

**РУКОВОДСТВО ПО УСТАНОВКЕ И ЭКСПЛУАТАЦИИ
ИНФОРМАЦИОННОЙ СИСТЕМЫ (ИС) «Граф-ИТ»
НА 17 ЛИСТАХ**

Определения, обозначения и сокращения

БД	База данных
ИС	Информационная система
СУБД	Система управления базами данных

Содержание

Определения, обозначения и сокращения	2
1. Назначение системы	4
2. Функциональные возможности системы	4
3. Технические требования системы	6
4. Инструкция по установке ИС Граф-ИТ	7
4.1 Состав дистрибутива	7
4.2 База данных	9
4.3 Настройка сервера PostgreSQL	9
4.3.1 Пароль администратора	9
4.3.2 Файл pg_hba.conf	9
4.3.3 Утилита pg_restore	9
4.4 Развертывание баз данных	9
4.5 Заполнение созданной БД данными (импорт)	10
4.6 Серверная часть	10
4.7 Предварительная настройка	11
4.7.1 CORS	11
4.7.2 Подключение к БД	12
4.7.3 Настройка Tomcat	12
4.7.4 Порт	13
4.8 Запуск	13
4.9 Клиентская часть	14
4.9.1 Перенос файлов клиентской части в папку Tomcat	14
4.9.2 Настройка клиентского приложения	15
4.9.3 Кэширование браузера	16

1. Назначение системы

Информационная система «Граф-ИТ» предназначена для визуализации, мониторинга и анализа данных.

2. Функциональные возможности системы

1. Просмотр тематических интерактивных информационных досок – дэшбордов, состоящих из «процессоров» - логических блоков, объединяющих в себе функционал запроса данных и отображения полученных по запросу данных в виде таблицы и/или графика.

Дэшборд предоставляет пользователю информацию об объекте в любом разрезе.

2. Управление запросом в процессоре, а именно

- переустановка фильтров;
- выбор дополнительных измерений;

Сохранение выбранных фильтров и измерений по кнопке для того, чтобы этот запрос применялся всегда в дальнейшем для процессора.

Есть возможность изменить сразу все фильтры дэшборда через «глобальные фильтры» и перезапросить данные (и, соответственно, перестроить процессоры) для всего дэшборда.

Типы фильтров:

- по справочнику (во всплывающем окне пользователь может выбрать одну, несколько или все сразу записи измерения, есть возможность поискать запись по любому полю);
- фильтр-календарь (во всплывающем окне выбирается дата в стандартном веб-календаре);
- фильтр по значению (пользователь вводит число в поле ввода);

3. Управление визуальными компонентами.

Со стороны графика:

- изменение типа графика;
- настройка осей, палитры, легенды;
- добавление уровней;
- транспонирование графика;

Доступные типы графиков:

- a) Гистограмма (нормальная, стекированная, стекированная и нормированная);
- b) Круговая диаграмма (обычная и кольцевая);
- c) Линейный (обычный, стекированный, стекированный и нормированный);
- d) Область (обычная, стекированная, стекированная и нормированная);
- e) Плоское дерево;
- f) Гаудж («спидометр»);
- g) Шкала;
- h) Карта (требуется .shp и .dbf файлы);
- i) Текстовая карточка (может содержать любой HTML-код, в том числе изображение).

Со стороны таблицы:

- a) настройка сортировки и фильтрации, поиск;
- b) выбор столбцов и колонок;
- c) сохранение группировки, развернутых рядов и столбцов.

Доступные типы таблиц:

- d) простая;
- e) пивотная;
4. Экспорт визуальных компонент: графиков в графические файлы, таблиц в файлы форматов электронных таблиц.
5. Перемещение процессоров и фильтров по дэшборду, изменение их размера с возможностью сохранить эти изменения для себя.

6. Режим просмотра процессора для удобства настройки, применения инструментов анализа, настройки фильтров.

