



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«МИРЭА - Российский технологический университет»

РТУ МИРЭА

Институт информационных технологий (ИТ)

Кафедра инструментального и прикладного программного обеспечения (ИиППО)

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА

На тему: «Интерактивное веб-приложение для анализа и визуализации данных с использованием алгоритмов машинного обучения».

Выполнил студент группы ИКБО-20-19 Миронов Дмитрий Сергеевич

Руководитель ВКР: к.т.н, доцент С.Б. Плотников

Консультант ВКР: старший преподаватель И.В. Белоусова



Цели и задачи выпускной квалификационной работы

Цель: проектирование и реализация интерактивного веб-приложения для анализа и визуализации данных с использованием алгоритмов машинного обучения. Проектирование архитектуры, реализация алгоритмов машинного обучения, разработка спроектированного решения для клиентской и серверной стороны, разработка базы данных.

Задачи:

1. Анализ существующих веб-приложений для анализа и визуализации данных, методов визуализации и алгоритмов машинного обучения.
2. Выбор и обоснование средств разработки веб-приложения, методов визуализации и алгоритмов машинного обучения.
3. Разработка архитектуры системы.
4. Разработка бизнес-логики приложения.
5. Тестирование приложения.
6. Планирование работ по теме и расчет полной стоимости проведения работ.



Актуальность и новизна выпускной квалификационной работы

Актуальность: данное веб-приложение поможет быстро анализировать большие наборы данных, выявлять закономерности и генерировать идеи, которые могут помочь компаниям принимать решения на основе данных.

Новизна:

1. Интеграции алгоритмов машинного обучения.
2. Анализ и визуализация в реальном времени.
3. Адаптированный интерфейс с учетом новизны и актуальности.



Анализ существующих веб-приложений для анализа и визуализации

Рассмотрим несколько веб-приложений

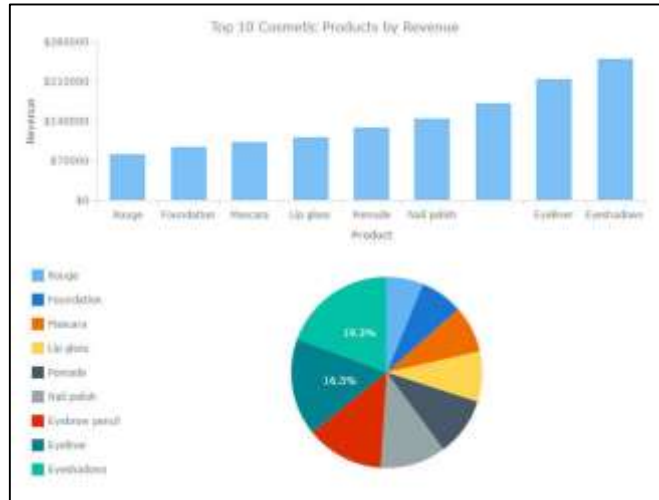


Рисунок 1 – Пример веб-приложения Ember Charts



Рисунок 2 – Пример веб-приложения ChartBlocks

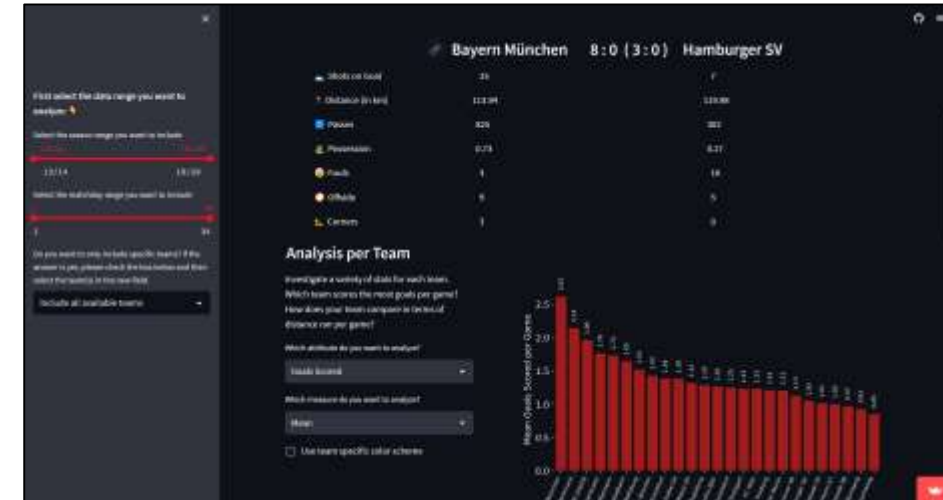


Рисунок 3 – Пример веб-приложения BuLiAn



Анализ методов визуализации

Выделим основные

методы визуализации данных:

1. Стандартные 2D/3D-образы (рисунок 4).
2. Отображение иконок.
3. Методы ориентированные на пиксели.



