей
6. Проведено тестирование приложения



# МЕТРИКА mIoU (ДОПОЛНИТЕЛЬНЫЙ СЛАЙД)

VISO.ai



**МЕТРИКА FID  
(ДОПОЛНИТЕЛЬНЫЙ СЛАЙД)**

$$FID = \| m - m_w \|^2 + \text{Tr}(C + C_w - 2(C C_w)^{\frac{1}{2}})$$

# СХЕМА БАЗЫ ДАННЫХ

HandlingResults
SessionId: text
BucketSourceImageName: text
Status: text
BucketHandledImageName: text
HandlingFinishedAtUtc: timestamp
StartedHandlingAtUtc: timestamp



# АРХИТЕКТУРА МОДЕЛИ PSPNet ДЛЯ ВЫДЕЛЕНИЯ МАСКИ (ДОПОЛНИТЕЛЬНЫЙ СЛАЙД)

