Сибирь.Аудит

Руководство по установке и эксплуатации

Экземпляр 1.2.0. Редакция 4 от 30 октября 2026 года.

Правообладатель: ООО «Финансовый искусственный интеллект». ИНН 5402076457, ОГРН 1235400001870.

Архив: sibir-audit-1.2.0.zip. Размер: 108 260 400 байт. SHA-256:

```
3a0913d4c46a482730bcd19689e1c4756692617e8d59f4dfefcc14ba3752e8f6
```

Архив полностью прочитан 29 октября по точной ссылке из предоставленной карточки заявления № 384870. Размер и контрольная сумма совпали с проверенным экземпляром. В этой редакции исправлено представление команд для копирования из PDF, программный архив не изменялся.

Экземпляр анализирует исходный HTML одной страницы. Браузерный рендеринг и построение карты перенаправлений всего сайта не выполняются. Семантическая модель вызывается отдельными функциями библиотеки и автоматически в HTTP-проверках не применяется. Подробные границы приведены в разделе 6.6.1 и отдельном уточнении функционального объема.

Внутренние метки сохранены: pyproject.toml указывает 1.0.1, sibir_audit.__version__ указывает 0.0.0. Поставка 1.2.0 идентифицируется именем, размером и SHA-256 архива.

Исполняемые примеры набраны без автоматического переноса строк. PowerShell использует продолжение обратной кавычкой, POSIX-оболочка - обратным слешем. Python-примеры сохраняются в отдельные файлы. Вывод программы и исторические журналы не являются командами для ввода.

Результаты функционального прогона от 28 октября сохранены в разделе 9. Проверка представления команд 30 октября не является новым функциональным прогоном. Linux, остановка под нагрузкой и полная сетевая изоляция этим документом не принимаются.

Для установки требуются источник пакетов и зависимости. Публичная документация: https://siberia-ai.ru/docs.html. Поддержка: info@siberia-ai.ru.

Информация, необходимая для установки экземпляра

Экземпляр поставляется архивом по лицензионному договору вместе с контрольной суммой SHA-256. Обученная модель входит в архив, отдельно ее скачивать не требуется.

1. Состав экземпляра

Архив дистрибутива

Значение: sibir-audit-1.2.0.zip

Размер архива, байт

Значение: 108 260 400

Записей в архиве

Значение: 73

Файлов исходного текста

Значение: 70

Строк исходного текста

Значение: 15 425

Контрольная сумма SHA-256

Значение: 3a0913d4c46a482730bcd19689e1c4756692617e8d59f4dfefcc14ba3752e8f6

В архив входит исходный текст программы на языке Python и каталог sibir_audit/models с обученной моделью представления текста и словарем токенизатора. Модель занимает 117 142 976 байт и загружается программой с файловой системы. Отдельно скачивать ее не требуется.

2. Проверка экземпляра перед установкой

Перед распаковкой сверьте контрольную сумму полученного архива с указанной в разделе 1. Несовпадение означает, что файл поврежден или подменен, и установку продолжать нельзя.

```
sha256sum sibir-audit-1.2.0.zip
```

В операционной системе Windows та же проверка выполняется командой:

```
certutil -hashfile sibir-audit-1.2.0.zip SHA256
```

3. Требования к среде

Операционная система

Значение: Windows Server 2022 проверена в новом прогоне. Linux заявлена в исходной документации, здесь не испытана

Язык исполнения

Значение: Python версии не ниже 3.12

Оперативная память

Значение: не менее 2 ГБ

Свободное место

Значение: не менее 1 ГБ

Сетевой доступ

Значение: К источнику пакетов при установке, к проверяемым ресурсам при аудите

Средства виртуализации, контейнеризации, отдельный веб-сервер и система управления базами данных не требуются. Связь с правообладателем для работы программы не нужна.

4. Установка

4.1. Распаковать архив в каталог эксплуатанта.

```
unzip sibir-audit-1.2.0.zip -d sibir-audit
```

4.2. Создать изолированное окружение и установить зависимости, перечисленные в файле pyproject.toml.

```
python3 -m venv .venv && . .venv/bin/activate
pip install -e sibir-audit
```

Пример для PowerShell в Windows. Сначала распакуйте архив в каталог sibir-audit. Команда python в этом примере должна запускать Python 3.12. Имена рабочих каталогов могут отличаться от использованных в протоколе.

```
Expand-Archive -LiteralPath .\sibir-audit-1.2.0.zip -DestinationPath .\sibir-audit
python -m venv .venv
.\.venv\Scripts\python.exe -m pip --isolated install `
    --index-url https://pypi.org/simple --no-cache-dir `
    -e .\sibir-audit
```

Активация окружения не обязательна, если запускать его python.exe по полному относительному пути. Для семантических функций с комплектной моделью дополнительно установите onnxruntime в это же окружение. Без него доступен лексический режим.

```
.\.venv\Scripts\python.exe -m pip --isolated install `
    --index-url https://pypi.org/simple --no-cache-dir onnxruntime
.\.venv\Scripts\python.exe -m uvicorn sibir_audit.public_audit:app `
    --host 127.0.0.1 --port 8080
```

4.3. Убедиться, что каталог sibir_audit/models распакован вместе с исходным текстом. Без него программа продолжит работать, но семантические оценки будут считаться запасным лексическим способом, о чем программа сообщит признаком основания оценки.