Из режима просмотра можно быстро перейти к «связанным процессорам» (если таковые указаны были при настройке процессора администратором)

7. Инструменты анализа данных:

а) Heatmap - подсвечивание цветом;

б) Вычисление абсолютного прироста, темпа роста, базисного и цепного темпа роста;

с) Вычисление суммы N первых/последних значений;

8. Прогнозирование.

Доступные методы:

– По среднему;

– Экспоненциальное сглаживание;

– Средний абсолютный прирост;

– Средний темп роста;

– Линейная регрессия;

– Взвешенное скользящее среднее.

3. Технические требования системы

ИС «Граф-ИТ» включает в себя:

1) БД;

2) Серверную часть (бэкэнд);

3) Клиентскую часть (фронтэнд).

Сервер БД:

– от 2Гб ОЗУ,

– Процессор 4x2 ГГц,

– Операционная система Ubuntu (или любая *nix ОС),

– Свободное дисковое пространство не менее 50 Гб (в зависимости от объема БД).

Сервер приложения (здесь разворачиваются фронтэнд и бэкэнд):

- от 4Гб ОЗУ,
- Процессор 4х2 ГГц,
- Операционная система Ubuntu (или любая *nix ОС)

Клиент:

- от 1024 Мб ОЗУ,
- один из браузеров:
 - a) Google Chrome версии ≥ 32.0 ;
 - b) Firefox версии ≥ 39 ;
 - c) Safari версии ≥ 8 ;
 - d) Opera версии ≥ 27 ;
 - e) Internet Explorer версии выше 9 или MS Edge;

4. Инструкция по установке ИС Граф-ИТ

4.1 Состав дистрибутива

В папке распространяемого дистрибутива ИС «Граф-ИТ» расположены файлы, указанные в таблице 1.

Таблица 1 – подробный состав и назначение файлов дистрибутива

№	Файл	Описание
1	database/bi_data.sql	Скрипт создания БД для хранения данных мониторинга
2	database/bi_metadata.sql	Скрипт создания БД, необходимого для работы самого приложения
3	database/bi_data.backup	Бэкап данных (полезные данные)
4	database/bi_metadata.backup	Бэкап метаданных (сервисные данные)
5	grafit-backend/bi-backend.war	Серверная часть
6	grafit-frontend/app	Клиентская часть
7	grafit-frontend/index.html	Главная страница
8	grafit-frontend/bower.json	Список зависимостей, которые надо установить

4.2 База данных

Для работы ИС «Граф-ИТ» необходима СУБД PostgreSQL (<https://www.postgresql.org/>). Установите на сервер БД сервер Postgres версии не ниже 9.5.

4.3 Настройка сервера PostgreSQL

4.3.1 Пароль администратора

После установки сервера PostgreSQL необходимо задать пароль администратора:

```
ALTER USER postgres WITH PASSWORD ':password'
```

где, :password – пароль администратора сервера PostgreSQL.

4.3.2 Файл pg_hba.conf

Предоставьте доступ к серверу PostgreSQL серверу приложения, на котором будет установлена серверная часть приложения (бэкэнд), пропишите IP адрес сервера приложения в файле **pg_hba.conf**.

После редактирования файла **pg_hba.conf** необходимо перезапустить сервис postgresql.

4.3.3 Утилита pg_restore

Замените строку: «local all postgres peer» на «local all postgres md5», в файле **pg_hba.conf** для возможности выполнения импорта данных с помощью утилиты pg_restore.

После редактирования файла **pg_hba.conf** необходимо перезапустить сервис postgresql.

4.4 Развертывание баз данных

Отредактируйте скрипты bi_data.sql и bi_metadata.sql: необходимо изменить строки:

```
LC_COLLATE = 'en_US.UTF-8'
```

```
LC_CTYPE = 'en_US.UTF-8'
```

на

```
LC_COLLATE = 'значение используемой локали'
```

```
LC_CTYPE = 'значение используемой локали'
```

Где 'значение используемой локали' можно узнать, выполнив на сервере PostgreSQL команду:

```
SHOW LC_COLLATE
```

Далее необходимо запустить предоставляемые в дистрибутиве (папка database) скрипты **bi_data.sql** и **bi_metadata.sql** на сервере PostgreSQL. В результате будут созданы две базы данных: **bi_data** для хранения данных мониторинга и **bi_metadata** для хранения сервисных данных.

