



GIMS AUTOMATION MANAGER

РАБОЧАЯ ДОКУМЕНТАЦИЯ

РУКОВОДСТВО ПО ЭКСПЛУАТАЦИИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Москва 2020



СОДЕРЖАНИЕ

1	ОБЩИЕ ПОЛОЖЕНИЯ	3
1.1	НАИМЕНОВАНИЕ АС	3
1.2	НАЗНАЧЕНИЕ СИСТЕМЫ.....	3
1.3	КРАТКОЕ ОПИСАНИЕ ВОЗМОЖНОСТЕЙ	3
1.4	УРОВЕНЬ ПОДГОТОВКИ ПОЛЬЗОВАТЕЛЯ	4
2	ПОДГОТОВКА К РАБОТЕ	4
2.1	СОСТАВ ТЕХНИЧЕСКИХ СРЕДСТВ	4
2.2	ПЕРЕЧЕНЬ ЭКСПЛУАТАЦИОННОЙ ДОКУМЕНТАЦИИ.....	5
2.3	ПОРЯДОК ПРОВЕРКИ РАБОТОСПОСОБНОСТИ.....	5
2.3.1	Обработка ошибок при запуске активатора.....	5
2.3.2	Обработка ошибок в настройке сценария автоматизации	6
2.4	ОБНОВЛЕНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ	9



1 ОБЩИЕ ПОЛОЖЕНИЯ

1.1 НАИМЕНОВАНИЕ АС

Полное наименование: Gelarm Infrastructure Management Systems Automation Manager

Обозначение: GIMS Automation Manager или GIMS или Система

1.2 НАЗНАЧЕНИЕ СИСТЕМЫ

Система GIMS Automation Manager предназначена для решения следующих задач:

1. Настройка различных API (REST, SOAP, DBlink, SNMP, CLI и т.д.) для обмена между информационными системами компании;
2. Реализация сценариев обработки информации, получаемой через эти API;
3. Предоставление пользователю настраиваемых web интерфейсов для работы с информацией (SCADA, BI, Reports);
4. Обеспечение отказоустойчивости и балансировки нагрузки путем реализации как горизонтальной, так и вертикальной масштабируемости.

1.3 КРАТКОЕ ОПИСАНИЕ ВОЗМОЖНОСТЕЙ

В соответствии с установленными правами доступа Система обеспечивает доступ к следующим функциональным возможностям:

- Конфигурирование инфраструктуры Системы;
- Разработка шаблонов подключения к источникам данных;
- Создание подключения к источникам данных;
- Разработка правил по обработке сигналов от источников данных и выполнение сценариев;
- Разработка сценариев для обработки данных, получаемых из API, согласно заданному алгоритму;
- Работа с конструктором графических интерфейсов.



1.4 УРОВЕНЬ ПОДГОТОВКИ ПОЛЬЗОВАТЕЛЯ

Для проведения работ по настройке системы GIMS Automation Manager предусмотрена работа пользователя с правами Администратор портала. Для данного пользователя предъявляются следующие требования:

- Опыт работы с персональным компьютером на базе операционной системы Microsoft Windows, уметь осуществлять базовые операции в стандартных Windows приложениях;
- Опыт разработки на языке программирования Python;
- Базовые знания работы с СУБД.

2 ПОДГОТОВКА К РАБОТЕ

2.1 СОСТАВ ТЕХНИЧЕСКИХ СРЕДСТВ

Система GIMS Automation Manager представляет собой веб-приложение, доступное для зарегистрированных пользователей, имеющих соответствующие права. В состав Системы входят следующие модули:

- Графический интерфейс пользователя Портала – модуль представляет собой веб-приложение, доступное для зарегистрированных пользователей, имеющих соответствующие права.
- Infrastructure Manager – модуль для централизованного управления системой;
- Listener – модуль, предоставляющий для внешних систем программный интерфейс (API) для взаимодействия с Системой;
- Worker – модуль, отвечающий за обработку информации согласно заданным сценариям автоматизации;
- Logs collectors – сбор и хранение журналов выполнения сценариев автоматизации для анализа и отладки.



2.2 ПЕРЕЧЕНЬ ЭКСПЛУАТАЦИОННОЙ ДОКУМЕНТАЦИИ

В состав эксплуатационной документации GIMS Automation Manager входят следующие документы:


- Руководство пользователя;
- Руководство по установке системы.