5. Запуск

Программный интерфейс поднимается штатным сервером приложений:

```
uvicorn sibir_audit.public_audit:app --host 127.0.0.1 --port 8080
```

Проверка одной страницы выполняется обращением к интерфейсу:

```
curl -X POST http://localhost:8080/check \
  -H "Content-Type: application/json" \
  -d '{"url": "https://example.org/"}'
```

5.1. Семантические оценки

Подсистема семантических оценок находится в модуле `sibir_audit.modules.semantika` и дает три оценки: `otvet_po_zagolovku` (отвечает ли начало раздела на вопрос заголовка), `samopodobie` (повторяет ли документ сам себя) и `novizna` (насколько текст отличается от эталонных текстов). Каждая возвращает число от 0 до 1 и признак основания: `model`, если число получено нейросетевой моделью, или `leksika`, если запасным способом по пересечению слов.

Подсистема вызывается отдельно. HTTP-служба из пункта 5 и функция технического аудита страницы ее не вызывают, на их результат она не влияет. Модель передается в оценку явно, объектом `Embedder.po_umolchaniyu()`. Он берет файлы `rubert-tiny2.onnx` и `vocab.txt` из каталога `sibir_audit/models` внутри установленного пакета. Без этого аргумента оценка считается лексическим способом.

Для работы модели нужна среда выполнения onnxruntime. В перечень зависимостей pyproject.toml она не входит и устанавливается отдельно, в то же окружение:

```
pip install onnxruntime
```

Проверка на текстах из обоснования отнесения программы к сфере искусственного интеллекта:

Сохраните следующий код в файл audit_example_9.py и выполните его интерпретатором окружения программы. Файл также приведен в приложенных примерах команд.

```
from sibir_audit.modules.semantika import Embedder, otvet_po_zagolovku
e = Embedder.po_umolchaniyu()
z = 'Как заменить счетчик воды'
for t in ('Прибор учета меняют так', 'История московского метрополитена кратко'):
    print(otvet_po_zagolovku(z, t, e))
    print(otvet_po_zagolovku(z, t))
```

```
python audit_example_9.py
```

Вывод 28 сентября 2026 года, onnxruntime 1.30.0, Python 3.12.10:

```
Ocenka(znachenie=0.8174937137223827, osnovanie='model', poyasnenie='')
Ocenka(znachenie=0.0, osnovanie='leksika', poyasnenie='модель недоступна, считано пересечением слов')
Ocenka(znachenie=0.6988008865467242, osnovanie='model', poyasnenie='')
Ocenka(znachenie=0.0, osnovanie='leksika', poyasnenie='модель недоступна, считано пересечением слов')
```

Первая и третья строки получены моделью: 0,817 для ответа, сформулированного другими словами, и 0,699 для постороннего текста. Вторая и четвертая строки получены без модели и равны нулю, потому что общих слов у этих текстов нет. Близость приводится к отрезку от 0 до 1 сдвигом косинуса, поэтому даже посторонний текст получает около 0,7. Сравнивать следует значения между собой, а не с серединой отрезка.

Те же оценки на разделах этой страницы документации. Эталоном для третьей оценки служит документация «Сибирь.Диалога»:

Сохраните следующий код в файл audit_example_11.py и выполните его интерпретатором окружения программы. Файл также приведен в приложенных примерах команд.

```
from bs4 import BeautifulSoup
from sibir_audit.modules.pr.safe_fetch import safe_fetch_page
from sibir_audit.modules.semantika import (
    Embedder,
    otvet_po_zagolovku,
    samopodobie,
    novizna,
)
e = Embedder.po_umolchaniyu()
def razdely(url):
    return BeautifulSoup(safe_fetch_page(url).body, 'lxml').select('main section')
docs = razdely('https://siberia-ai.ru/docs')
for r in docs:
    print(r.get('id'), otvet_po_zagolovku(r.h2.get_text(), r.p.get_text(), e))
teksty = [r.get_text(' ', strip=True) for r in docs]
print('samopodobie', samopodobie(teksty, e))
etalon = ' '.join(
    r.get_text(' ', strip=True)
    for r in razdely('https://siberia-ai.ru/dialog-docs')
)
print('novizna', novizna(' '.join(teksty), [etalon], e))
```

```
python audit_example_11.py
```

```
functional Ocenka(znachenie=0.6930483691270071, osnovanie='model', poyasnenie='')
install Ocenka(znachenie=0.7802163705151968, osnovanie='model', poyasnenie='')
operation Ocenka(znachenie=0.7902162853197296, osnovanie='model', poyasnenie='')
lifecycle Ocenka(znachenie=0.7597422891267605, osnovanie='model', poyasnenie='')
samopodobie Ocenka(znachenie=0.9727690332105996, osnovanie='model', poyasnenie='')
novizna Ocenka(znachenie=0.05178191592006476, osnovanie='model', poyasnenie='')
```

Модель читает каждый раздел не дальше первых 256 токенов, поэтому для длинных разделов оценка относится к их началу.