Рисунок 4 – Примеры визуализации



Анализ методов машинного обучения

Методы машинного обучения делятся на обучение с учителем и без учителя. Выделим основные методы машинного обучения:

1. Регрессия (рисунок 5).
2. Классификация (рисунок 6).
3. Кластеризация.

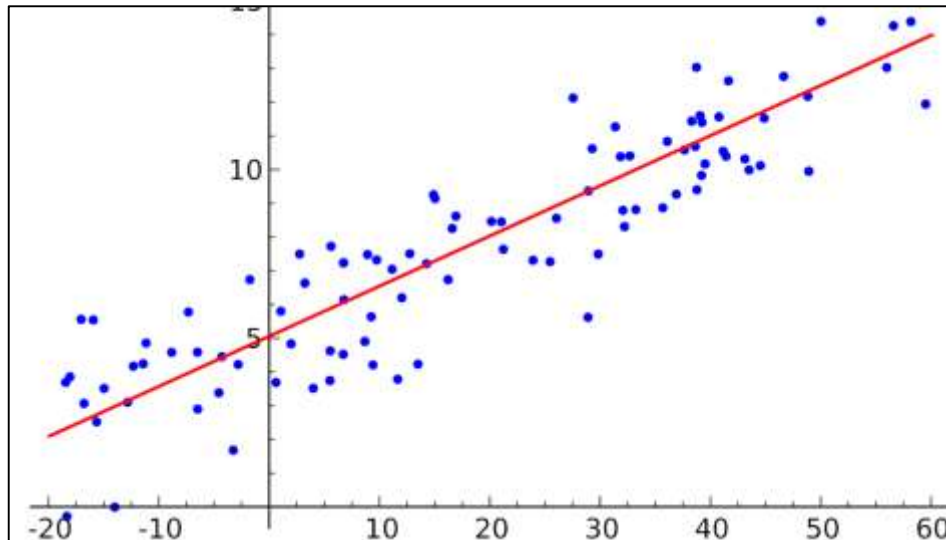


Рисунок 5 – Примеры регрессии

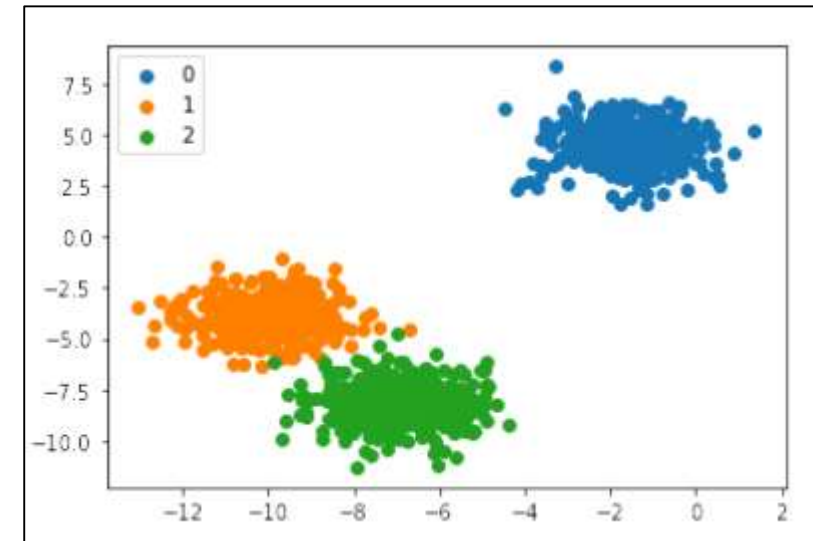


Рисунок 6 – Примеры классификации



