

УДК 004.415.2

EDN: VMWYAT

## ПРИЛОЖЕНИЕ ДЛЯ УДАЛЕННОГО ВЗАИМОДЕЙСТВИЯ С БАЗОЙ ДАННЫХ НАБЛЮДЕНИЙ АПВЕЛЛИНГОВ

**Н.А. Санников**

ORCID: 0000-0001-9609-9786 e-mail: sannikov\_na@mail.ru

Нижегородский государственный технический университет им. Р.Е. Алексеева  
*Нижний Новгород, Россия***А.С. Епифанова**

ORCID: 0000-0003-1061-0964 e-mail: epifanova.anastasia.s@gmail.com

Нижегородский государственный технический университет им. Р.Е. Алексеева  
*Нижний Новгород, Россия***А.В. Рыбин**

ORCID: 0000-0001-5084-3610 e-mail: arybin93@gmail.com

Нижегородский государственный технический университет им. Р.Е. Алексеева  
*Нижний Новгород, Россия***С.А. Епифанов**

ORCID: 0009-0000-5585-6082 e-mail: epifanov.s.a@yandex.ru

Нижегородский государственный технический университет им. Р.Е. Алексеева  
*Нижний Новгород, Россия*

Дано описание веб-приложения для удаленного взаимодействия с базой данных наблюдений апвеллингов, содержащей информацию о собранных из литературных источников случаях подъемов глубинных вод к поверхности водоемов. Рассмотрены поля базы данных, схема работы приложения, входящие в него структурные элементы, приведены примеры реализации графического интерфейса. Разработанный программный продукт может применяться при разработке устойчивых стратегий использования природных ресурсов прибрежных зон и оптимизации экономической деятельности на шельфе.

**Ключевые слова:** веб-приложение, Django, Python, база данных, апвеллинг, Мировой океан.

**ДЛЯ ЦИТИРОВАНИЯ:** Санников, Н.А. Приложение для удаленного взаимодействия с базой данных наблюдений апвеллингов / Н.А. Санников, А.С. Епифанова, А.В. Рыбин, С.А. Епифанов // Труды НГТУ им. Р.Е. Алексеева. 2024. № 3. С. 14-24. EDN: VMWYAT

## WEB APPLICATION FOR REMOTE INTERACTION WITH UPWELLING OBSERVATIONS DATABASE

**N.A. Sannikov**

ORCID: 0000-0001-9609-9786 e-mail: sannikov\_na@mail.ru

Nizhny Novgorod State Technical University n.a. R.E. Alekseev  
*Nizhny Novgorod, Russia***A.S. Epifanova**

ORCID: 0000-0003-1061-0964 e-mail: epifanova.anastasia.s@gmail.com

Nizhny Novgorod State Technical University n.a. R.E. Alekseev  
*Nizhny Novgorod, Russia*

**A.V. Rybin**

ORCID: **0000-0001-5084-3610** e-mail: **arybin93@gmail.com**  
Nizhny Novgorod State Technical University n.a. R.E. Alekseev  
*Nizhny Novgorod, Russia*

**S.A. Epifanov**

ORCID: **0009-0000-5585-6082** e-mail: **epifanov.s.a@yandex.ru**  
Nizhny Novgorod State Technical University n.a. R.E. Alekseev  
*Nizhny Novgorod, Russia*

**Abstract.** The article describes a web application for remote interaction with a database of upwelling observations containing information on cases of motion of deep water towards the ocean surface collected from references. The database fields, the application operation scheme, the structural elements included in it are considered, and examples of the implementation of the graphical interface are given. The results of the study can be used to develop sustainable strategies for the use of natural resources in coastal zones and to optimize economic activity on the shelf.

**Key words:** web application, Django, Python, database, upwelling, World Ocean.

**FOR CITATION:** N.A. Sannikov, A.S. Epifanova, A.V. Rybin, S.A. Epifanov. WEB application for remote interaction with upwelling observations database. Transactions of NNSTU n.a. R.E. Alekseev. 2024. № 3. Pp. 14-24. EDN: VMWYAT