Без onnxruntime программа не прерывает работу. Каждая оценка возвращается лексическим способом с признаком `osnovanie='leksika'` и пояснением «модель недоступна, считано пересечением слов». Первая команда без onnxruntime печатает четыре одинаковые строки:

```
Ocenka(znachenie=0.0, osnovanie='leksika', poyasnenie='модель недоступна, считано пересечением слов')
```

Команды выполнены правообладателем 28 сентября 2026 года в Windows Server 2022, в оболочках bash и PowerShell. В Linux не проверялось.

Информация, необходимая для эксплуатации экземпляра

Порядок работы, прекращение работы и техническая поддержка.

6. Эксплуатация

6.1. Результат проверки возвращается в виде структуры с перечнем выявленных дефектов, их значимостью, сведениями о происхождении каждого вывода и рекомендацией по устранению.

6.2. Пороги принятия решений в программу не заложены и настраиваются эксплуатирующей организацией под свои задачи.

6.3. Журнал работы ведется в стандартный поток вывода и может быть перенаправлен средствами операционной системы.

6.4. Обновление выполняется заменой каталога программы на новую редакцию с предварительной проверкой контрольной суммы по разделу 2.

6.5. Загрузка и запуск

Программа работает в одном из двух режимов: как HTTP-служба (программный интерфейс) или как библиотека Python, которую вызывает программа эксплуатанта. Оба режима используют окружение, созданное по пункту 4.2. Перед запуском окружение активируется:

```
. .venv/bin/activate
```

В Linux не проверялось. В Windows выполнено `. .venv/Scripts/activate` в bash и `.venv\Scripts\Activate.ps1` в PowerShell.

Служба запускается из того же окружения:

```
uvicorn sibir_audit.public_audit:app --host 127.0.0.1 --port 8080
```

Параметр `--host` задает адрес, на котором служба принимает обращения, `--port` задает порт. Параметр `--no-access-log` отключает строки журнала о каждом обращении (пункт 6.11). Проверки подлинности обращений служба не выполняет, поэтому ее поднимают на адресе 127.0.0.1 или во внутренней сети эксплуатанта.

Признак успешного старта: последняя из четырех строк, «Uvicorn running on», после нее служба принимает обращения.

```
INFO:     Started server process [5336]
INFO:     Waiting for application startup.
INFO:     Application startup complete.
INFO:     Uvicorn running on http://127.0.0.1:8080 (Press CTRL+C to quit)
```

Готовность проверяется обращением к пути `/healthz`:

```
curl http://127.0.0.1:8080/healthz
```

```
{"ok":true,"service":"public-audit"}
```

Если порт занят, строки «Uvicorn running on» не будет. Вместо нее служба печатает сообщение об ошибке и завершается с кодом 3:

```
ERROR:    [Errno 10048] error while attempting to bind on address ('127.0.0.1', 8080): [winerror 10048]
only one usage of each socket address (protocol/network address/port) is normally permitted
```

Текст ошибки приведен для Windows. В Linux номер и текст ошибки системные, начало строки то же, не проверялось.

6.6. Команды программного интерфейса

Все команды, кроме `/healthz` и `/openapi.json`, принимаются методом POST. Тело запроса передается в формате JSON с заголовком `Content-Type: application/json`.

GET /healthz

Назначение: Признак жизни службы. Отвечает, пока процесс работает, проверяемые ресурсы не запрашивает

POST /check

Назначение: Полный отчет по одной странице: все находки с доказательствами и сведениями о происхождении, пройденные проверки, версия набора правил, метод расчета балла, результат проверки robots.txt

POST /audit

Назначение: Сокращенный отчет по одной странице: балл, число находок по значимости, до трех приоритетных исправлений значимости CRITICAL, HIGH и MEDIUM, готовность страницы к системам поиска на основе искусственного интеллекта

POST /ai-access

Назначение: Доступ поисковых роботов систем искусственного интеллекта к сайту по его файлу robots.txt и наличие файла llms.txt

POST /llms-txt

Назначение: Формирование файла llms.txt по карте сайта и заголовку главной страницы