Выбор средств разработки



Рисунок 7 – Python



Рисунок 8 – Streamlit



Рисунок 9 – база данных PostgreSQL



Проектирование адаптированной модели жизненного цикла

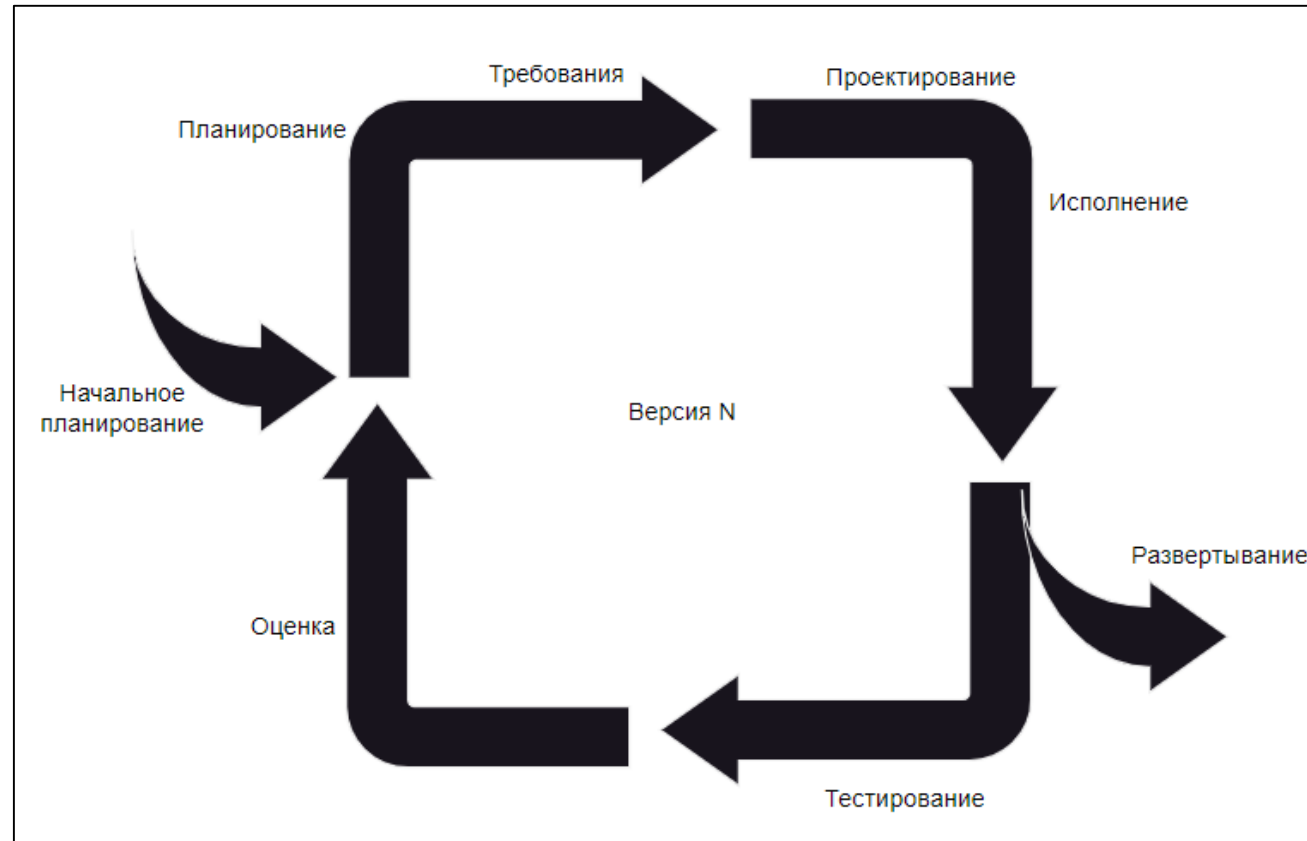


Рисунок 10 – Итерационная модель
жизненного цикла



Разработка структурной схемы веб-приложения

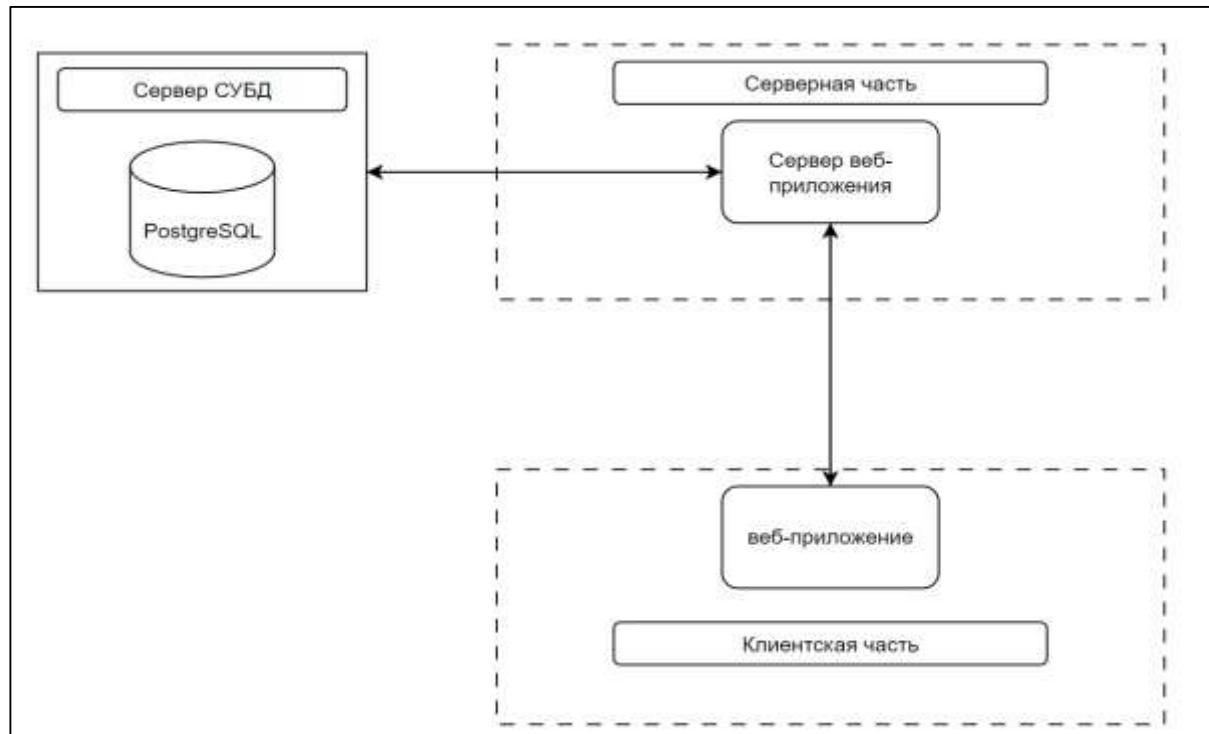
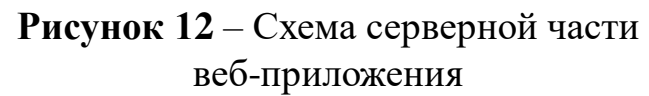