## Введение

Изучение процессов водообмена в шельфовых морских зонах в настоящее время является задачей, требующей пристального внимания. Динамические явления, происходящие в этих областях, оказывают значительное воздействие на экологическое состояние окружающей среды в морских акваториях. Одним из ключевых процессов, вносящих значительный вклад в водо- и массообмен шельфовых вод, является апвеллинг – процесс подъема воды из глубин водоема к его поверхности [1]. Это явление можно наблюдать как у берегов, так и в центральных районах океанов, морей и озер. Глубинные воды, поднимающиеся к поверхности, характеризуются более низкой температурой, наличием биогенных веществ и растворенных в них химических элементов; поверхностные воды обогащаются питательными веществами, такими как фосфор и азот [2, 3]. В морских акваториях это способствует активному развитию фитопланктона, который играет ключевую роль в пищевой цепи морских экосистем, являясь основным источником пищи для многих морских животных [4]. В то же время апвеллинг может вызывать экологические проблемы (несбалансированная эвтрофикация вод, образование сероводорода [5, 6], климатические изменения температуры вод, изменение структуры морских сообществ, цветение фито- и зоопланктона [5]). Кроме того, этот процесс способен привести к изменению солевого состава толщи воды, что может негативно отразиться на многих морских организмах [7, 8] и нарушить биологическое равновесие.

Таким образом, создание базы данных, содержащей информацию об описанных в литературе случаях апвеллингов, а также веб-приложения для работы с ней является актуальной задачей. Хранение и обработка тематических данных с помощью подобных программных комплексов в последнее время становится неотъемлемым элементом исследований в области океанологии [9-14].

## Схема работы приложения

Для разработки веб-приложения была выбрана бесплатная программная платформа *Django*. Она реализована на объектно-ориентированном языке программирования *Python*, что обеспечивает модульность всего проекта в целом и облегчает как начальную разработку программного продукта, так и дальнейшее его развитие. Кроме того, *Django* изначально

была создана для реализации веб-интерфейсов, поэтому в ней заложен весь необходимый инструментарий для безопасного взаимодействия с записями базы данных.

Кратко схему работы приложения (рис. 1) можно описать следующим образом:

- пользователь с помощью графического интерфейса (браузер) формирует HTTP GET запрос на получение данных и отправляет его на сервер приложения;
- сервер обрабатывает запрос пользователя и формирует ответ в виде шаблонного файла, т.е. т.н. *template* в контексте *Django Framework*, формата *html* и контекстных данных, которые могут состоять как из содержимого базы данных, так и из частей пользовательского интерфейса, которые сформированы со стороны сервера; при этом пользователь не имеет права вносить какие-либо изменения в базу данных – в данном случае ему доступен только просмотр информации.

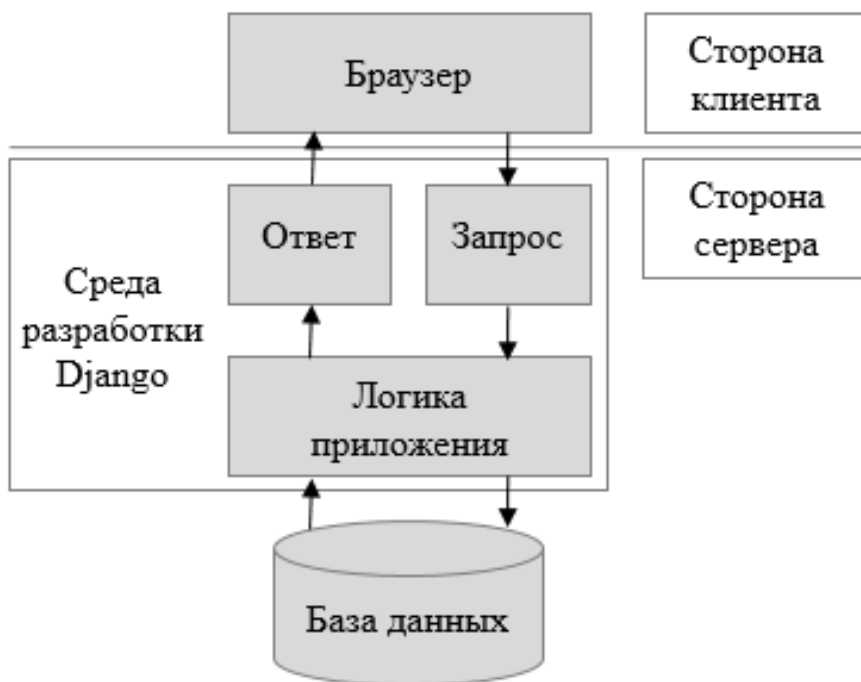


Рис. 1. Блок-схема работы приложения для удаленного взаимодействия с базой данных наблюдений апвеллингов

Fig. 1. Flowchart of the web application for interaction with the upwelling observation database

Добавление записей происходит через отдельно реализованный интерфейс администратора, работающий по схожему принципу.