POST /site

Назначение: Обзор сайта целиком. В этой сборке не поддерживается, ответ 501

GET /openapi.json

Назначение: Машиночитаемое описание интерфейса в формате OpenAPI 3.1

Поля запроса у всех команд POST одинаковые:

url

Значение: Обязательное. Адрес страницы, строка длиной от 1 до 2048 знаков. Принимаются схемы http и https, адрес без схемы дополняется до https://. Для /ai-access и /llms-txt используется только схема и имя сайта

lang

Значение: Необязательное. Язык отчета: `ru`, `en` или `zh`. Пустое значение означает русский. Неизвестное значение заменяется на `en`, и ответ сообщает об этом полем `lang_fallback`

Команда /site дополнительно принимает поля `max_pages`, `max_depth` и `exclude`, но в этой сборке отвечает отказом до их обработки.

6.6.1. Границы функций этого экземпляра

Проверяется исходный HTML одной страницы. Скрипты страницы в браузере не выполняются. Проверка viewport является проверкой метатега, а не визуальной проверкой мобильной верстки. Анализируются заголовки, метаданные, содержание, отдельные признаки структурированных данных и доступности для поисковых роботов. Это не полный валидатор Schema.org и не прогноз позиций или цитирования.

Цепочка перенаправлений относится к одному запрошенному адресу. Обход всего сайта и построение карты его перенаправлений в этой сборке недоступны, /site возвращает 501. Команда /llms-txt формирует текст по карте сайта и заголовку, она не анализирует содержание всех страниц. Семантическая ONNX-модель вызывается отдельными функциями библиотеки и не применяется автоматически в /check, /audit или audit_url.

Число проверок в ответе зависит от страницы, языка и набора применимых правил. В проверенном примере получен 31 результат. Это не постоянное значение для каждого запроса.

6.7. Полный отчет: POST /check

```
curl -X POST http://127.0.0.1:8080/check \
  -H "Content-Type: application/json" \
  -d '{"url": "https://siberia-ai.ru/"}'
```

Ответ программы 29 сентября 2026 года, код 200. Сокращены два перечня: из 30 пройденных проверок показаны две, из семи ограничений одно. Значения остальных полей приведены без изменений, изменены только переносы строк.


```
{
 "ok": true,
 "lang": "ru",
 "lang_requested": null,
 "lang_fallback": false,
 "requested_url": "https://siberia-ai.ru/",
 "final_url": "https://siberia-ai.ru/",
 "redirect_chain": [],
 "http_status": 200,
 "checked_at": "2026-09-29T00:35:20+00:00",
 "elapsed_ms": 2406,
 "rules_version": "2026.08.8",
 "score": 98,
 "score_method": {
  "formula": "100 минус взвешенный вычет за каждую находку; недоступность страницы (не открылась, 4xx, 5xx) обнуляет балл.",
  "weights": {"CRITICAL": 40, "HIGH": 15, "MEDIUM": 7, "LOW": 2, "INFO": 0},
  "note": "Это ориентир по данному набору правил, а не метрика поисковой системы и не предсказание позиций. В балле учитываются все находки, которые набор умеет выдавать, поэтому технически исправная страница может получить низкий балл из-за формулировок."
 },
 "needs_render": false,
 "robots": {"checked": true, "allowed": true, "robots_url": "https://siberia-ai.ru/robots.txt"},
 "counts_by_severity": {"LOW": 1},
 "findings_total": 1,
 "findings": [
  {
   "code": "schema_sameas_no_kg_anchor",
   "category": "GEO",
   "severity": "LOW",
   "title": "sameAs ведет только в соцсети, без якоря на граф знаний",
   "recommendation": "Добавьте в sameAs хотя бы одну ссылку на запись в публичной базе знаний - Викиданных (wikidata.org) или Википедии.",
   "evidence": {

"organizations": ["ООО «ФИИ»"],
    "org_nodes_without_kg_anchor": 1,
    "provenance": {
     "status": "verified",
     "sources": [{"source_id": null, "tier": 5, "type": "self-measured"}],
     "max_tier": 5,
     "method": "self-measured"
    }
   },
   "effort": 1,
   "impact": 1,
   "why": "Ссылки на соцсети подтверждают только присутствие в соцсетях; сущность организации ИИ-системы сверяют по графу знаний, и без такого якоря разметка ее не опознает.",
   "title_en": "Organization sameAs links to social profiles, not a knowledge graph",
   "recommendation_en": "The Organization's sameAs points only to its own social/brand profiles. Add at least one public knowledge-graph entry - Wikidata or Wikipedia - so AI engines can resolve the publisher to a known entity, not just corroborate its social presence.",
   "why_en": "sameAs that lists only social profiles proves the brand has a presence, not which entity it is; an engine resolves the publisher against a knowledge graph, and without a Wikidata/Wikipedia link there is nothing to resolve it to."
  }
```

```
 ],
 "passed_checks": [
  {"code": "reachability", "title": "Доступность и код ответа", "title_en": "Reachability & HTTP status"},
  {"code": "https", "title": "HTTPS и смешанное содержимое", "title_en": "HTTPS & mixed content"}
 ],
 "checks_total": 31,
 "not_checked": 0,
 "limitations": [
  "Проверен один снимок одной страницы, скачанной с наших серверов, - это не весь сайт и не то, что уже
находится в индексе поисковой системы."
 ]
}
```


