



## **Рекомендательные сервисы поддержки научно-технической и научно-образовательной деятельности**

Информация, необходимая для установки программного  
обеспечения

## **Оглавление**

1. Рекомендательные сервисы поддержки научно-технической и научно-образовательной деятельности.....	3
2. Установка Рекомендательных сервисов поддержки научно-технической и научно-образовательной деятельности.....	4

## **1. Рекомендательные сервисы поддержки научно-технической и научно-образовательной деятельности**

Рекомендательные сервисы поддержки научно-технической и научно-образовательной деятельности (далее - Сервисы) предназначены для автоматизации процессов проведения исследований, разработки и выполнения экспертиз: поиска технологий, поиска изданий для опубликования результатов, поиск релевантных экспертов и рецензентов.

Сервисы предназначены для решения следующих задач:

- Подбор релевантных технологий по технологическим запросам (на основе смыслового сопоставления текстов научно-технической документации с технологическими запросами и техническими заданиями).
- Подбор экспертов / рецензентов по заданному полным текстом объекту экспертизы (грантовая заявка, текст подготовленной к печати научной статьи, диссертации и др.);
- Рекомендация изданий для опубликования научных результатов на основе анализа полного текста препринта научной статьи.
- Интерактивное формирование и уточнение поисковых запросов с применением рекомендательной модели ассоциированных понятий предметных областей науки и техники.
- Автоматическая классификация научно-технических и научно-образовательных документов по рубрикатору ГРНТИ.

Взаимодействие со всеми сервисами осуществляется через API по протоколу HTTP. Поддерживаемые методы — GET для операций получения данных и POST для операций отправки данных. Все запросы должны содержать корректные HTTP-заголовки. Успешные ответы от сервисов возвращаются с кодом состояния HTTP 200, а данные передаются в теле ответа в формате JSON. Структура запросов и описание параметров для каждой точки доступа приведены в соответствующих подразделах.

## 2. Установка Рекомендательных сервисов поддержки научно-технической и научно-образовательной деятельности

Для развертывания Рекомендательных сервисов поддержки научно-технической и научно-образовательной деятельности» (далее – Рекомендательные сервисы) необходимо получить у разработчика заархивированный образ, включающий программное обеспечение и данные Рекомендательных сервисов. Для установки Рекомендательных сервисов необходимы вычислительные средства (далее – хост) под управлением операционной системы Debian 12 или выше с характеристиками не ниже указанных:

- CPU: x86-64, 10 ядер с базовой тактовой частотой не менее 3.0 ГГц.
- RAM: ~512GB.
- Тип накопителей: SSD, HDD.
- Общий физический объем накопителей: 10TB (SSD), 21TB (HDD).

На указанном хосте необходимо выполнить следующие шаги:

1. Скопировать полученный у разработчика архив на хост по адресу <PATH\_TO>/rec-services.tar.gz, где *rec-services.tar.gz* – условное наименование файла с архивом, <PATH\_TO> – директория, в которую будет загружен архив.

1. Подготовка хоста (на примере хоста с ОС Debian 12).

В качестве сервиса управления сетью должен использоваться *systemd-networkd*.

1.1. Установка *systemd-nspawn*:

- 1.1.1. \$ sudo su
- 1.1.2. # apt install systemd-container
- 1.1.3. # echo 'kernel.unprivileged\_userns\_clone=1' >/etc/sysctl.d/nspawn.conf
- 1.1.4. # systemctl restart systemd-sysctl.service

1.2. Установка *systemd-networkd* и настройка сетевого моста *br0*

- 1.2.1. \$ sudo su
- 1.2.2. # apt install bridge-utils
- 1.2.3. # mv /etc/network/interfaces /etc/network/interfaces.bac
- 1.2.4. Создать три файла в /etc/systemd/network/:
  - *br0.network*
  - *br0.netdev*
  - *enp0s3.network*

, где *enp0s3* – название основного сетевого интерфейса (для просмотра списка доступных интерфейсов можно воспользоваться командой «\$ ip -a»). Название на конкретном устройстве (хосте) может быть другим.

