|  |  |
| --- | --- |
| СОГЛАСОВАНО | УТВЕРЖДАЮ |
| Сторона ЗАКАЗЧИКА  Доцент кафедры  ИАНИ ННГУ, к.т.н.  С.М. Липкин  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_2023 г. | Сторона ИСПОЛНИТЕЛЯ  Профессор кафедры  ИАНИ ННГУ, д.т.н.  Н.В. Старостин  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_2023 г. |

**СИСТЕМА АНАЛИЗА КОРРОЗИОННЫХ ПОВРЕЖДЕНИЙ ТРУБОПРОВОДОВ С ИСПОЛЬЗОВАНИЕМ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ**

**Руководство системного программиста**

**Этап 3. Подготовка программной документации**

**НИР «Разработка и реализация программного обеспечения анализа коррозионных повреждений фольги   
с использованием методов машинного обучения»**

**(Шифр ПО «Око Эйлера 2»)**

Ответственные исполнители

Живаев А. Е.

Архипова А. А.

Васильев А. С.

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г.

2023 г.

# АННОТАЦИЯ

#### Руководство системного программиста представляет собой информацию по установке и настройке программы анализа дефектов поверхности фольги (ПО «Око Эйлера 2»).

Руководство предназначено для системных программистов, отвечающих за функционирование программы.

**СОДЕРЖАНИЕ**

1. [ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММЕ](#_30j0zll) **4**
2. [СТРУКТУРА ПРОГРАММЫ](#_5rvcyvuy9pe9) **5**
3. [НАСТРОЙКА ПРОГРАММЫ](#_1t3h5sf) **7**
4. [ПРОВЕРКА ПРОГРАММЫ](#_35nkun2) **8**
5. [СООБЩЕНИЯ СИСТЕМНОМУ ПРОГРАММИСТУ](#_2x6p641sfn7t) **17**

# **ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММЕ**

## Функции программы

ПО «Око Эйлера 2» выполняет следующие функции:

* осуществляет чтение исходных данных в виде графического файла формата jpg;
* осуществляет проверку исходных данных на соответствие формату;
* строит маски обнаруженных дефектов;
* предоставляет возможность пользователю ввести корректировки для масок соответствующих дефектов для своего изображения;
* предоставляет возможность просмотра результирующего изображения;

## Обеспечивающие технические и программные средства

Для функционирования ПО «Око Эйлера 2» необходимы следующие технические и программные средства

на стороне сервера:

* ЭВМ на базе процессора с PR-рейтингом не менее 2000, оперативной памятью не менее 4ГБ, доступной дисковой памятью в размере не менее 100ГБ, подключение к сети Интернет с пропускной способностью не меньше 100 Мбит/сек;
* программное обеспечение:
* операционная система Windows не ниже Windows 10
* Python 3.9.2 с установленными модулями:
  + Flask 2.0.1
  + NumPy 1.21.2
  + Pandas 1.3.3
  + Pillow 8.3.2
  + PyTorch 1.9.1
  + Torchvision 0.10.1

для клиентской стороны:

* ЭВМ на базе процессора с PR-рейтингом не менее 2000, оперативной памятью не менее 4ГБ, доступной дисковой памятью в размере не менее 32ГБ, монитором, клавиатурой и мышью; (ПК или мобильное устройство, подключенное к сети Интернет; браузер, поддерживающий HTML5)
* операционная система Windows 10 с установленным браузером Google Chrome версии 97.0.4692.99 и выше.

# **СТРУКТУРА ПРОГРАММЫ**

В Таблице 1 отражена файловая структура программы и ее соотношения с функциональными элементами (подсистемами) ПО “Око Эйлера 2”

Таблица 1

|  |  |
| --- | --- |
| [auth.py](https://github.com/EulerDL/def1/blob/main/auth.py) app.py [db.py](https://github.com/EulerDL/def1/blob/main/db.py) [blog.py](https://github.com/EulerDL/def1/blob/main/blog.py) [schema.sql](https://github.com/EulerDL/def1/blob/main/schema.sql) static/css/style.css static/scripts/script.js templates/auth/[login.html](https://github.com/EulerDL/def1/blob/main/templates/auth/login.html) templates/auth/register[.html](https://github.com/EulerDL/def1/blob/main/templates/auth/login.html) templates/blog/[create.html](https://github.com/EulerDL/def1/blob/main/templates/blog/create.html) templates/blog/feedback[.html](https://github.com/EulerDL/def1/blob/main/templates/blog/create.html) templates/blog/index[.html](https://github.com/EulerDL/def1/blob/main/templates/blog/create.html) templates/blog/show[.html](https://github.com/EulerDL/def1/blob/main/templates/blog/create.html) templates/blog/update[.html](https://github.com/EulerDL/def1/blob/main/templates/blog/create.html) templates/base.html | Подсистема взаимодействия с пользователем |
| new\_dict\_new\_30.pth  Data.py  Simple\_Unet.py  Training\_functions.py  Training\_script.py  benchmark.py | Подсистема сегментации |
| Dataset generator/dataset\_generator.py | Подсистема генерации обучающих данных |
| Gradient\_Boosting.py  Decision\_Tree.py  gradient\_weight.pkl | Подсистема принятия решений |

# Таблица 1. Структура программы

# **НАСТРОЙКА ПРОГРАММЫ**

## Установка зависимостей

## Перед установкой программы на сервер необходимо установить программное обеспечение, описанное в пункте 1.3 настоящего руководства, в соответствии с их инструкциями по установке:

* Установить python версии не менее 3.7, но менее 3.10 с официального сайта <https://www.python.org/downloads/>
* Убедиться, что расположение интерпретатора python добавлено в системные пути.

## Установка программы

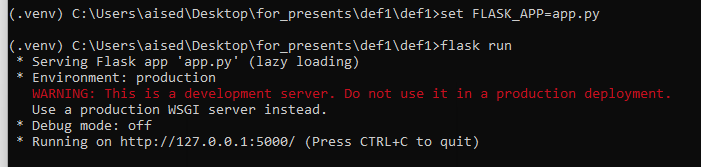
* Создать папку для репозитория, полный адрес установки может состоять из символов латиницы и знака “\_”.
* Перейти в папку.
* Склонировать в неё репозиторий <https://github.com/ZhivaevArtem/pipe-corrosion-detection> .
* Зайти в терминал (командную строку)
* cd def1
* Создать окружение  
  python -m venv .venv
* Активировать окружение:  
  .venv\Scripts\activate
* Установить библиотеки  
  pip install -r requirements.txt
* Инициализировать базу данных  
  flask init-db



## Запуск сервера

## set FLASK\_APP=app.py - убеждаемся, что app.py установлен в качестве исполняемого файла при запущенном окружении .venv\Scripts\activate

## flask run - запуск приложения



# 

# **ПРОВЕРКА ПРОГРАММЫ**

Программа проверяется в соответствии с ПМИ «Око Эйлера 2»