ok

Значение: true, если отчет сформирован

lang, lang_requested, lang_fallback

Значение: Язык отчета, язык из запроса и признак замены неизвестного языка на английский

requested_url, final_url, redirect_chain

Значение: Запрошенный адрес, адрес после перенаправлений и цепочка перенаправлений

http_status

Значение: Код ответа проверяемого сервера

checked_at, elapsed_ms

Значение: Время проверки (UTC) и ее длительность в миллисекундах

rules_version

Значение: Версия набора правил

score, score_method

Значение: Итоговый балл от 0 до 100 и формула его расчета с весами значимости

needs_render

Значение: true, если сервер отдал почти пустую страницу, которая формируется сценариями в браузере. Тогда балл не считается (null)

robots

Значение: Проверялся ли robots.txt, разрешает ли он загрузку страницы и его адрес

counts_by_severity, findings_total

Значение: Число находок по значимости и всего

findings

Значение: Находки: код правила, категория, значимость (CRITICAL, HIGH, MEDIUM, LOW, INFO), заголовок, рекомендация, доказательства со сведениями о происхождении (evidence.provenance), трудоемкость и эффект исправления по шкале от 1 до 3, пояснение why. Поля с окончанием _en содержат английский текст

passed_checks

Значение: Проверки, выполненные без замечаний

checks_total, not_checked

Значение: Число выполненных проверок и число проверок, которые не выполнялись, потому что страница не открылась

limitations

Значение: Ограничения проверки

cached

Значение: Присутствует со значением true, если ответ выдан из хранилища результатов (пункт 6.9)

6.7.1. Сохранение отчета в файл

Запрос можно выполнить с сохранением ответа в файл report.json. Не закрывая окно службы, откройте второе окно оболочки и активируйте окружение.

```
curl -X POST http://127.0.0.1:8080/check \
  -H "Content-Type: application/json" \
  -d '{"url":"https://siberia-ai.ru/","lang":"ru"}' \
  -o report.json
python -m json.tool report.json
```

В PowerShell для той же операции можно использовать встроенную команду, чтобы не зависеть от псевдонима curl:

```
Invoke-WebRequest -UseBasicParsing -Method Post `
    -Uri 'http://127.0.0.1:8080/check' `
    -ContentType 'application/json' `
    -Body '{"url":"https://siberia-ai.ru/","lang":"ru"}' `
    -OutFile report.json
.\.venv\Scripts\python.exe -m json.tool report.json
```

Перед использованием результата проверьте код HTTP и содержимое ответа. Сообщение об ошибке тоже может сохраниться в report.json. Успешный ответ /check содержит score, findings, passed_checks, checks_total и rules_version. Проверяемая страница может изменяться, поэтому числовые значения в отчете не обязаны совпадать с историческими примерами.

6.8. Другие команды

Сокращенный отчет:

```
curl -X POST http://127.0.0.1:8080/audit \
  -H "Content-Type: application/json" \
  -d '{"url": "https://siberia-ai.ru/"}'
```

Ответ 29 сентября 2026 года, код 200. Поля ai_readiness, opportunity_teaser и recommended_actions опущены:

```
{"ok": true, "url": "https://siberia-ai.ru/", "final_url": "https://siberia-ai.ru/", "http_status": 200, "
score": 98, "needs_render": false, "counts_by_severity": {"LOW": 1}, "findings_total": 1, "top_fixes":
[], "locked": 1}
```

Поле `top_fixes` содержит до трех находок значимости CRITICAL, HIGH или MEDIUM, `locked` равно числу находок, не вошедших в сокращенный отчет. Полный перечень выдает команда /check. Поле `ai_readiness` содержит оценку готовности к системам поиска на основе искусственного интеллекта от 0 до 100 и ее составляющие.

Доступ роботов систем искусственного интеллекта:

```
curl -X POST http://127.0.0.1:8080/ai-access \
  -H "Content-Type: application/json" \
  -d '{"url": "https://siberia-ai.ru/"}'
```

Ответ, код 200. Перечень `bots` (девять роботов) и `limitations` опущены:

```
{"ok": true, "lang": "ru", "base": "https://siberia-ai.ru", "rules_version": "2026.08.8", "robots":
{"url": "https://siberia-ai.ru/robots.txt", "found": true}, "llms_txt": {"url": "
https://siberia-ai.ru/llms.txt", "present": true}, "summary": {"verdict": "all_open", "
retrieval_blocked": 0, "retrieval_total": 4, "training_blocked": 0, "training_total": 5, "
unclassified_blocked": 0, "unclassified_total": 0}}
```

Вердикт `summary.verdict` принимает значения `all_open` (ничего не закрыто), `training_optout` (закрыты роботы, собирающие данные для обучения), `retrieval_blocked` (закрыты роботы, которые читают страницы для ответов) и `unclassified_blocked` (закрыты роботы с неподтвержденной ролью).