Пример содержимого файлов приведен ниже:

– *br0.network*

```
[Match]  
Name=br0
```

```
[Network]  
IPForward=ipv4  
IPMasquerade=yes  
DHCP=ipv4
```

– *br0.netdev*

```
[NetDev]  
Name=br0  
Kind=bridge
```

– *enp0s3.network*

```
[Match]  
Name=enp0s3
```

```
[Network]  
Bridge=br0
```

- 1.2.5.      # systemctl stop networking
- 1.2.6.      # systemctl disable networking
- 1.2.7.      # systemctl enable systemd-networkd
- 1.2.8.      # systemctl start systemd-networkd
- 1.2.9.      # reboot

Пример корректного результата установки приведен на Рис. 1.

```

dserenko@dserenko:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: enp1s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast master br0 state UP group default qlen 1000
    link/ether 1c:1b:0d:ed:e0:ac brd ff:ff:ff:ff:ff:ff
    valid_lft forever preferred_lft forever
3: br0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether ee:0f:8b:e7:4e:59 brd ff:ff:ff:ff:ff:ff
    inet 83.149.249.142/24 metric 1024 brd 83.149.249.255 scope global dynamic br0
        valid_lft 12900sec preferred_lft 12900sec
    inet6 fe80::ec0f:8bff:fee7:4e59/64 scope link
        valid_lft forever preferred_lft forever
4: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:54:27:bf:92 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
5: br-876c070c0573: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:31:18:e2:37 brd ff:ff:ff:ff:ff:ff
    inet 172.18.0.1/16 brd 172.18.255.255 scope global br-876c070c0573
        valid_lft forever preferred_lft forever
    inet6 fe80::42:31ff:fe18:e237/64 scope link
        valid_lft forever preferred_lft forever
9: veth41d65b3@if8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-876c070c0573 state UP group default
    link/ether da:b2:12:e4:78:f5 brd ff:ff:ff:ff:ff:ff link-netnsid 1
    inet6 fe80::d8b2:12ff:fee4:78f5/64 scope link
        valid_lft forever preferred_lft forever
11: veth0346aa1@if10: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master br-876c070c0573 state UP group default
    link/ether 6a:7f:d3:b4:ff:b1 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet6 fe80::687f:d3ff:feb4:ffb1/64 scope link
        valid_lft forever preferred_lft forever
24: ve-rec-services@if2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default qlen 1000
    link/ether aa:a4:d8:99:47:d5 brd ff:ff:ff:ff:ff:ff link-netnsid 2
    inet 169.254.237.187/16 metric 2048 brd 169.254.255.255 scope link ve-rec-services
        valid_lft forever preferred_lft forever
    inet 192.100.100.1/24 brd 192.100.100.255 scope global ve-rec-services
        valid_lft forever preferred_lft forever
    inet6 fe80::a8a4:d8ff:fe99:47d5/64 scope link
        valid_lft forever preferred_lft forever
dserenko@dserenko:~$

```

Рисунок 1 – Пример правильно настроенного моста сетевого интерфейса

1.3. Настройка сети выполняется в соответствии со следующим руководством

[<https://gist.github.com/lamafab/a626cfe7be5e3c1ecfe06251ddab4130>]:

1.3.1. \$ sudo su

1.3.2. # cp /lib/systemd/network/80-container-ve.network /etc/systemd/network/70-container-rec-services.network

1.3.3. # nano /etc/systemd/network/70-container-rec-services.network

Содержимое файла /etc/systemd/network/70-container-rec-services.network:

[Match]

Name=ve-rec-services

Driver=veth

[Network]

Address=192.100.100.1/24

LinkLocalAddressing=yes

DHCP=off

IPMasquerade=both

LLDP=yes

EmitLLDP=customer-bridge

IPv6SendRA=yes

Если на хосте используется фаерволл UFW и по умолчанию forwarding выключен, тогда лучше на хосте включить forwarding для ve-rec-services:

```
# ufw route allow in on ve-rec-services out on br0
```

```
# ufw route allow in on br0 out on ve-rec-services
```

1.4. Распаковка архива в папку /var/lib/machines/rec-services (в примере это сделано через создание символической ссылки, т.к., как правило, в разделе /var немного места)

```
$ sudo su
```

```
# cd /var/lib/machines
```

```
# mkdir <DATA_DIR>/rec-services
```

```
# ln -s <DATA_DIR>/rec-services rec-services
```

```
# tar -xf <PATH_TO>/rec-services.tar.gz -C /var/lib/machines/rec-services/
```

, где <DATA\_DIR> – директория с данными, в которой будет расположен распакованный образ контейнера;

<PATH\_TO> – директория, в которую был ранее загружен образ rec-services.