БД **bi_data** предоставляется в демонстрационных целях.

4.5 Заполнение созданной БД данными (импорт)

Для импортирования данных воспользуйтесь утилитой **pg_restore** или **pgAdmin**. Импорт данных выполняется из бэкапов **bi_data.sql** и **bi_metadata.sql** в папке **database** предоставляемого дистрибутива.

В **pgAdmin** импорт происходит через команду восстановление (Restore), для восстановления через утилиту **pg_restore** – выполнить:

```
pg_restore -c -d bi_data -U postgres bi_data.sql и pg_restore -c -d bi_metadata -U postgres bi_metadata.sql
```

4.6 Серверная часть

Для развертывания серверной части на сервере приложения необходимо установить:

1. Java 1.8 <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>;

Скачайте установочный файл с официального сайта и запустите его.

При этом если Java SDK или Java Runtime Environment уже стояли на машине, то, скорее всего, придется скорректировать переменную JAVA_HOME, которая все еще может указывать на папку, в которую установлена предыдущая версия Java.

2. Apache Tomcat 8 <https://tomcat.apache.org/download-80.cgi>;

Если вы выбрали установку через архив, то, после скачивания архива, распакуйте его – в разархивированной папке будет скелет сервера Tomcat. Tomcat работает с системной переменной CATALINA_HOME. Если Tomcat был скачан архивом, надо изменить переменную, чтобы она указывала на папку, в которую был распакован архив Tomcat.

4.7 Предварительная настройка

4.7.1 CORS

Необходимо выполнить настройку CORS-фильтра. Для этого следует открыть предоставляемый в дистрибутиве файл **grafit-backend/bi-backend.war** как папку/архив. Внутри **bi-backend** найти **WEB-INF\app-context.xml**.

В нем найти строку «белый список CORS»

```
<util:list id="corsDomainswhiteList" value-type="java.lang.String">
```

и добавить записи

```
<value>url:port</value>
```

где url – адрес, с которого клиентская часть будет обращаться к backend-серверу,

port – порт, по которому обращаются.

Нужно внести в «белый список CORS» адрес машины, на которой установлена клиентская часть (например, `<value>http://localhost:8080</value>`).

4.7.2 Подключение к БД

Для настройки подключения к БД необходимо открыть предоставляемый в дистрибутиве файл **grafit-backend/bi-backend.war** как папку/архив. Внутри **bi-backend** найти **WEB-INF\conf\bi-backend\env-context.xml**.

В нем найти строку

```
<property name="url" value="jdbc:postgresql://url_address:port/bi_metada">
```

заменить url_address – адрес установки postgresql, port – порт запуска.

В строках

```
<property name="username" value="username_value"/>
```

заменить username_value на имя пользователя,

```
<property name="password" value="password_value">
```

заменить password_value на пароль для доступа к серверу PostgreSQL (логин и пароль администратора задавались в п. 4.3.1).

4.7.3 Настройка Tomcat

Необходимо сделать файлы **catalina.sh** и **startup.sh** выполняемыми, для этого необходимо кликнуть правой клавишей мыши по файлам, выбрать «Свойства» (Properties), в появившемся меню выбрать вкладку «Разрешения» (Permission) и установить галку у пункта «Разрешить запускать файл как программу» (Allow executing file as program) или в консоли выполнить команды:

```
chmod +x {ТОМСАТ_НОМЕ}/bin/catalina.sh,  
chmod +x {ТОМСАТ_НОМЕ}/bin/startup.sh
```

Добавьте роли для пользователя администратор. Для этого в файле **{ТОМСАТ_НОМЕ}/conf/tomcat-users.xml** найдите секцию **<tomcat-users>** и добавьте в неё строку:

```
<user username="admin" password="password" roles="manager-gui,admin-gui">,
```

придумав “password” – пароль пользователя-администратора “admin”, который будет иметь доступ к панели администрирования через веб-интерфейс Tomcat (<http://localhost:8080/manager>).