Рисунок 11 – Структурная схема приложения





Разработка серверной части веб-приложения

```
server {
    listen 443 ssl;
    server_name example.com;
    ssl_certificate
/etc/letsencrypt/live/example.com/fullchain.pem;
    ssl_certificate_key
/etc/letsencrypt/live/example.com/privkey.pem;
    location / {
        proxy_pass http://localhost:8888;
    }
    proxy_http_version 1.1;
    proxy_set_header Upgrade $http_upgrade;
    proxy_set_header Connection $connection_upgrade;
    proxy_set_header Host $host; }
```

Рисунок 13 – Использование веб-сокетов и защита трафика SSL

```
import tornado.ioloop
import tornado.web
import tornado.websocket

class MainHandler(tornado.web.RequestHandler):
    def get(self):
        self.get_apps()

class WebSocketHandler(tornado.websocket.WebSocketHandler):
    def open(self):
        self.open_apps()
    def on_message(self, message):
        self.on_mess(message)
    def on_close(self):
        self.close_apps()

def make_app():
    return tornado.web.Application([
        (r"/", MainHandler),
        (r"/ws", WebSocketHandler),
    ])

if __name__ == "__main__":
    app = make_app()
    app.listen(8888)
    tornado.ioloop.IOLoop.current().start()
```