## 2. Запуск контейнера

```
$ sudo machinectl start rec-services
```

2.1. Проверка получения IP-адреса контейнером. На Рис.2 приведен пример корректного получения контейнером IP, на Рис.3 – ошибка получения IP. В таком случае следует перепроверить шаги 1.2 и 1.3 по настройке сети.

```
dserenko@dserenko:~$ machinectl
MACHINE      CLASS      SERVICE      OS      VERSION  ADDRESSES
rec-services  container  systemd-nspawn  debian  13       192.100.100.203...

1 machines listed.
dserenko@dserenko:~$
```

Рисунок 2 – Пример правильно настроенных сетевых настроек

```
dserenko@dserenko:~$ machinectl
MACHINE      CLASS      SERVICE      OS      VERSION  ADDRESSES
recservices  container  systemd-nspawn  debian  13       -

1 machines listed.
dserenko@dserenko:~$ machinectl
```

Рисунок 3 – Пример некорректно настроенных сетевых настроек. Контейнер не получил IP адрес

2.2. Доступ к контейнеру будет осуществляться по полученному IP (192.100.100.203 для данного примера).

2.2.1. \$ ssh -p2222 exuser@192.100.100.203

логин – <login>

пароль – <password>

### 3. Просмотр состояния системы (опционально)

При просмотре состояния системы в случае несоответствия фактического вывода подсистемы с ожидаемым можно перезапустить подсистему: `$ sudo systemctl restart <SERVICE_NAME>`, где `<SERVICE_NAME>` – название сервиса.

В рекомендательных сервисах используются следующие основные подсистемы. Они запускаются автоматически при старте контейнера:

- *exactus* – ядро Рекомендательных сервисов;
- *wordsdb* – сервис базы данных сущностей (слов и понятий) и их статистики встречаемости, используемый для анализа и классификации научно-технических и научно-образовательных документов по рубрикатору ГРНТИ;
- *postgresql* – СУБД для хранения метаданных документов и данных для веб-интерфейса;
- *lighttpd* – веб-сервер для обслуживания веб-интерфейса рекомендательных сервисов;
- *gunicorn* – сервер приложений для выполнения Python-кода рекомендательных сервисов;
- *gunicorn-grnti* – дополнительный экземпляр сервера приложений для обработки запросов, связанных с рубрикатором ГРНТИ;
- *sim-words* – сервис для интерактивного формирования и уточнения поисковых запросов с применением рекомендательной модели ассоциированных понятий предметных областей науки и техники. Работает совместно с сервисом *ling*.

Далее перечислены команды получения статусов основных сервисов и ожидаемый вывод.