Формирование llms.txt:

```
curl -X POST http://127.0.0.1:8080/llms-txt \
  -H "Content-Type: application/json" \
  -d '{"url": "https://siberia-ai.ru/"}'
```

Ответ, код 200. Поля sections, llms_txt (текст файла) и limitations опущены:

```
{"ok": true, "base": "https://siberia-ai.ru", "brand_source": "homepage_title", "sitemap": {"url": "
https://siberia-ai.ru/sitemap.xml", "found": true, "from_robots": true}, "page_count": 91, "truncated":
true, "valid": true}
```

Обзор сайта:

```
curl -X POST http://127.0.0.1:8080/site \
  -H "Content-Type: application/json" \
  -d '{"url": "https://siberia-ai.ru/"}'
```

```
{"ok":false,"code":"unsupported","message":"Обход сайта целиком в этой сборке недоступен: проверяется
одна страница."}
```

6.9. Ограничения частоты, одновременности и хранилище результатов

Значения по умолчанию задаются переменными окружения при запуске службы. Действие переменных PUBLIC_AUDIT_TIMEOUT и PUBLIC_CHECK_CACHE_TTL проверено, остальные значения взяты из исходного текста программы, а предел 10 обращений в минуту к /check подтвержден ответом 429 на одиннадцатое обращение.

PUBLIC_AUDIT_TIMEOUT

По умолчанию: 20

Значение: Предельное время загрузки проверяемой страницы, секунд

PUBLIC_CHECK_RATE_PER_MIN

По умолчанию: 10

Значение: Обращений к /check в минуту с одного адреса клиента

PUBLIC_AUDIT_RATE_PER_MIN

По умолчанию: 10

Значение: Обращений к /audit в минуту с одного адреса. Для /check, /ai-access и /llms-txt это значение по умолчанию

PUBLIC_AUDIT_GLOBAL_RATE_PER_MIN

По умолчанию: 300

Значение: Обращений в минуту со всех адресов вместе, на каждую команду

PUBLIC_CHECK_MAX_CONCURRENCY

По умолчанию: 4

Значение: Одновременно выполняемых проверок /check

PUBLIC_CHECK_CACHE_TTL

По умолчанию: 120

Значение: Сколько секунд ответ /check выдается повторно из хранилища результатов. Значение 0 отключает повторную выдачу

PUBLIC_AI_CACHE_TTL, PUBLIC_LLMSGEN_CACHE_TTL

По умолчанию: 300, 600

Значение: То же для /ai-access и /llms-txt

Повторный запрос того же адреса на том же языке в пределах срока хранения выдается из хранилища без обращения к проверяемому ресурсу, с полем `"cached": true`. Для повторной проверки после изменения страницы выждите срок хранения или перезапустите службу.

6.10. Сообщения об ошибках

При отказе служба отвечает кодом HTTP и телом `{"ok": false, "code": ..., "message": ...}`. Поле `code` предназначено для обработки программой, `message` для человека.

400 invalid_url

Ответ программы: `Enter a valid http(s) website URL.`

Когда возникает и что делать: Адрес пуст, схема не http и не https, имя сайта содержит пробелы или пустые части. Исправить адрес

400 blocked_target

Ответ программы: `That host can't be checked.` (для /audit: `That host can't be audited.`)

Когда возникает и что делать: Адрес ведет на петлевой, частный, служебный или зарезервированный адрес, либо на порт, отличный от 80 и 443. Такие адреса служба не запрашивает, это защита от обращений к внутренним ресурсам. Страницу во внутренней сети проверяют функцией audit_url (пункт 6.12). Команда /audit так же отвечает и на имя сайта, которое не удалось разрешить

400 unreachable

Ответ программы: `That domain could not be resolved - check the spelling.`

Когда возникает и что делать: Имя сайта не удалось разрешить (/check, /ai-access, /llms-txt). Проверить написание адреса и доступ машины к службе имен

422 unsupported_content

Ответ программы: `This looks like image/png - the checker analyzes HTML pages.`

Когда возникает и что делать: По адресу лежит не страница HTML (изображение, архив). Указать адрес страницы

429 rate_limited

Ответ программы: `Too many checks - try again in a minute.`

Когда возникает и что делать: Превышено число обращений в минуту (пункт 6.9). Повторить через минуту

