

ВИЛИОН

ПРОГРАММНЫЙ ПРОДУКТ

ИНСТРУКЦИЯ ПО ЭКСПЛУАТАЦИИ
«ПРЕДИКТИВНОЕ ОПОВЕЩЕНИЕ О
ЧРЕЗВЫЧАЙНЫХ СИТУАЦИЯХ И
ИХ ПОСЛЕДСТВИЯХ»

Краснодар

2022

Оглавление

1. Назначение	3
2. Установка	4
3. Работа с конфигурационным файлом	6
4. Перечень сокращений.....	9

1. Назначение

Настоящий документ содержит инструкцию по установке программного обеспечения «Предиктивное оповещение о чрезвычайных ситуациях и их последствиях» и описывает развертывание ПО и формирование конфигурационного файла.

2. Установка

Установка ПО «Предиктивное оповещение о чрезвычайных ситуациях и их последствиях» осуществляется на ОС Linux Ubuntu 21.10 или совместимую.

Перед установкой необходимо убедиться в наличии в системе следующих пакетов:

- `librdkafka-dev`
- `python3`
- `python3-venv`
- `python3-pip`
- `python3-wheel`

Так же рекомендуется обновить пакеты Python следующей командой:

```
| python3 -m pip install --upgrade pip setuptools wheel
```

Установка осуществляется распаковкой архива с последующим выполнением следующей команды:

```
| pip install py_predict-1.2.0-py3-none-any.whl
```

Перед запуском необходимо провести конфигурацию в файле `conf.yml`, произвести запуск брокера `kafka`.

После установки доступны следующие исполняемые модули:

- `pypredict-api` — REST API подключения датчиков и получения результатов

- `pypredict-router` — Модуль сортировки
- `pypredict-colorlimit` — Модуль классификации уровня опасности
- `pypredict-criticallevel` — Модуль определения зоны подтопления
- `pypredict-predictor` — Диспетчер модулей прогноза
- `pypredict-consumer` — Модуль отладки очередей
- `pypredict-producer` — Модуль ручной загрузки

Для работы необходимо запустить все модули (кроме модулей отладки и ручной загрузки). Последовательность запуска не имеет значения.

3. Работа с конфигурационным файлом

Формат конфигурационного файла `conf.yml` с комментариями и примером заполнения:

```
# Порт запуска REST сервисов
```

```
  REST_SERVICE_PORT: 6066
```

```
# Параметры подключения к Kafka и Schema Registry
```

```
  KAFKA_CONSUMER_GROUP: 'pyPredictRouter-producer'
```

```
  KAFKA_BOOTSTRAP_SERVERS: 'localhost:9092'
```

```
  SCHEMA_REGISTRY_URL: 'http://localhost:8081'
```

```
# Входной топик для датчиков/данных
```

```
  SENSORS_TOPIC: 'sensors'
```

```
# Роутер входных данных - сортировка источников по разным очередям
```

```
  GRAINS_ROUTER: {  
    'river:a1': 'route-kuban',  
    'river:a2': 'route-belaya'  
  }
```

```
# Привязка алгоритмов к разным очередям
```

```
  GRAINS_PREDICTOR: {  
    'route-kuban': { CLASS: 'PredictorKuban', OUT_TOPIC: 'predicted-  
kuban' },
```

```
'route-belaya': { CLASS: 'PredictorBelaya', OUT_TOPIC: 'predicted-  
belaya' },  
}
```

```
# Уровни опасности для рек
```

```
COLOR_IN_TOPICS: ['predicted-kuban', 'predicted-belaya']
```

```
COLOR_LEVELS: {
```

```
  'river:a1': { 15: "Зеленый", 20: "Желтый", 25: "Красный" },
```

```
  'river:a2': { 15: "Зеленый", 20: "Желтый", 25: "Красный" },
```

```
}
```

```
COLOR_OUT_TOPIC: 'warn-levels'
```

```
# Определение зоны затопления для рек
```

```
CRITICAL_IN_TOPICS: ['predicted-kuban', 'predicted-belaya']
```

```
CRITICAL_LEVELS: {
```

```
  'river:a1': {
```

```
    24: [ 'с.Калинино', 'мрн.Ньювасюки' ],
```

```
    26: [ 'Дальнее Бутово', 'х.Ленина', 'пос.Грушевка' ],
```

```
    28: [ 'аул Веселое' ],
```

```
  },
```

```
  'river:a2': {
```

```
    19: [ 'снт.Люблино' ],
```

```
    22: [ 'Гадюкино', 'х.Светлое' ],
```

```
    30: [ 'с.Грустное' ],
```

```
  },
```

```
}  
CRITICAL_OUT_TOPIC: 'warn-levels'
```

Дополнительные настройки отображения имен датчиков

```
FAMILY_NAMES: {  
  'river:a1': 'Кубань (Усть-Лабинс)',  
  'river:a2': 'Белая (Майкоп)'  
}
```

Настройки логгирования

```
LOGGING_LEVEL: 'INFO'
```

```
LOGGING_FORMAT:   '%(asctime)-15s   |   %(levelname)s   |  
%(filename)s#%(lineno)d | %(message)s'
```


4. Перечень сокращений

ОС — Операционная система

ПО — Программное обеспечение