#### 3.1. Центральное программное ядро. Пример ожидаемого вывода приведён на Рис. 4.

```
$ sudo systemctl status exactus
```



```

exuser@rec-services:/opt/services/src$ sudo systemctl status exactus
[sudo] пароль для exuser:
● exactus.service - Exactus system
   Loaded: loaded (/etc/init.d/exactus; enabled; preset: enabled)
   Active: active (running) since Mon 2025-12-29 15:48:58 MSK; 1h 50min ago
 Invocation: db88ac90cdc7446484e70dcd9d3bd31d
   Process: 501 ExecStart=/etc/init.d/exactus start (code=exited, status=0/SUCCESS)
  Tasks: 493 (limit: 629145)
  Memory: 2.6G (peak: 2.9G)
    CPU: 3min 5.926s
   CGroup: /system.slice/exactus.service
           └─521 /compiled/bin/icegridregistry --Ice.Config=/compiled/config/registry.cfg --pidfile /tmp/icegridr
           └─538 /compiled/bin/icegridnode --Ice.Config=/compiled/config/node.cfg --pidfile /tmp/icegridnode.pid
           └─553 /compiled/bin/Crawler --app_name=RecServ --config_file=/compiled/share/config/empty.cfg --verbo
           └─558 /compiled/bin/DocAnalyzer.real --app_name=RecServ --config_file=/compiled/config/doc_analyzer.cf
           └─561 python /compiled/bin/ExDocConverter --app_name=RecServ --log_file=/data/RecServ/DocConverter_1/D
           └─565 /compiled/bin/Frontier --app_name=RecServ --config_file=/compiled/share/config/empty.cfg --verbo
           └─568 /compiled/bin/IntelProc.real --app_name=RecServ --config_file=/compiled/share/config/empty.cfg --
           └─569 /compiled/bin/Index --app_name=RecServ --cfg_dir=/compiled/config/colls.d --config_file=/compil
           └─571 python /compiled/bin/ExLockService --app_name=RecServ --log_file=/data/RecServ/LockService_1/Loc
           └─572 python /compiled/bin/ExLogger --app_name=RecServ --log_file=/data/RecServ/Logger_1/Logger_1.log
           └─573 python /compiled/bin/ExMetricsCollector --app_name=RecServ --log_file=/data/RecServ/MetricsColle
           └─574 python /compiled/bin/ExMetricsProxy --app_name=RecServ --log_file=/data/RecServ/MetricsProxy/Met
           └─575 /compiled/bin/Queue --app_name=RecServ --config_file=/compiled/share/config/empty.cfg --log_file
           └─576 /compiled/bin/Storage --app_name=RecServ --config_file=/compiled/share/config/empty.cfg --log_fi
           └─577 python /compiled/bin/SystemCtl --app_name=RecServ --log_file=/data/RecServ/SystemCtl-rec-service
           └─578 /compiled/bin/TextAnalyzer --app_name=RecServ --config_file=/compiled/share/config/empty.cfg --T
           └─959 python /compiled/bin/ExMetricsCollector --app_name=RecServ --log_file=/data/RecServ/MetricsColle
           └─960 python /compiled/bin/ExMetricsProxy --app_name=RecServ --log_file=/data/RecServ/MetricsProxy/Met
           └─961 python /compiled/bin/ExMetricsProxy --app_name=RecServ --log_file=/data/RecServ/MetricsProxy/Met

дек 29 15:48:57 rec-services systemd[1]: Starting exactus.service - Exactus system...
дек 29 15:48:58 rec-services exactus[501]: Registry config is found! Start it!:Node config is found! Start it!:
дек 29 15:48:58 rec-services systemd[1]: Started exactus.service - Exactus system.
lines 1-32/32 (END)

```

Рисунок 4 – Статус центрального программного ядра

3.2. База данных сущностей и их статистика встречаемости. Пример ожидаемого вывода приведён на Рис. 5.

\$ sudo systemctl status wordsdb

```

exuser@rec-services:/opt/services/src$ sudo systemctl status wordsdb
● wordsdb.service - wordsdb service
   Loaded: loaded (/etc/systemd/system/wordsdb.service; enabled; preset: enabled)
   Active: active (running) since Mon 2025-12-29 15:48:50 MSK; 1h 52min ago
 Invocation: 9fc0d2c85298437bb2c2cdfa98f07899
   Main PID: 101 (wordsdb)
  Tasks: 49 (limit: 629145)
  Memory: 211.7M (peak: 212.2M)
    CPU: 39.837s
   CGroup: /system.slice/wordsdb.service
           └─101 /opt/grnt_service/thirdparty/wordsdb/build/src/wordsdb -c /opt/grnt_service/thirdparty/wordsdb/

дек 29 15:48:50 rec-services systemd[1]: Started wordsdb.service - wordsdb service.
дек 29 15:48:50 rec-services wordsdb[101]: Reading wdb from extcl.wdb
дек 29 15:49:31 rec-services wordsdb[101]: Wdb read from extcl.wdb
дек 29 15:49:31 rec-services wordsdb[101]: Starting wordsdb_app on 8550: true
exuser@rec-services:/opt/services/src$

```

Рисунок 5 – Статус подсистемы wordsdb

3.3. СУБД postgresql. Пример ожидаемого вывода приведён на Рис. 6.