### Описание полей базы данных

База данных наблюдений апвеллингов в Мировом океане создана на основе реляционной системы управления базой данных *MySQL*. Поскольку на текущий момент база данных активно пополняется, нет полной уверенности в том, что выделены все основные сущности и их атрибуты. Поэтому схема ее на данный момент денормализована и будет нормализовываться в дальнейшем. В табл. 1 представлено описание всех текущих параметров, которые учитываются при добавлении наблюдения.

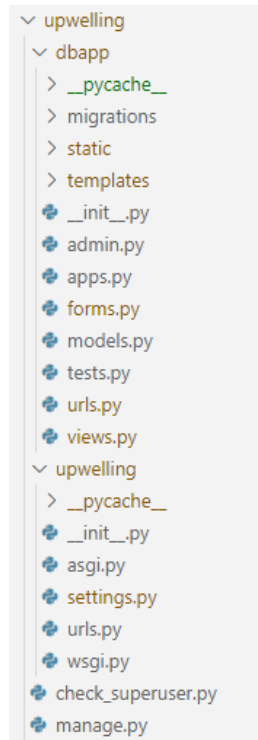
Таблица 1.  
Таблица параметров наблюдаемого явления

Table 1.  
Observed phenomenon parameters table

Поле	Тип данных	Описание
type_welling	Varchar(45)	тип наблюдаемого явления
reservoir	Varchar(45)	название водоема, в котором проводились замеры
measurement_place	Varchar(45)	место, в котором проводились замеры (бухта, мыс и т.п.)
coordinate_y	Float	широта (градусы)
latitude	Varchar(1)	сторона света по широте
coordinate_x	Float	долгота (градусы)
longitude	Varchar(1)	сторона света по долготе
source_title	Varchar(1000)	литературный источник, содержащий информацию о наблюдении
source_url	Varchar(512)	ссылка на источник
strip_length	Float	протяженность полосы (км)
avg_sea_depth	Float	средняя глубина моря (м)
layer_thickness	Float	толщина слоя (м)
width_on_surface	Float	ширина на поверхности (км)
width_on_bottom	Float	ширина у дна (км)
distance_from_shore	Float	расстояние от берега (км)
temperature_diff	Float	разница температур (°C)
temperature_diff_max	Float	максимальный перепад температур между поднятой и окружающей водами (°C)
flow_speed	Float	скорость течения (м/с)
seawater_dens_diff	Float	разность плотностей морской воды (епс)
salinity	Float	соленость (psu)
sea_surf_temp_min	Float	минимальное значение температуры поверхности моря (°C)
sea_surf_temp_max_diff	Float	максимальный перепад температуры поверхности моря между зоной явления и смежными водами (°C)
wind_speed	Float	скорость ветра (м/с)
vertical_speed	Float	вертикальная скорость (м/с)
water_temp_max_out	Float	температура водной массы за пределами апвеллинга $T_{max}$ (°C)
territorial_waters	Varchar(45)	территориальные воды, в которых проводились замеры
start_date	Date	дата начала явления (гг-мм-дд)
end_date	Date	дата окончания явления (гг-мм-дд)
duration	Int	продолжительность явления (дни)

### Структура приложения

Поскольку разработка велась в рамках готовой среды *Django* [15], структура приложения представляет собой шаблонную иерархию файлов с возможностью внедрения собственных реализаций разработчика и выглядит следующим образом (рис. 2):



**Рис. 2. Файловая структура веб-приложения для удаленного взаимодействия с базой данных наблюдений апвеллингов**

**Fig. 2. File structure of the web application for interaction with the upwelling observation database**

Здесь:

- `upwelling` – корневая директория проекта;
- `dbapp` – директория приложения, в которой расположены файлы, связанные с взаимодействием с базой данных и пользователем;
- `__pycache__` – автоматическая директория, хранящая скомпилированные байт-коды файлов;
- `migrations` – директория для хранения файлов миграции;
- `static` – директория с файлами стилей формата `css`, изображений и скриптов формата `javascript`;
- `templates` – директория для хранения шаблонных файлов формата `html`;
- `__init__.py` – автоматически сгенерированный файл, позволяет *Python* распознать директорию как пакет;
- `admin.py` – исполняемый файл, в котором реализуется функционал панели администратора;
- `apps.py` – файл с указанием названий подключаемых сторонних приложений;
- `forms.py` – исполняемый файл, содержащий реализацию форм запроса данных для пользователя;
- `models.py` – исполняемый файл с описанием таблиц базы данных и их полей;
- `tests.py` – исполняемый файл приложения, который хранит тесты приложения;
- `urls.py` – исполняемый файл с описанием путей к различным страницам сайта;
- `views.py` – исполняемый файл проекта с реализациями функций, генерирующих различные страницы сайта и обрабатывающих запросы пользователя;
- `upwelling` – директория проекта, в которой расположены основные настоечные файлы;
- `asgi.py` – автоматически сгенерированный файл, добавляет поддержку взаимодействия между асинхронными веб-серверами и приложениями;

- settings.py – исполняемый файл проекта с настроечными данными для его работы;
  - urls.py – исполняемый файл проекта с описанием путей к различным страницам сайта;
  - wsgi.py – автоматически сгенерированный файл, позволяющий приложению работать с веб-сервером по протоколу WSGI (Web Server Gateway Interface);
  - check\_superuser.py – скрипт для проверки количества суперпользователей в проекте;
  - manage.py – основной файл для управления проектом.
- Наиболее значимые файлы будут рассмотрены ниже.

### Организация работы с базой данных

Вследствие того, что используемая среда разработки основана на объектно-ориентированном языке программирования *Python*, большая часть функционала приложения реализуется в виде классов и методов классов. Так, например, реализация таблиц базы данных и их полей представляет собой набор классов и атрибутов соответственно в файле models.py, т.н. ORM (Object-relational mapping) технология (рис. 3). Такой подход значительно упрощает разработку, так как при использовании команд «python./manage.py makemigrations» и «python ./manage.py migrate» в корневой папке информация о структуре схемы базы данных отправляется на локальный сервер. После выполнения первой команды можно проверить правильность составления запроса, посмотрев содержимое файлов в папке приложения migrations. Предварительно необходимо указать адрес локального сервера *MySQL* и имя базы данных в настроечном файле проекта settings.py. Для добавления, редактирования и удаления записей в среде *Django* используется административная панель, которая так же, как и все модули, реализовывается через определение классов в файле admin.py. Благодаря этому разработчик способен выполнять необходимые настройки: определять порядок и количество отображаемых полей базы данных, редактировать шаблон полей для заполнения данных и показывать дочерние таблицы по внешнему ключу (рис. 4).

```
from django.db import models

# Create your models here.

class WellingTable(models.Model):
    type_welling = models.CharField(max_length=45, null=True, blank=True, default=None, verbose_name="Welling type")

class DateTable(models.Model):
    welling_table = models.ForeignKey(WellingTable, on_delete=models.CASCADE, verbose_name="WellingTableFK")
    start_date = models.DateField(null=True, blank=True, default=None, verbose_name="Start date")
    end_date = models.DateField(null=True, blank=True, default=None, verbose_name="End date")
    duration = models.IntegerField(null=True, blank=True, default=None, verbose_name="Duration of observation")

class SpatialTable(models.Model):
    welling_table = models.ForeignKey(WellingTable, on_delete=models.CASCADE, verbose_name="WellingTableFK")
    strip_length = models.FloatField(null=True, blank=True, default=None, verbose_name="Strip length")
    avg_sea_depth = models.FloatField(null=True, blank=True, default=None, verbose_name="Avg sea depth")
    layer_thickness = models.FloatField(null=True, blank=True, default=None, verbose_name="Layer thickness")

class SourceTable(models.Model):
    welling_table = models.ForeignKey(WellingTable, on_delete=models.CASCADE, verbose_name="WellingTableFK")
    source_title = models.CharField(max_length=1000, null=True, blank=True, default=None, verbose_name="Source title")
    source_url = models.CharField(max_length=512, null=True, blank=True, default=None, verbose_name="Source URL")
```