4.7.4 Порт

Если есть необходимость изменения порта, на котором будет запущен сервер, зайдите в папку **{TOMCAT_HOME}/conf/server.xml** и измените параметр **Connector port=""**. По умолчанию сервер Tomcat запускает приложения на порту 8080.

Адрес и порт, по которому доступен бэкэнд потом нужно будет указать при настройке клиентской части (см. п.4.9.2).

4.8 Запуск

Сначала необходимо запустить сервер Tomcat. Для этого используется:

{TOMCAT_HOME}/bin/startup.sh

Затем требуется развернуть серверную часть. Для установки приложения бэкэнд (war-файла) необходимо открыть панель администрирования сервера Tomcat (логин и пароль задавались в файле **tomcat-users.xml**, см. п.4.7.3). в браузере.

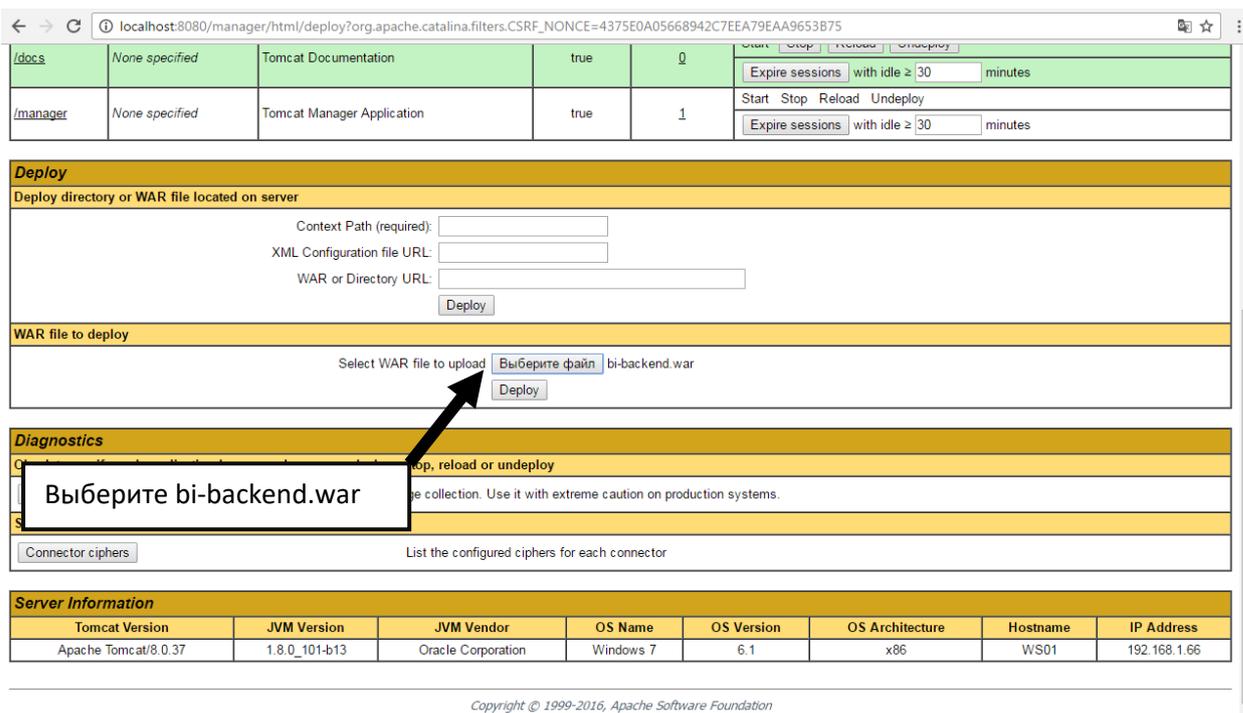


Рисунок 1 – Выбор установочного war файла в панели администрирования

В секции «WAR file to deploy» (Рисунок 1) нужно выбрать предоставляемый в дистрибутиве war файл **grafit-backend/bi-backend.war** и нажать кнопку Deploy.

