Техническое задание для приложения обработки данных водопользования

---

1. Цель проекта

Разработка Python-приложения для автоматической обработки XLS-файлов из государственного водного реестра, содержащих данные о водопользователях. Приложение должно:

- Разделять данные по административным округам/областям.
- Генерировать KML-файлы для визуализации на картографических сервисах (Google Maps, Yandex Maps).
- Выделять аномальные данные (некорректные/отсутствующие координаты) в отдельные файлы.

---

2. Требования к входным данным

- Формат файла: XLS/XLSX.
- Структура данных (на основе примера):
    - Колонки: № п/п, Водопользователь, ИНН, Наименование водного объекта, Место водопользования, Цель водопользования, Дата прекращения действия договора.
    - Координаты указаны в колонке "Место водопользования" в форматах:
        - Градусы, минуты, секунды (например: 53°46'28"СШ 127°16'28"ВД).
        - Десятичные градусы (например: 50.15'2"СШ).

---

3. Функциональные требования

3.1. Парсинг и обработка данных

- Автоматическое извлечение административного округа/области из колонки "Место водопользования" (пример: "Зейский район р-н", "Амурская область").
- Стандартизация координат в десятичный формат (широта, долгота).
- Проверка корректности координат:
    - Широта: от -90 до 90.
    - Долгота: от -180 до 180.
- Обработка ошибок:
    - Записи с некорректными координатами или их отсутствием помечаются как аномальные.

3.2. Генерация выходных файлов

- **Для корректных данных**:

- Создание KML-файлов с метками водопользователей.
- Название файла: [Название_округа].kml (например: Амурская_область.kml).
- В метках KML указывать: наименование водопользователя, цель водопользования, дату прекращения действия договора.

- **Для аномальных данных**:
  - Создание KML-файла с префиксом ANO_ (например: ANO_Амурская_область.kml).
  - Создание XLS-файла с префиксом ANO_ (например: ANO_Амурская_область.xlsx), содержащего исходные данные с пометкой причины аномалии.

3.3. Дополнительные требования

- Логирование ошибок в файл errors.log.
- Поддержка обработки больших файлов (оптимизация по памяти и скорости).

---

4. Технологический стек

- **Python 3.9+**.
- **Библиотеки**:
  - pandas — для работы с XLS-файлами.
  - openpyxl — чтение/запись Excel.
  - simplekml — генерация KML-файлов.
  - re — регулярные выражения для парсинга координат.
  - logging — логирование ошибок.
- **Интеграция с ИИ** (опционально):
  - Использование NLP-моделей (например, spaCy) для автоматического извлечения названий регионов из текста.
  - Классификация аномалий с помощью ML-моделей (например, scikit-learn).

---

5. Этапы разработки

1. **Анализ структуры данных**: Изучение форматов координат и шаблонов названий регионов.
2. **Парсинг данных**:
   - Извлечение административных единиц.
   - Конвертация координат в десятичный формат.
3. **Валидация данных**:

- Проверка диапазонов широты/долготы.
- Фильтрация аномалий.

4. **Генерация KML/XLS**:
    - Создание файлов для каждого региона.
    - Формирование меток с данными водопользователей.
5. **Тестирование**:
    - Проверка на примере из ТЗ.
    - Обработка крайних случаев (отсутствие координат, разнородные форматы).

---

6. Требования к качеству

- Корректное отображение 95%+ записей на карте.
- Обработка файла объемом до 10 000 строк за время ≤ 5 минут.
- Четкая структура выходных файлов и логирование ошибок.

---

7. Пример работы

Входные данные:
Строка: 1 | дата | ООО "Малый Гармакан" | ... | Зея г (залив Малый Гармакан) | 53°46'28"СШ 127°16'28"ВД ...

Выходные данные:

- Файл: Амурская_область.kml с меткой по координатам (53.774444, 127.274444).
- Для аномалий: ANO_Амурская_область.xlsx с записью: "Неверный формат координат: 55°XX'YY"ZZ".

---

8. Сроки и бюджет

- Срок разработки: 4 недели.
- Бюджет: определяется исполнителем.