Рисунок 14 – Обработчик запросов

Разработка архитектуры клиентской части веб-приложения



Рисунок 15 – Схема клиентской части приложения



Разработка клиентской части веб-приложения

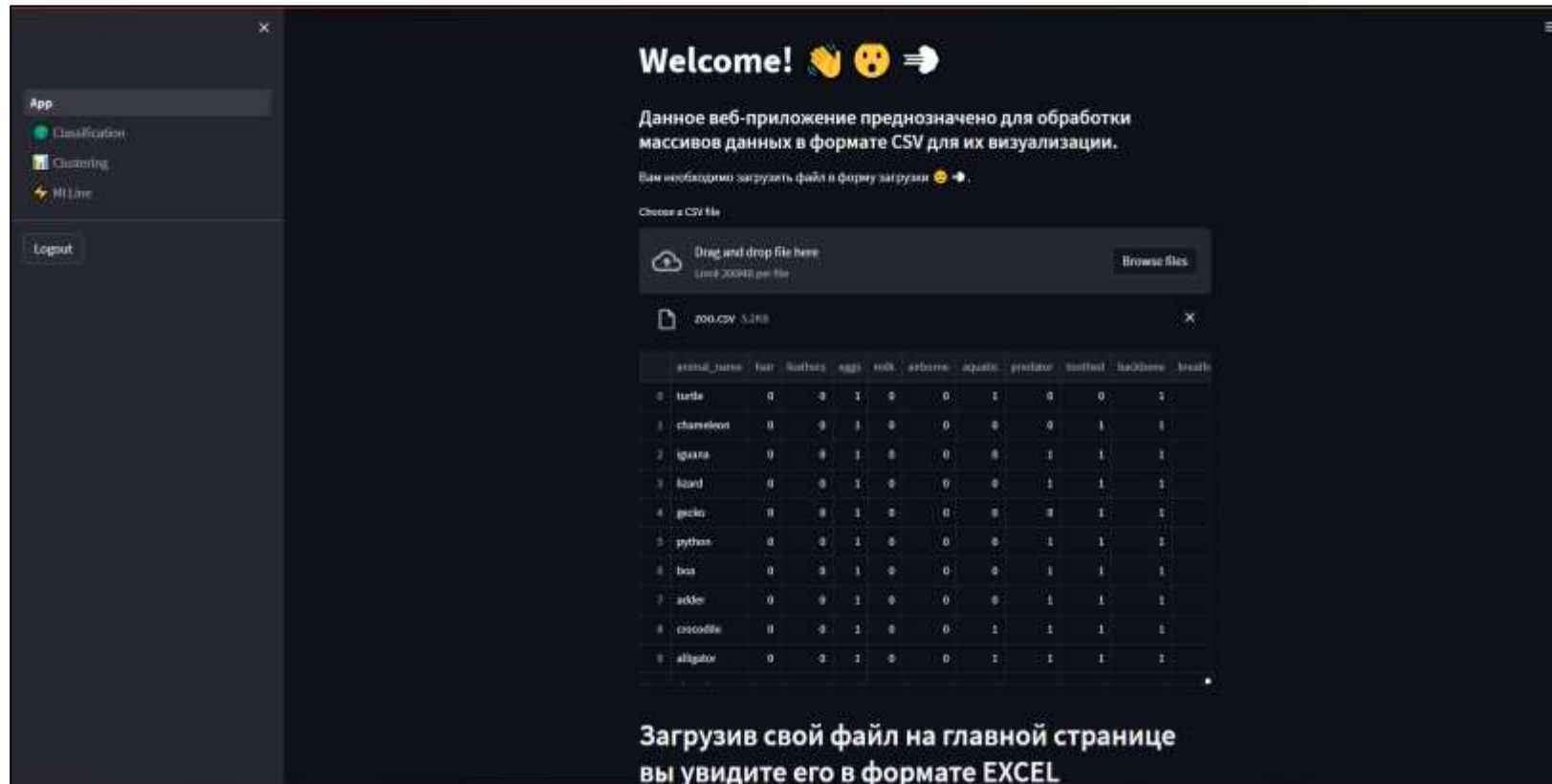


Рисунок 16 – Главная страница с загруженным массивом данных

Регрессия

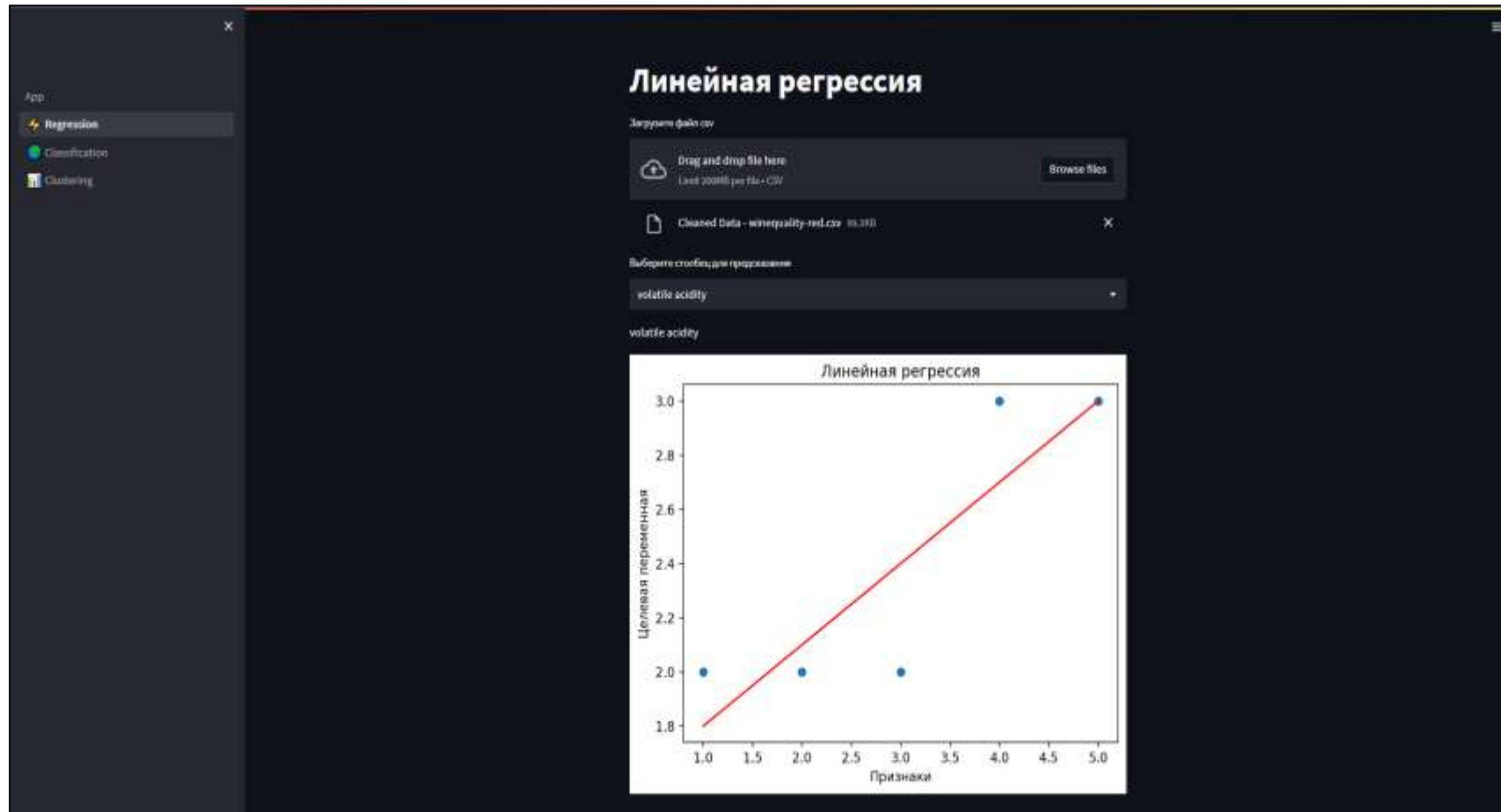


Рисунок 17 – Вкладка с регрессией



Классификация

Классификация

Choose a CSV file

Drag and drop file here
(max 200MB per file)

Browse files

Iris.csv (116K)

id	Sepal.Length(Cm)	Sepal.Width(Cm)	Petal.Length(Cm)	Petal.Width(Cm)	Species
0	5.1000	3.5000	1.4000	0.2000	Iris-setosa
1	4.9000	3.0000	1.4000	0.2000	Iris-setosa
2	4.7000	3.2000	1.3000	0.3000	Iris-setosa
3	4.6000	3.1000	1.5000	0.2000	Iris-setosa
4	5.0000	2.6000	1.4000	0.2000	Iris-setosa
5	5.4000	3.6000	1.7000	0.4000	Iris-setosa
6	4.6000	3.4000	1.4000	0.3000	Iris-setosa
7	5.0000	3.4000	1.3000	0.2000	Iris-setosa
8	4.4000	2.9000	1.4000	0.2000	Iris-setosa
9	4.8000	3.1000	1.5000	0.1000	Iris-setosa

Iris.csv данные выбраны!

Logistic Regression алгоритм выбран!

Формат набора данных предиктора: 116, 4

Количество классов: 3

Давайте посмотрим на данные предиктора!

Sepal.Length(Cm)	Sepal.Width(Cm)	Petal.Length(Cm)	Petal.Width(Cm)
5.1000	3.5000	1.4000	0.2000