\$ sudo systemctl status postgresql  
\$ sudo pg\_lsclusters

```

exuser@rec-services:/opt/services/src$ sudo systemctl status postgresql
● postgresql.service - PostgreSQL RDBMS
   Loaded: loaded (/usr/lib/systemd/system/postgresql.service; enabled; preset: enabled)
   Active: active (exited) since Mon 2025-12-29 15:48:53 MSK; 1h 53min ago
  Invocation: e3e4793c7b1c46cf862e31f3fed99ed4
    Process: 321 ExecStart=/bin/true (code=exited, status=0/SUCCESS)
   Main PID: 321 (code=exited, status=0/SUCCESS)
    Mem peak: 1.7M
       CPU: 30ms

дек 29 15:48:53 rec-services systemd[1]: Starting postgresql.service - PostgreSQL RDBMS...
дек 29 15:48:53 rec-services systemd[1]: Finished postgresql.service - PostgreSQL RDBMS.
exuser@rec-services:/opt/services/src$ sudo pg_lsclusters
Ver Cluster Port Status Owner    Data directory          Log file
16  main      5432 online postgres /var/lib/postgresql/16/main /var/log/postgresql/postgresql-16-main.log
exuser@rec-services:/opt/services/src$

```

Рисунок 6 – Статус СУБД postgresql

### 3.4. Веб-сервер lighttpd. Пример ожидаемого вывода приведён на Рис. 8.

\$ sudo systemctl status lighttpd

```

exuser@rec-services:/opt/services/src$ sudo systemctl status lighttpd
● lighttpd.service - Lighttpd Daemon
   Loaded: loaded (/usr/lib/systemd/system/lighttpd.service; enabled; preset: enabled)
   Drop-In: /etc/systemd/system/lighttpd.service.d
            └─override.conf
   Active: active (running) since Mon 2025-12-29 15:48:57 MSK; 1h 55min ago
  Invocation: 2a9c2a4bca76480fb502ef29a5d4a6f2
     Docs: man:lighttpd
           https://wiki.lighttpd.net
    Process: 482 ExecStartPre=/usr/sbin/lighttpd -tt -f /etc/lighttpd/lighttpd.conf (code=exited, status=0/SUCCESS)
   Main PID: 500 (lighttpd)
      Tasks: 1 (limit: 629145)
     Memory: 2.7M (peak: 4.1M)
        CPU: 901ms
    CGroup: /system.slice/lighttpd.service
            └─500 /usr/sbin/lighttpd -D -f /etc/lighttpd/lighttpd.conf

дек 29 15:48:57 rec-services systemd[1]: Starting lighttpd.service - Lighttpd Daemon...
дек 29 15:48:57 rec-services lighttpd[482]: 2025-12-29 15:48:57: (mod_deflate.c.741) DEPRECATED: compress.filetype re
дек 29 15:48:57 rec-services lighttpd[482]: 2025-12-29 15:48:57: (mod_deflate.c.804) DEPRECATED: compress.cache-dir
дек 29 15:48:57 rec-services lighttpd[482]: 2025-12-29 15:48:57: (configfile.c.1400) WARNING: unknown config-key: in
дек 29 15:48:57 rec-services systemd[1]: Started lighttpd.service - Lighttpd Daemon.