**Рис. 3. Пример реализации таблиц базы данных в виде классов на языке Python в файле models.py**

**Fig. 3. Example of database tables implementation as Python classes in models.py file**

Администрирование Django ДОБРО ПОЖАЛОВАТЬ, RO

Начало · Dbapp · Wellings tables

Выберите welling table для изменения

Q

Действие:   Выбрано 0 объектов из 6

<input type="checkbox"/>	ID	WELLING TYPE	DATE	SPATIAL	SOURCE	LOCATION	PHYSIC
<input type="checkbox"/>	7	апвеллинг	Начало замеров: 15 июля 2006 Конец замеров: 25 августа 2006 Продолжительность: 41 Д.	Протяженность полосы 300.0 км Средняя глубина моря None м Толщина слоя None м	Козлов И. Е., Дайлидене И. Изучение апвеллинга в Балтийском море на основе спутниковых оптических и РЛ-данных //Современные проблемы дистанционного зондирования Земли из космоса – 2010. – Т. 7. – №. 3. – С. 161-167.	Страна: Россия Водоем: Балтийское море Территория: юго-восточная часть Балтийского моря Координаты: 20.6 E 57.0 N	Разница температур: 10.0 °C Макс. перепад темп. между глуб. и пов. водами: None °C Скорость течения: None м/с Разность плотностей морской воды: None eps Соленость: None psu Мин. значение темп. пов. воды: 6.0 °C Макс. перепад темп. пов. воды между зоной явления и смеж. водами: 15.0 °C Скорость ветра: 5.0 м/с Вертикальная скорость: None м/с Темп. вод. массы за пределами апвеллинга Tmax: None °C
<input type="checkbox"/>	6	апвеллинг	Начало замеров: 26 августа 2002 Конец замеров: 23 сентября 2002 Продолжительность: 28 Д.	-	Журбас М. и др. Мезомасштабная изменчивость апвеллинга в юго-восточной Балтике: ИК-изображения и численное моделирование //Океанология. – 2004. – Т. 44. – №. 5. – С. 660-669.	Страна: Россия Водоем: Балтийское море Территория: юго-восточная часть Балтийского моря Координаты: None None None None	Разница температур: None °C Макс. перепад темп. между глуб. и пов. водами: None °C Скорость течения: None м/с Разность плотностей морской воды: None eps Соленость: None psu Мин. значение темп. пов. воды: None °C Макс. перепад темп. пов. воды между зоной явления и смеж. водами: None °C Скорость ветра: 7.0 м/с Вертикальная скорость: None м/с Темп. вод. массы за пределами апвеллинга Tmax: None °C
<input type="checkbox"/>	5	апвеллинг	Начало замеров: 12 августа 2009	Протяженность полосы 20.0 км	Демидов А. Н. и др. Особенности структуры и динамики вод в прибрежной части Балтийского моря вблизи Самбийского полуострова //Вестник Московского	Страна: Россия Водоем: Балтийское море	Разница температур: None °C Макс. перепад темп. между глуб. и

Рис. 4. Пример графического интерфейса панели администратора разработанного приложения

Fig. 4. Example of the graphical interface of the administrator panel of developed web application

Для доступа как к панели администратора, так и к основному функционалу веб-приложения, пользователь должен ввести в адресную строку браузера путь до нужного ему раздела (или перейти по контекстным ссылкам в шаблонах html). Подобный тип запроса обрабатывается модулем в файле `urls.py`. В рамках данной работы существует два файла с подобным названием: один – в папке проекта, другой – в папке приложения. Это сделано для удобства разработки, так как все пути в разделы приложения реализованы только в его директории и нигде более. Перечисление доступных пользователю разделов веб-приложения приведено в файле `urls.py`, расположенном в директории `dbapp`. Шаблоны html страниц с данными формируются и отправляются пользователю путем вызова функций из файла `views.py` модуля `views`.

Модуль `views` представлен, в основном, как набор функций, реализованных разработчиком. В качестве аргумента передается массив данных запроса пользователя и возвращается название шаблонного файла html и сами данные. Именно в этом разделе происходит все взаимодействие с базой данных и обработка хранящихся в ней записей.