503 busy

Ответ программы: `The checker is at capacity - try again in a few seconds.`

Когда возникает и что делать: Заняты все места одновременной обработки. Повторить через несколько секунд

504 timeout

Ответ программы: `The page took too long to respond.`

Когда возникает и что делать: Проверяемый сервер не ответил в срок PUBLIC_AUDIT_TIMEOUT

501 unsupported

Ответ программы: `Обход сайта целиком в этой сборке недоступен: проверяется одна страница.`

Когда возникает и что делать: Команда /site в этой сборке не поддерживается

422 без поля code

Ответ программы: `{"detail":[{"type":"missing","loc":["body","url"],"msg":"Field required","input":{}}]}`

Когда возникает и что делать: В запросе нет поля url или тело не является JSON (тип json_invalid). Исправить запрос

404, 405

Ответ программы: `{"detail":"Not Found"},{"detail":"Method Not Allowed"}`

Когда возникает и что делать: Неверный путь или метод, например GET вместо POST

Если страница не открылась по иной причине, отчет формируется с кодом 200 и одной находкой `unreachable` значимости CRITICAL, балл равен 0, а поле `not_checked` называет число невыполненных проверок. Так же отвечает функция audit_url, пример в пункте 6.12. Для функции это проверено, для HTTP-службы описано по исходному тексту программы и не проверялось.

6.11. Журнал работы

Сообщения о запуске служба печатает в стандартный поток ошибок, сообщения об обращениях в стандартный поток вывода. Пример строк журнала при обращениях:

```
2026-09-28 17:35:26 [info     ] check.blocked_target           host=127.0.0.1
INFO:     127.0.0.1:50719 - "POST /check HTTP/1.1" 400 Bad Request
2026-09-28 17:35:43 [info     ] check.ok                       findings=9 host=siberia-ai.ru lang=en
ms=1281 status=200
2026-09-28 17:35:43 [info     ] check.rate_limited
```

Строки вида `check.ok` пишет программа: в них только имя проверяемого сайта, без полного адреса и без адреса клиента. Строки с INFO пишет сервер приложений, в них есть адрес клиента. Параметр запуска `--no-access-log` их отключает. Время в журнале местное.

6.12. Работа без HTTP-службы: функция audit_url

Функция проверяет одну страницу и возвращает словарь Python. Вызов:

Сохраните следующий код в файл audit_example_33.py и выполните его интерпретатором окружения программы. Файл также приведен в приложенных примерах команд.

```
from sibir_audit.modules.audit.audit import audit_url
from sibir_audit.modules.audit.checks import RULES_VERSION
r = audit_url('https://siberia-ai.ru/', lang='ru')
print(RULES_VERSION, r['http_status'], r['score'], r['checks_total'])
for f in r['findings']:
    print(f['severity'], f['code'], f['title'])
```

```
python audit_example_33.py
```

```
2026.08.8 200 98 31
LOW schema_sameas_no_kg_anchor Organization sameAs links to social profiles, not a knowledge graph
```

url

Значение: Адрес страницы. Функция принимает и петлевые, и внутренние адреса, в отличие от HTTP-службы

lang

Значение: Набор проверок по языку, по умолчанию `"en"`. Функция возвращает заголовки находок по-английски при любом значении, перевод выполняет HTTP-служба

timeout

Значение: Предельное время загрузки страницы, секунд, по умолчанию 20

check_robots

Значение: Загружать ли robots.txt для проверки индексируемости, по умолчанию да

Возвращаемый словарь содержит поля url, final_url, http_status, elapsed_ms, score, counts_by_severity, findings, passed_checks и checks_total в том же значении, что в ответе /check (пункт 6.7), а также geo_readiness (эвристическая оценка готовности исходного HTML к обработке системами искусственного интеллекта) и clean_but_invisible. Если страница не открылась, находка одна:

Сохраните следующий код в файл audit_example_35.py и выполните его интерпретатором окружения программы. Файл также приведен в приложенных примерах команд.

```
from sibir_audit.modules.audit.audit import audit_url
r = audit_url('https://net-takogo-domena.invalid/', lang='ru')
print(
    r['score'],
    r['checks_total'],
    r['not_checked'],
    r['findings'][0]['code'],
    r['findings'][0]['severity'],
)
```

```
python audit_example_35.py
```

```
0 1 31 unreachable CRITICAL
```

Семантические оценки текста вызываются отдельными функциями, не функцией audit_url и не HTTP-службой. Порядок вызова, установка onnxruntime и ответы программы приведены в пункте 5.1.

7. Прекращение работы и удаление

Работа прекращается остановкой процесса сервера. Удаление выполняется удалением каталога программы и изолированного окружения. При аудите программа обращается к проверяемым адресам. Ограничения сетевого доступа задает администратор среды. Наблюдение за семантическими функциями в разделе 9 не является заключением о сетевой изоляции всего приложения.