```

Рисунок 8 – Статус веб-сервера lighttpd

### 3.5. Сервер приложений gunicorn. Пример ожидаемого вывода приведён на Рис. 9.

\$ sudo systemctl status gunicorn

```

exuser@rec-services:/opt/services/src$ sudo systemctl status gunicorn
● gunicorn.service - gunicorn daemon
   Loaded: loaded (/etc/systemd/system/gunicorn.service; enabled; preset: enabled)
   Active: active (running) since Mon 2025-12-29 16:28:17 MSK; 1h 18min ago
  Invocation: baf98e45175b476cae8f8fdab02d0268
 TriggeredBy: ● gunicorn.socket
   Main PID: 4669 (gunicorn)
      Tasks: 11 (limit: 629145)
     Memory: 189.3M (peak: 190M)
        CPU: 50.075s
    CGroup: /system.slice/gunicorn.service
            └─4669 /opt/services/.venv/bin/python3 /opt/services/.venv/bin/gunicorn --workers 5 -k uvicorn.workers.Uvic
               4670 /opt/services/.venv/bin/python3 /opt/services/.venv/bin/gunicorn --workers 5 -k uvicorn.workers.Uvic
               4671 /opt/services/.venv/bin/python3 /opt/services/.venv/bin/gunicorn --workers 5 -k uvicorn.workers.Uvic
               4672 /opt/services/.venv/bin/python3 /opt/services/.venv/bin/gunicorn --workers 5 -k uvicorn.workers.Uvic
               4673 /opt/services/.venv/bin/python3 /opt/services/.venv/bin/gunicorn --workers 5 -k uvicorn.workers.Uvic
               4674 /opt/services/.venv/bin/python3 /opt/services/.venv/bin/gunicorn --workers 5 -k uvicorn.workers.Uvic

дек 29 16:28:20 rec-services gunicorn[4671]: [2025-12-29 16:28:20 +0300] [4671] [INFO] Application startup complete.
дек 29 16:28:20 rec-services gunicorn[4672]: [2025-12-29 16:28:20 +0300] [4672] [INFO] Started server process [4672]
дек 29 16:28:20 rec-services gunicorn[4672]: [2025-12-29 16:28:20 +0300] [4672] [INFO] Waiting for application startup.
дек 29 16:28:20 rec-services gunicorn[4672]: [2025-12-29 16:28:20 +0300] [4672] [INFO] Application startup complete.
дек 29 16:28:20 rec-services gunicorn[4673]: [2025-12-29 16:28:20 +0300] [4673] [INFO] Started server process [4673]
дек 29 16:28:20 rec-services gunicorn[4673]: [2025-12-29 16:28:20 +0300] [4673] [INFO] Waiting for application startup.
дек 29 16:28:20 rec-services gunicorn[4673]: [2025-12-29 16:28:20 +0300] [4673] [INFO] Application startup complete.
дек 29 16:28:20 rec-services gunicorn[4674]: [2025-12-29 16:28:20 +0300] [4674] [INFO] Started server process [4674]
дек 29 16:28:20 rec-services gunicorn[4674]: [2025-12-29 16:28:20 +0300] [4674] [INFO] Waiting for application startup.
дек 29 16:28:20 rec-services gunicorn[4674]: [2025-12-29 16:28:20 +0300] [4674] [INFO] Application startup complete.

```

Рисунок 9 – Статус сервера приложений gunicorn

### 3.6. Сервер приложений gunicorn-grnti. Пример ожидаемого вывода приведён на Рис. 10.

\$ sudo systemctl status gunicorn-grnti

```
exuser@rec-services:/opt/services/src$ sudo systemctl status gunicorn-grnti
● gunicorn-grnti.service - gunicorn daemon for GRNTI classifier
   Loaded: loaded (/etc/systemd/system/gunicorn-grnti.service; enabled; preset: enabled)
   Active: active (running) since Mon 2025-12-29 16:28:37 MSK; 1h 20min ago
     Invocation: bee0ab5a3645467a89ea917a5f0ee235
   TriggeredBy: ● gunicorn-grnti.socket
    Main PID: 4687 (gunicorn)
       Tasks: 97 (limit: 629145)
      Memory: 53G (peak: 53.1G)
         CPU: 3min 4.545s
    CGroup: /system.slice/gunicorn-grnti.service
            └─4687 /opt/grnt_service/venv/bin/python3 /opt/grnt_service/venv/bin/gunicorn --workers 1 -k uvicorn.workers.
              4688 /opt/grnt_service/venv/bin/python3 /opt/grnt_service/venv/bin/gunicorn --workers 1 -k uvicorn.workers.

дек 29 16:28:38 rec-services gunicorn[4688]: [2025-12-29 16:28:38 +0300] [4688] [INFO] Booting worker with pid: 4688
дек 29 16:28:49 rec-services gunicorn[4688]: 2025-12-29 16:28:49,530 [4688] INFO: faiss.loader: Loading faiss with AVX512
дек 29 16:28:49 rec-services gunicorn[4688]: 2025-12-29 16:28:49,663 [4688] INFO: faiss.loader: Successfully loaded faiss
дек 29 16:29:18 rec-services gunicorn[4688]: [2025-12-29 16:29:18 +0300] [4688] [INFO] Started server process [4688]
дек 29 16:29:18 rec-services gunicorn[4688]: [2025-12-29 16:29:18 +0300] [4688] [INFO] Waiting for application startup.
дек 29 16:29:18 rec-services gunicorn[4688]: 2025-12-29 16:29:18,497 [4688] INFO: root: Reading index from data/extcl_workd
дек 29 16:31:29 rec-services gunicorn[4688]: 2025-12-29 16:31:29,734 [4688] INFO: root: Done with reading index from data/
дек 29 16:31:29 rec-services gunicorn[4688]: 2025-12-29 16:31:29,735 [4688] INFO: root: Connecting to db data/extcl_workd
дек 29 16:31:29 rec-services gunicorn[4688]: 2025-12-29 16:31:29,737 [4688] INFO: root: Connected to db data/extcl_workd
дек 29 16:31:29 rec-services gunicorn[4688]: [2025-12-29 16:31:29 +0300] [4688] [INFO] Application startup complete.
```