Рисунок 18 – Вкладка с классификацией



Классификация

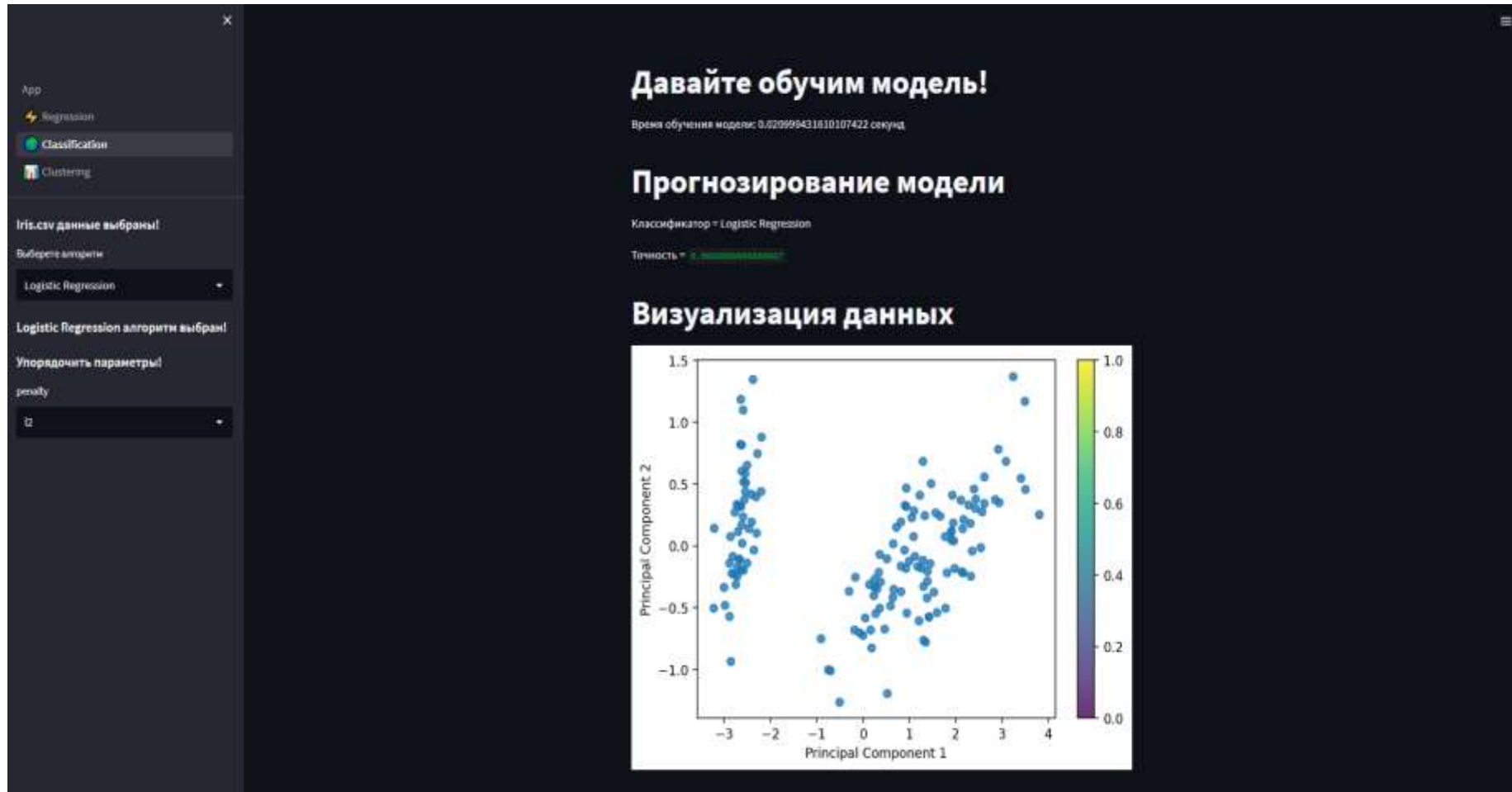


Рисунок 19 – Вкладка с классификацией



Кластеризация

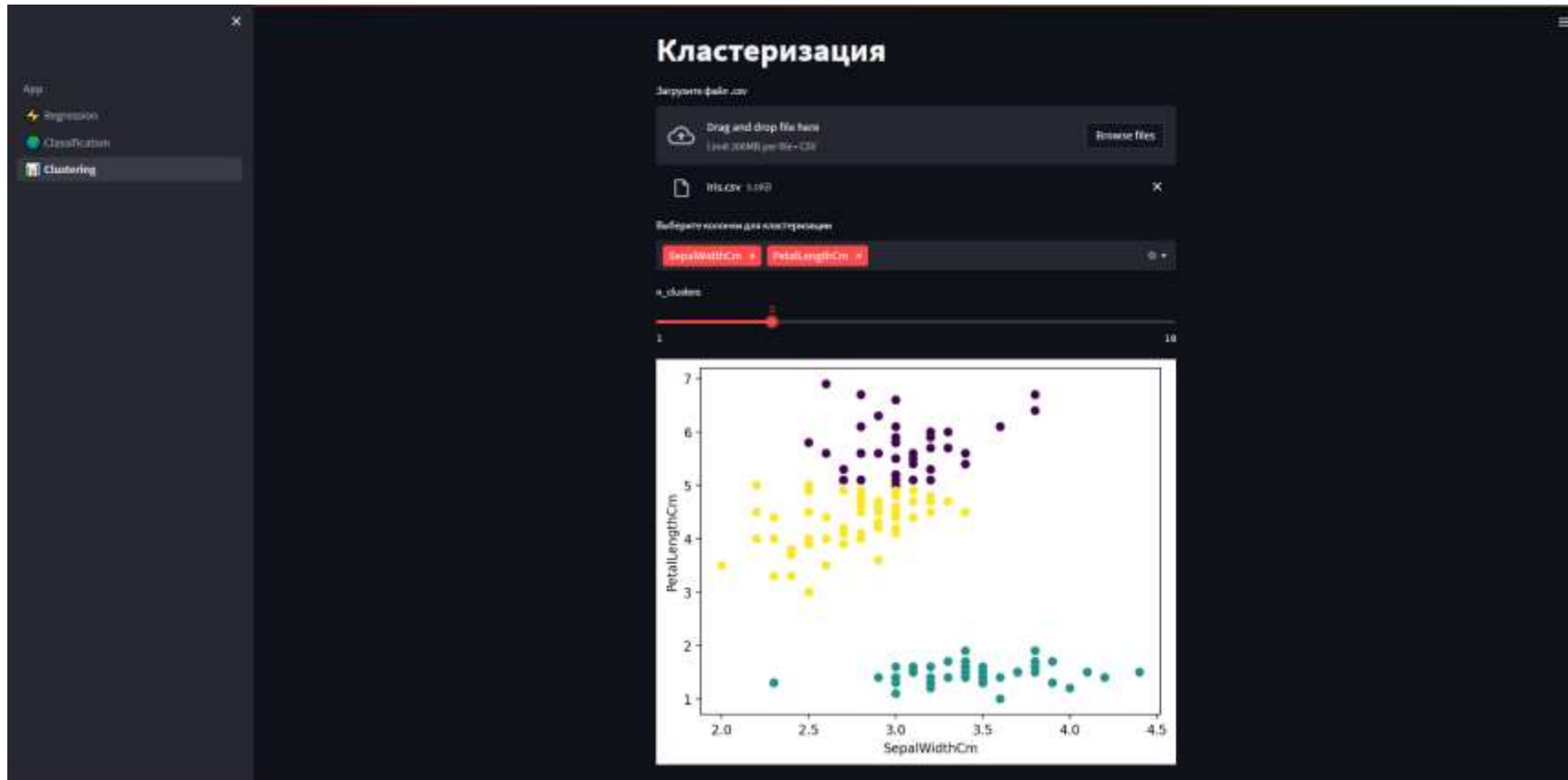


Рисунок 20 – Вкладка с кластеризацией



Планирование работ по теме и расчет полной стоимости проведения работ

График проведения работ (Рисунок 21):

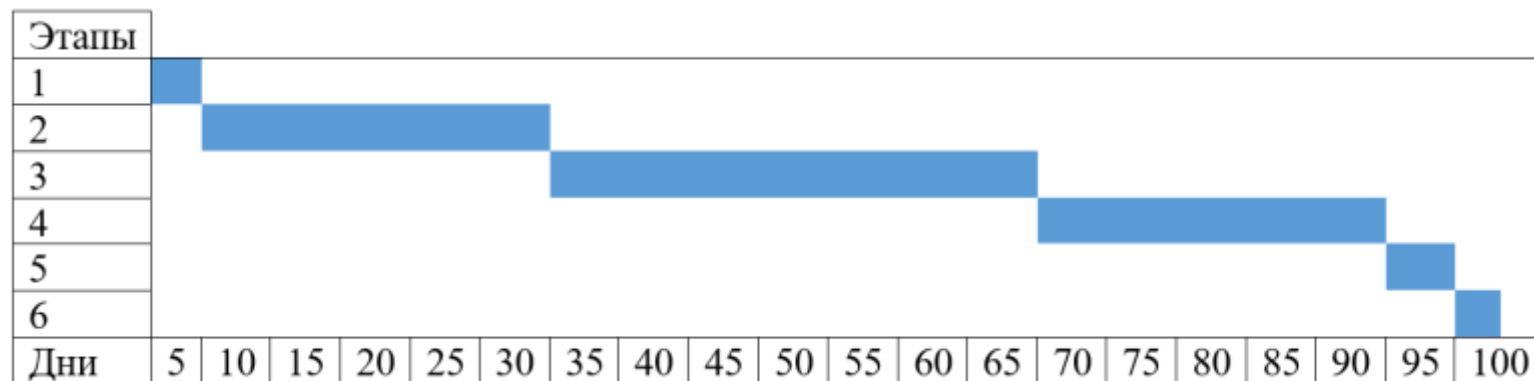


Рисунок 21 – График проведения работ

Договорная цена будет представлять собой:

$$\begin{aligned} \text{ДЦ} &= \text{С} + \text{П} + \text{НДС} \\ &= 696\,214,125 + 208\,864,238 + 905\,078,363 \\ &= 1\,810\,156,726 \end{aligned}$$



Результаты выпускной квалификационной работы

1. Проведен анализ предметной области и конкурентных веб-приложений.
2. Выбраны и обоснованы средства разработки веб-приложения и алгоритмы машинного обучения.
3. Разработана бизнес-логика приложения.
4. Спроектировано и разработано интерактивное веб-приложения для анализа и визуализации данных с использованием алгоритмов машинного обучения.
5. Проведено тестирование разработанного интерактивного веб-приложения.
6. Проведен расчет себестоимости проекта.



Апробация выпускной квалификационной работы



Рисунок 22 – Сертификат публикации учебно-методического материала

Спасибо за внимание!