### Графический интерфейс пользователя

Графический интерфейс пользователя в среде *Django* представлен в виде файлов формата html, которые представляют собой шаблоны веб-страниц с набором тегов, к которым можно применить дополнительный функционал. Он может быть представлен как в виде скриптов на основе javascript, так и отправкой форм из среды *Django*. Пользователю доступен следующий функционал: просмотр географического распределения зарегистрированных случаев апвеллинга, поиск данных по автору или названию источника, доступ к файлу статьи (если он открыт для общего ознакомления), просмотр гистограмм распределения наблюдений по акваториям, по годам и по сезонам. Визуализация отдельных наблюдений осуществляется с помощью интерактивной карты, реализованной при помощи картографического сервиса Яндекс Карты. На рис. 5 представлен пример отображения записей, реализованный в элементе `uamar.js`, расположенном в директории `static`.

```

ymaps.ready(init);
function init() {
  yaMap = new ymaps.Map("map", {
    center: [59.20, 21.25],
    zoom: 5,
    type: 'yandex#satellite',
    controls: ['zoomControl', 'fullscreenControl', 'rulerControl']
  });
}

```

Рис. 5. Функция, реализующая визуализацию интерактивной карты в файле yamap.js

Fig. 5. Interactive map visualization function in yamap.js file

Подобные файлы вызываются в шаблонах html в разделе <head> (рис. 6). Карта будет расположена в том месте, где указан нужный тег «<div>» с необходимым идентификатором. В рассматриваемом случае необходимый для карты тег будет выглядеть следующим образом: <div id="map"></div>. Результат представлен на рис. 7.

```

<!DOCTYPE html>
<html lang="ru">
<head>
  {% block title %}<title>Upwelling</title>{% endblock %}
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1" />

  <!-- style -->
  {% load static %}
  <link rel="stylesheet" href="{% static 'css/styles.css' %}" />
  <link
    rel="stylesheet"
    href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" />

  <!-- scripts -->
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
  <script src="https://api-maps.yandex.ru/2.1/?apikey=08bbb439-3bb4-4624-9139-732963ddac86&lang=ru_RU" type="text/javascript"></script>
  <script src="{% static 'script/yamap.js' %}" type="text/javascript"></script>
  {% block head %}{% endblock %}
</head>

```

Рис. 6. Пример подключения статичных файлов в разделе <head>

Fig. 6. Example of connection of static files in the section <head>

#### Разделы сайта

Теоретическая справка  
 Карта  
 Распределение по акваториям  
 Распределение по годам  
 Распределение по сезонам  
 Поиск по авторам

#### Карта наблюдений

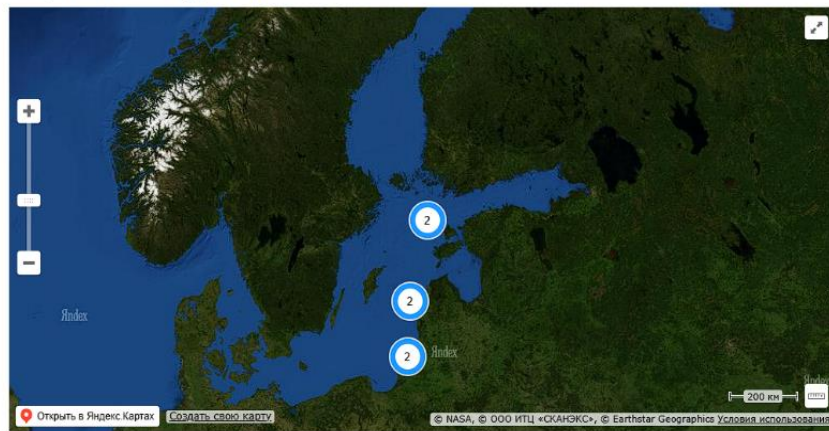


Рис. 7. Окно интерактивной карты веб-приложения

Fig. 7. Web application interactive map window



Для отображения заданного шаблона списка данных со стороны сервера приходят записи из базы данных и формы для их поиска в формате JSON, которые в *Django* реализуются в виде классов на языке *Python* и расположены в файле `forms.py`. В результате может быть сформирован динамический интерфейс (рис. 8).