Рисунок 10 – Статус сервера приложений gunicorn-grnti

### 3.7. Сервис для интерактивного формирования и уточнения поисковых запросов sim-words. Пример ожидаемого вывода приведён на Рис. 11.

\$ systemctl status --user sim-words.v0\_2.monitor.service

```
exuser@rec-services:/opt/services/src$ systemctl status --user sim-words.v0_2.monitor.service
● sim-words.v0_2.monitor.service
   Loaded: loaded (/home/exuser/.config/systemd/user/sim-words.v0_2.monitor.service; static)
   Active: active (running) since Mon 2025-12-29 15:48:51 MSK; 2h 2min ago
     Invocation: 02e13d44bc06492784fe818d43fdd3e7
    Main PID: 290 (.pplmonitor-wra)
         IO: 288K read, 0B written
       Tasks: 8 (limit: 629145)
      Memory: 39.2M (peak: 39.8M)
         CPU: 7.984s
    CGroup: /user.slice/user-1000.slice/user@1000.service/app.slice/app-sim_words.v0_2.slice/sim
            └─290 /nix/store/sd81bvmch7njdpwx3lkjslixcbj5mivz-python3-3.13.4/bin/python3.13 /nix

дек 29 17:41:48 rec-services pplmonitor[290]: INFO JobRunner - job_finished, name=health_check, t
дек 29 17:41:48 rec-services pplmonitor[290]: INFO JobRunner - next_time, name=health_check, iter
дек 29 17:45:18 rec-services pplmonitor[290]: INFO JobRunner - run_job, name=health_check
дек 29 17:45:23 rec-services pplmonitor[290]: INFO HealthCheck - done, str_cnt=100, ids_cnt=100,
дек 29 17:45:23 rec-services pplmonitor[290]: INFO JobRunner - job_finished, name=health_check, t
дек 29 17:45:23 rec-services pplmonitor[290]: INFO JobRunner - next_time, name=health_check, iter
дек 29 17:48:53 rec-services pplmonitor[290]: INFO JobRunner - run_job, name=health_check
дек 29 17:49:04 rec-services pplmonitor[290]: INFO HealthCheck - done, str_cnt=100, ids_cnt=100,
дек 29 17:49:04 rec-services pplmonitor[290]: INFO JobRunner - job_finished, name=health_check, t
дек 29 17:49:04 rec-services pplmonitor[290]: INFO JobRunner - next_time, name=health_check, iter
lines 1-22/22 (END)
```

Рисунок 11 – Статус сервиса sim-words

## 4. Доступ к программному интерфейсу экземпляра «Рекомендательные сервисы поддержки научно-технической и научно-образовательной деятельности».