7.1. Служба останавливается нажатием Ctrl+C в окне терминала, где она запущена, или сигналом SIGTERM процессу службы. Служба перестает принимать новые обращения, дожидается завершения обращений, которые уже выполняются, и завершает процесс. Журнал при этом:

```
INFO:     Shutting down
INFO:     Waiting for connections to close. (CTRL+C to force quit)
2026-09-28 17:36:12 [info     ] check.ok                       findings=5 host=siberia-ai.ru lang=ru
ms=1406 status=200
INFO:     127.0.0.1:50787 - "POST /check HTTP/1.1" 200 OK
INFO:     Waiting for application shutdown.
INFO:     Application shutdown complete.
INFO:     Finished server process [13644]
```

Здесь проверка страницы, начатая до сигнала остановки, завершилась с кодом 200, после чего процесс завершился. Повторное нажатие Ctrl+C во время ожидания прерывает незавершенные обращения. Промежуточных данных между запусками служба не хранит: хранилище результатов и счетчики обращений находятся в памяти процесса и при остановке очищаются.

7.2. Функция audit_url завершается вместе с вызвавшей ее программой и отдельной остановки не требует.

Команды и ответы этого раздела получены правообладателем 28 и 29 сентября 2026 года на экземпляре 1.2.0 в Windows Server 2022, Python 3.12.10, сервер приложений uvicorn 0.54.0, в оболочке bash. Не проверялись: активация окружения командой `. .venv/bin/activate` в Linux (в Windows выполнена `.venv/Scripts/activate`), нажатие Ctrl+C с клавиатуры и сигнал SIGTERM (остановка проверена сигналом CTRL_BREAK процессу службы), повторное нажатие Ctrl+C, ответ 503 busy, язык отчета zh. Ответы 504 timeout и 429 rate_limited получены при измененных значениях PUBLIC_AUDIT_TIMEOUT и после серии обращений.

8. Техническая поддержка

Обращения принимаются по адресу info@siberia-ai.ru. Порядок и сроки поддержки определяются лицензионным договором.

Условия предоставления прав использования: Лицензионная политика ООО «ФИИ» и приложение № 1 с тарифными позициями «Сибирь.Аудит».

9. Проверка редакции от 28 октября 2026 года

Экземпляр установлен заново из архива с контрольной суммой, указанной на титульной странице. Создано отдельное виртуальное окружение на существующей машине Windows Server 2022, Python 3.12.10. Для установки использован публичный источник пакетов. Это новая установка программы, не проверка на новой операционной системе и не автономная установка.

В этом руководстве используется рабочее имя модуля запуска: sibir_audit.public_audit:app. Оно совпадает с текущим публичным HTML. Старый подписанный AUDIT-16 указывает sibir_audit.public, которого в этом архиве нет. Подтверждены целостность всех 73 файлов архива, импорт из распакованного каталога, запуск и повторный запуск службы, основные операции, сохранение отчета, проверка локальной контролируемой страницы через библиотеку, недоступный адрес, ошибки запроса, ограничение частоты и занятый порт. На доступной странице в наборе правил 2026.08.8 выполнена 31 проверка.

Семантические функции с комплектной моделью вернули признак osnovanie='model'. Без файла модели, без передачи модели и при недоступном импорте onnxruntime вернулся резервный режим osnovanie='leksika'. Во время этих вызовов Python не пытался открыть сокет. Изоляция сети на уровне операционной системы не применялась.

Известное ограничение кеша /check. Если сначала отправить неизвестный язык, а затем тот же адрес с lang='en', повторный ответ может сохранить lang_requested и lang_fallback от первого запроса. Поле lang и язык содержания при этом корректны. Для точной метаинформации каждого запроса установите PUBLIC_CHECK_CACHE_TTL=0 перед запуском службы. Этот обходной режим воспроизведен: второй запрос вернул lang_requested=en и lang_fallback=false. При изменении этой настройки нужен перезапуск. Исходный архив в рамках редакции документа не исправлялся.

Остановка на проверенном стенде. После CTRL_BREAK_EVENT обе службы завершились, журналы содержат Application shutdown complete и Finished server process, без исключений. Код процесса равен 3, поэтому критерий завершения с кодом 0 не принят. Повторный запуск и /healthz успешны. Отдельная программная остановка Uvicorn через should_exit=True завершилась с кодом 0. Нажатие Ctrl+C с клавиатуры и SIGTERM, Linux и сценарий остановки под нагрузкой в новом прогоне не проверены.

Первичный протокол содержит 22 проверки: 19 успешны, один дефект метаинформации кеша и два несоответствия критерию кода завершения процесса. Это не заключение о полной приемке на всех заявленных средах. Полный протокол, состав зависимостей и границы проверки оформлены отдельным приложением к рабочему комплекту.