<p>Разделы сайта</p> <ul style="list-style-type: none"> <li>Теоретическая справка</li> <li>Карта</li> <li>Распределение по акваториям</li> <li>Распределение по годам</li> <li>Распределение по сезонам</li> <li>Поиск по авторам</li> </ul>	<p>Список источников</p> <p>Автор/Название: <input type="text" value="Введите имя автора или н"/> <input type="button" value="Поиск"/></p> <ul style="list-style-type: none"> <li>• Козлов И. Е., Дайлидене И. Изучение апвеллинга в Балтийском море на основе спутниковых оптических и РЛ-данных // Современные проблемы дистанционного зондирования Земли из космоса. – 2010. – Т. 7. – №. 3. – С. 161-167.</li> <li>• Голенко М. Н. Структурные особенности апвеллинга в Юго-Восточной Балтике // Вестник Балтийского федерального университета им. И. Канта. Серия: Естественные и медицинские науки. – 2009. – №. 1. – С. 35-42.</li> <li>• Голенко Н. Н., Голенко М. Н., Щука С. А. Наблюдение и моделирование апвеллинга в Юго-Восточной Балтике // Океанология. – 2009. – Т. 49. – №. 1. – С. 20-27.</li> <li>• Демидов А. Н. и др. Особенности структуры и динамики вод в прибрежной части Балтийского моря вблизи Самбийского полуострова // Вестник Московского университета. Серия 5. География. – 2011. – №. 1. – С. 41-47.</li> <li>• Журбас М. и др. Мезомасштабная изменчивость апвеллинга в юго-восточной Балтике: ИК-изображения и численное моделирование // Океанология. – 2004. – Т. 44. – №. 5. – С. 660-669.</li> <li>• Козлов И. Е., Дайлидене И. Изучение апвеллинга в Балтийском море на основе спутниковых оптических и РЛ-данных // Современные проблемы дистанционного зондирования Земли из космоса. – 2010. – Т. 7. – №. 3. – С. 161-167.</li> </ul>
--	---

**Рис. 8. Отображение реализованного шаблона списка данных**

**Fig. 8. Implemented references list template**

Также в веб-приложении осуществляется первичная обработка данных. На рис. 9, 10 показана реализация постобработки: статистика записей по годам, сезонам и месяцам со стороны сервера и пример графического интерфейса с результатом запроса по временам года соответственно. Для построения гистограмм использовалась бесплатно распространяемая библиотека `matplotlib`.

```
def plot_stat(request):
    data = defaultdict(int)

    if 'year_stat' in request.GET:
        values = DataTable.objects.values("start_date")
        debug = values
        for value in values:
            data[value["start_date"].year] += 1
    elif 'season_stat' in request.GET:
        values = DataTable.objects.values("start_date")
        debug = values
        for value in values:
            if value["start_date"].month in [12, 1, 2]:
                data["Winter"] += 1
            elif value["start_date"].month in [3, 4, 5]:
                data["Spring"] += 1
            elif value["start_date"].month in [6, 7, 8]:
                data["Summer"] += 1
            else:
                data["Fall"] += 1
    elif 'month_stat' in request.GET:
        values = DataTable.objects.values("start_date")
        debug = values
        for value in values:
            data[value["start_date"].month] += 1

    plt.bar(range(len(data)), list(data.values()), align='center')
    plt.xticks(range(len(data)), list(data.keys()))
    plt.yticks(np.arange(0, max(list(data.values()))+1, 1.0))
```

**Рис. 9. Реализация обработки данных в файле `views.py`**

**Fig. 9. Data processing in `views.py` file**



Рис. 10. Пример гистограммы распределения записей базы данных апвеллингов по сезонам

Fig. 10. Example of a histogram of upwelling database records by season

### Выводы

Изучение апвеллинга имеет важное значение для исследований Мирового океана, поскольку данный процесс может оказывать существенное влияние на биологическую продуктивность акваторий, распределение морских ресурсов и равновесие экосистем. Однако, несмотря на то, что изучению этого явления посвящены многочисленные исследования отечественных и зарубежных ученых, на сегодняшний день нам неизвестно ни одного общедоступного и удобного в использовании каталога апвеллингов, содержащего информацию об описанных в литературе случаях глубинного подъема вод за весь период наблюдений. Для решения этой задачи разработаны веб-приложение на основе фреймворка *Django* и интегрированная в него постоянно пополняемая база данных. Пользователю доступен следующий функционал: просмотр географического распределения зарегистрированных случаев апвеллинга на интерактивной Яндекс Карте, поиск данных по автору или названию источника, а также доступ к файлу статьи (если он открыт для общего ознакомления), просмотр гистограмм распределения наблюдений по акваториям, годам и сезонам.

Разработанное приложение может представлять значительный интерес для океанологов, климатологов и специалистов в области морской экологии.

*Представленные результаты получены в рамках государственного задания в сфере научной деятельности (тема № FSWE-2023-0004 «Нелинейная волновая динамика прибрежной зоны в условиях меняющегося климата и антропогенного воздействия»).*

### Библиографический список

1. Сапожников, В.В. Механизм образования и развития апвеллинга в северо-восточной части Черного моря (район Анапа-Новороссийск) / В.В. Сапожников, Ю.Ю. Ткаченко, В.И. Денисов // Географические исследования Краснодарского края. – 2007. С. 136-146.