Для доступа к установленному экземпляру «Рекомендательных сервисов поддержки научно-технической и научно-образовательной деятельности» необходимо открыть веб-браузер и перейти по ссылке <http://192.100.100.203:8080/docs> (для данного примера). При переходе на указанный адрес будет открыта начальная страница

инструментария Swagger (**Ошибка! Источник ссылки не найден.**12), на которой будут доступны следующие функции:

- функция рекомендации изданий для опубликования научных результатов на основе анализа полного текста препринта научной статьи;
- функция подбора экспертов / рецензентов по заданному полным текстом объекту экспертизы;
- функция подбора релевантных технологий по технологическим запросам интерактивное формирование и уточнение поисковых запросов с применением рекомендательной модели ассоциированных понятий предметных областей науки и техники;
- функция интерактивного формирования и уточнение поисковых запросов с применением рекомендательной модели ассоциированных понятий предметных областей науки и техники;
- функция автоматической классификации научно-технических и научно-образовательных документов по рубриктору ГРНТИ.

(см. детальное описание функций в Руководстве пользователя «Рекомендательные сервисы поддержки научно-технической и научно-образовательной деятельности»).

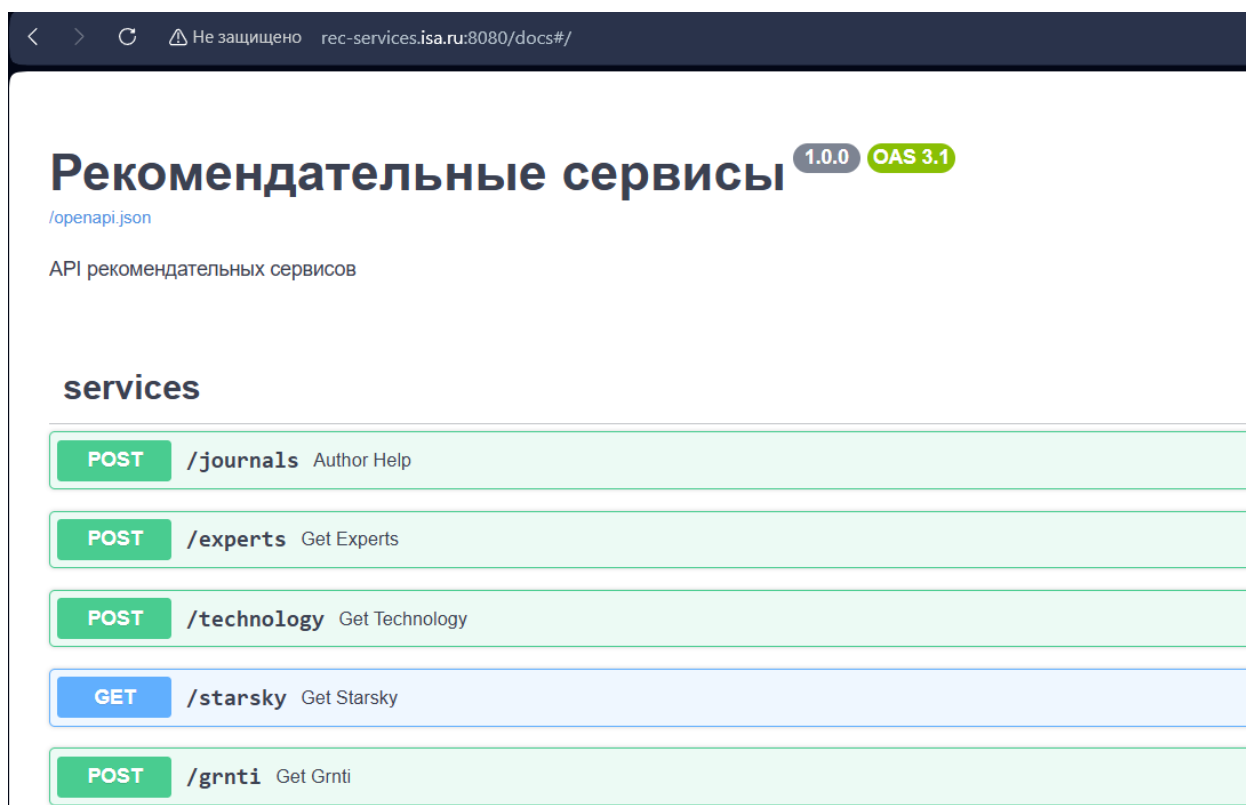


Рисунок 12 – Основная страница Рекомендательных сервисов