Приложение будет развернуто и автоматически запущено.

Если приложение не запустилось, можно запустить его вручную, нажав кнопку «Start».

В случае возникновения ошибок требуется изучить логи Tomcat.

4.9 Клиентская часть

4.9.1 Перенос файлов клиентской части в папку Tomcat

Папку **grafit-frontend** дистрибутива нужно скопировать в директорию Tomcat **{Tomcat_Home}/webapps**, предназначенную для статических файлов, где уже должна находиться папка с развернутой серверной частью **bi-backend** и установочный файл **bi-backend.war**.

Можно переименовать папку **grafit-frontend** по своему усмотрению. Тогда адрес, по которому будет доступно клиентское приложение, будет не localhost:8080\grafit-frontend, а localhost:8080\новое_имя_папки.

4.9.2 Настройка клиентского приложения

Для того, чтобы клиентская часть могла обращаться к backend-серверу, нужно прописать базовую часть URL в файле, содержащем константы, используемые приложением – **app.constants.js**, который находится в папке **grafit-frontend/app**. Найдите в файле **app.constants.js** переменную **urlToBackend** и измените ее так, чтобы она содержала действующий URL, по которому доступна серверная часть.

8080 – порт Tomcat по умолчанию, **bi-backend** – путь к бэкэнду в папке **{Tomcat_Home}/webapps**, а часть адреса «**api**» изменять не нужно.

Чтобы проверить, что приложение работает, откройте браузер и перейдите по адресу, по которому должно быть доступно клиентское приложение:

[имя_машины_или_localhost] : [порт_Tomcat] / [имя_папки_frontend]

Фронтэнд-часть [имя_папки_frontend] из папки **{Tomcat_Home}/webapps**. Например, если порт и имя frontend-папки были оставлены по умолчанию, адрес localhost:8080/grafit-frontend.

Приложение, не обнаружив активной пользовательской сессии, перейдет на страницу авторизации (Рисунок 2).