2. **Шевченко, Г.В.** Гидролого-гидрохимическая структура вод в зоне апвеллинга у юго-западного побережья о. Сахалин / Г.В. Шевченко, В.Е. Марыжихин, В.Н. Частиков // *Океанологические исследования*. 2023. Т. 51. №. 1. С. 54-70.
3. **Духова, Л.А.** Гидрохимическая структура и формирование биологической продуктивности вод в районе Канарского апвеллинга: автореф. дис... канд. геогр. наук: 25.00.28 / Л.А. Духова. – Калининград, 2010. – 23 с.
4. **Гусейнова, Н.О.** Изучение апвеллинга по содержанию хлорофилла а в фитопланктоне западного Каспия за 2017 г. по данным дистанционного зондирования / Н.О. Гусейнова, А.А. Багомаев, Л.Ш. Ахмедова, Б.М. Курамагомедов // *Юг России: экология, развитие*. 2021. № 4 (61). С. 159-172.
5. **Полонский, А.Б.** Современная пространственно-временная изменчивость апвеллингов северо-западной части Черного моря и у побережья Крыма / А.Б. Полонский, М.А. Музылѐва // *Известия Российской академии наук. Серия географическая*. 2016. №. 4. С. 96-108.
6. **Брянцев, В.А.** Причины черноморских заморозов / Брянцев В.А., Фащук Д.Я. // *Сборник материалов 5-ой Всесоюзной конференции: Вопросы промышленной океанологии Мирового океана*. – Калининград, 1979. С. 23-26.
7. **Жуков, Л.А.** Общая океанология / Л.А. Жуков. – Л.: Гидрометеиздат, 1976.
8. **Gaines, S.** Upwelling. [Электронный ресурс] / S. Gaines, S. Airame // Режим доступа: <https://oceanexplorer.noaa.gov/explorations/02quest/background/upwelling/upwelling.html> (Дата обращения 27.05.2024).
9. **Епифанова, А.С.** База данных наблюдений внутренних волн в Мировом океане / А.С. Епифанова, А.В. Рыбин, Т.Е. Моисеенко, О.Е. Куркина, А.А. Куркин, Д.Ю. Тюгин // *Морской гидрофизический журнал*. 2019. Т. 35. №. 4. С. 395-403.
10. **Епифанова, А.С.** О разработке цифрового атласа наблюдений внутренних волн в мировом океане / А.С. Епифанова, А.А. Куркин, О.Е. Куркина, Т.Е. Моисеенко, А.В. Рыбин // *Труды НГТУ им. П.Е. Алексеева*. 2019. № 4. С.17-26.
11. **Кокоулина, М.В.** Океанографическая геоинформационная онлайн-система для расчета и анализа характеристик внутренних волн в стратифицированных бассейнах / М.В. Кокоулина, А.С. Епифанова, О.Е. Куркина, А.А. Куркин, М.С. Винокуров, С.А. Епифанов // *Труды НГТУ им. П.Е. Алексеева*. 2023. № 3. С. 18-29.
12. **Рувинская, Е.А.** Первая версия информационной системы обработки и анализа данных натуральных измерений внутренних волн в океане / Е.А. Рувинская, А.С. Епифанова, М.В. Кокоулина // *Труды НГТУ им. П.Е. Алексеева*. 2020. № 3(130). С. 53-63.
13. **Тюгин, Д.Ю.** Программный комплекс для численного моделирования внутренних гравитационных волн в мировом океане / Д.Ю. Тюгин, О.Е. Куркина, А.А. Куркин // *Фундаментальная и прикладная гидрофизика*. 2011. Т. 4. № 2. С. 32-44.
14. **Тюгин, Д.Ю.** Повышение производительности программного комплекса для моделирования внутренних гравитационных волн IGW Research с помощью Intel® Parallel Studio XE 2013 / Д.Ю. Тюгин, А.А. Куркин, Е.Н. Пелиновский, О.Е. Куркина // *Фундаментальная и прикладная гидрофизика*. 2012. Т. 5. № 3. С. 89-95.
15. Django [Электронный ресурс] // Режим доступа: <https://www.djangoproject.com/>

**Дата поступления  
в редакцию: 27.05.2024**

**Дата принятия  
к публикации: 16.07.2024**