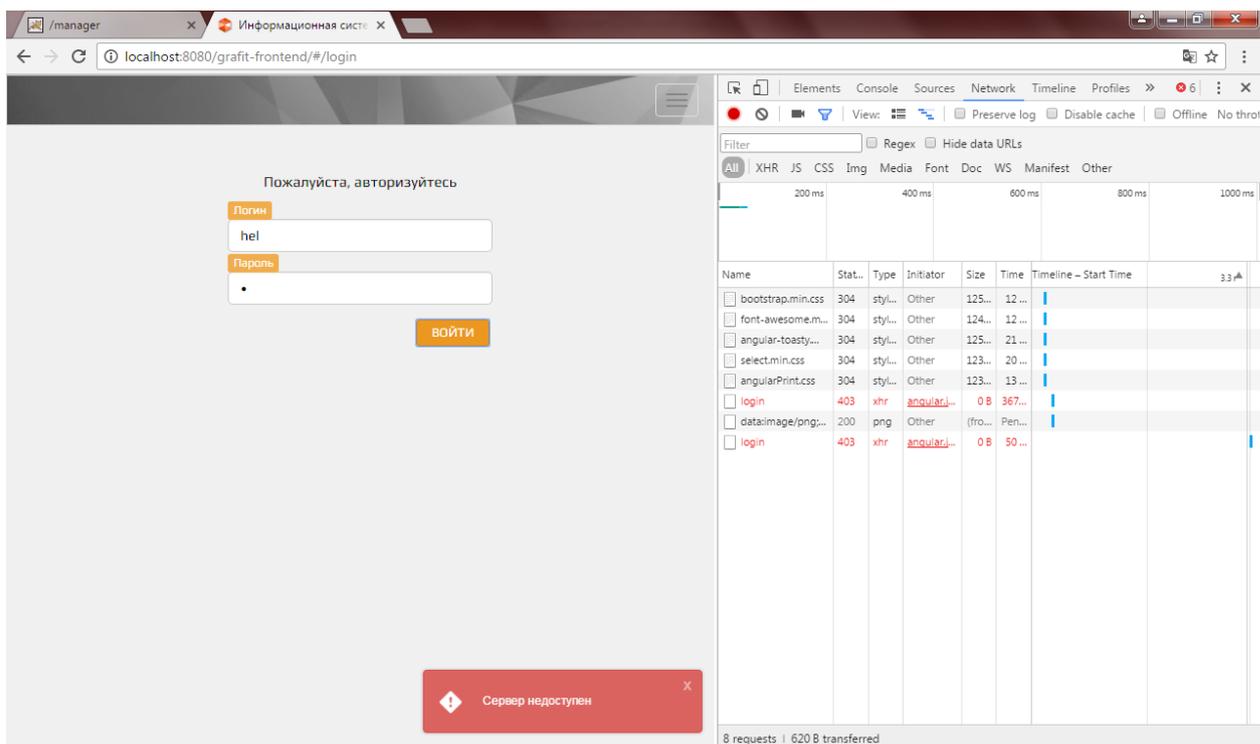


Рисунок 2 – Страница авторизации

Тестовая учетная запись для входа:

Логин: «testuser1»

Пароль: «1»

4.9.3 Кэширование браузера

Иногда браузер кэширует JS-файлы, и, например, если сначала приложение было открыто в браузере, а потом появилась необходимость внести изменения в файл конфигурации **grafit-frontend/app/app.constants.js** (допустим, изменить URL бэкенд-сервера), то изменения могут не примениться, потому что браузер использует кэшированную версию файла **app.constants.js**.

В таком случае в первую очередь нужно посмотреть на сам файл, используемый браузером. Для этого клавишей F12 откройте консоль разработчика, найдите файл среди JS-файлов приложения и убедитесь, что внесенных в **app.constants.js** изменений в нем нет (Рисунок 3).

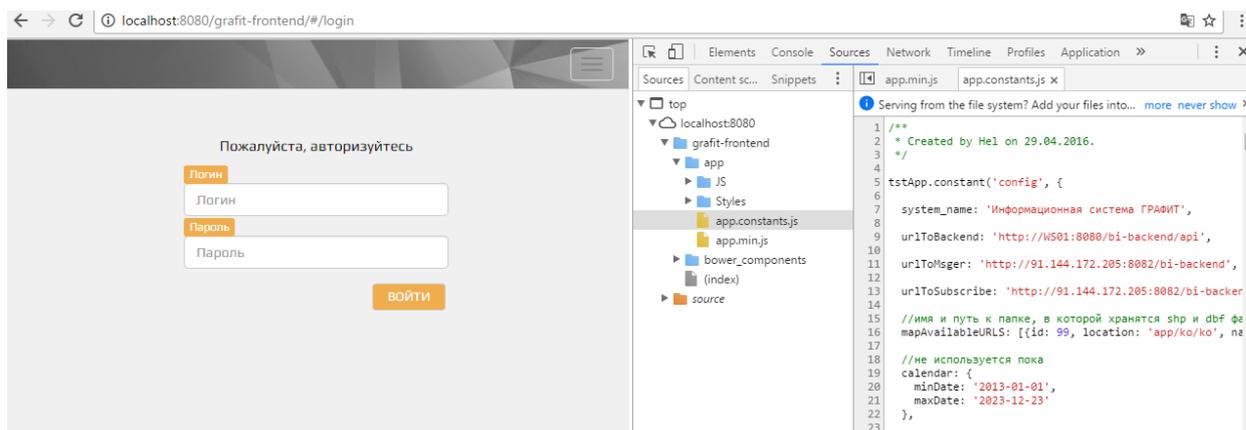


Рисунок 3 – Консоль разработчика с открытым скриптом app.constants.js

В этом случае, отключите кэширование файлов и/или обновите страницу браузера с опцией «Очистить кэш».