2.3 ПОРЯДОК ПРОВЕРКИ РАБОТОСПОСОБНОСТИ

Работоспособность Системы проверяется в ходе его эксплуатации. В модулях должны корректно выполняться операции, обрабатываться ошибочные и аварийные ситуации и выдаваться пользователю соответствующие диагностические сообщения.

Все описанные ниже сценарии выполняются после авторизации пользователя в системе GIMS Automation Manager с правами Администратора портала.

2.3.1 Обработка ошибок при запуске активатора

- 1) Выберите пункт меню «Активаторы сервисов». После нажатия на кнопку Система откроет форму «Активаторы сервисов» разделенную на панель с перечнем настроенных в Системе активаторов со статусом их работы, а также рабочую область.
- 2) Выберите в панели активаторов пункт, напротив которого стоит статус не активен .

Имя	Тип	Состояние
Get_VMWare_Config	Расписание	✓
SNMP_Trapping_Prod	SNMP_Trapping	✗
Syslog_Prod	Syslog	✓
Syslog_Test	Syslog	✓

Тип активатора: Расписание

Сценарий: Сценарий 1

Автозапуск

Интервал запуска, сек.: 600


Таймаут выполнения, сек.: 300

```
2020-08-05 16:26:24,019 [INFO] === START POLICY (pid: 1509) ===
2020-08-05 16:26:24,298 [INFO] idb_sandbox_pgsql:
2020-08-05 16:26:24,298 [INFO] idb_sandbox_pgsql.jdbc_class = org.postgresql.Driver
2020-08-05 16:26:24,298 [INFO] idb_sandbox_pgsql.url = jdbc:postgresql://192.168.88.12:5432/srt_monitoring?
2020-08-05 16:26:24,298 [INFO] idb_sandbox_pgsql.login = srt
2020-08-05 16:26:24,299 [INFO] idb_sandbox_pgsql.password = Vjybnjhi23
2020-08-05 16:26:24,299 [INFO] idb_sandbox_pgsql.jdbc_path =
```

Рисунок 1 – Форма «Активаторы»

- 3) Нажмите на кнопку «Отобразить журнал».
- 4) В описании работы активатора определите причину сбоя.

Примечание: Так как при работе активатора Система задействует работу сразу нескольких логических компонент (сценарий автоматизации, активатор, источник данных) в описании ошибки Система указывает в каком из компонент произошел сбой при работе активатора.

- 5) Устраните причину сбоя.
- 6) Нажмите на кнопку «Выполнить код» на форме «Активаторы».
- 7) Убедитесь, что в Журнале выполнения сценария ошибка устранена.
- 8) Нажмите кнопку «Сохранить» .
- 9) Через заданный в Системе промежуток времени будет выполнено соединение с источником и в панели «Статус» сменится статус подключения.

2.3.2 Обработка ошибок в настройке сценария автоматизации

- 1) Выберите пункт меню «Управление сценариями». После нажатия на кнопку Система откроет форму «Управление сценариями» разделенную на панель с перечнем ранее созданных сценариев, а также рабочую область.

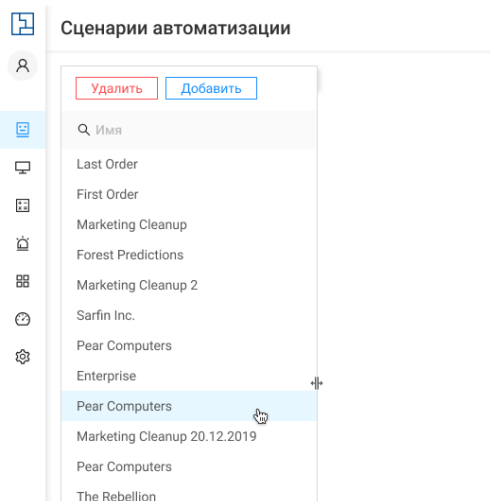


Рисунок 2 – Форма «Управление сценариями»

- 2) Выберите из списка сценарий автоматизации, который работает не корректно. После нажатия на кнопку в рабочей области формы «Управления сценариями» будет открыта вкладка. Открывшаяся вкладка состоит из одного блока для ввода текста исполняемого кода. (Система поддерживает ввод кода на языке Python.)

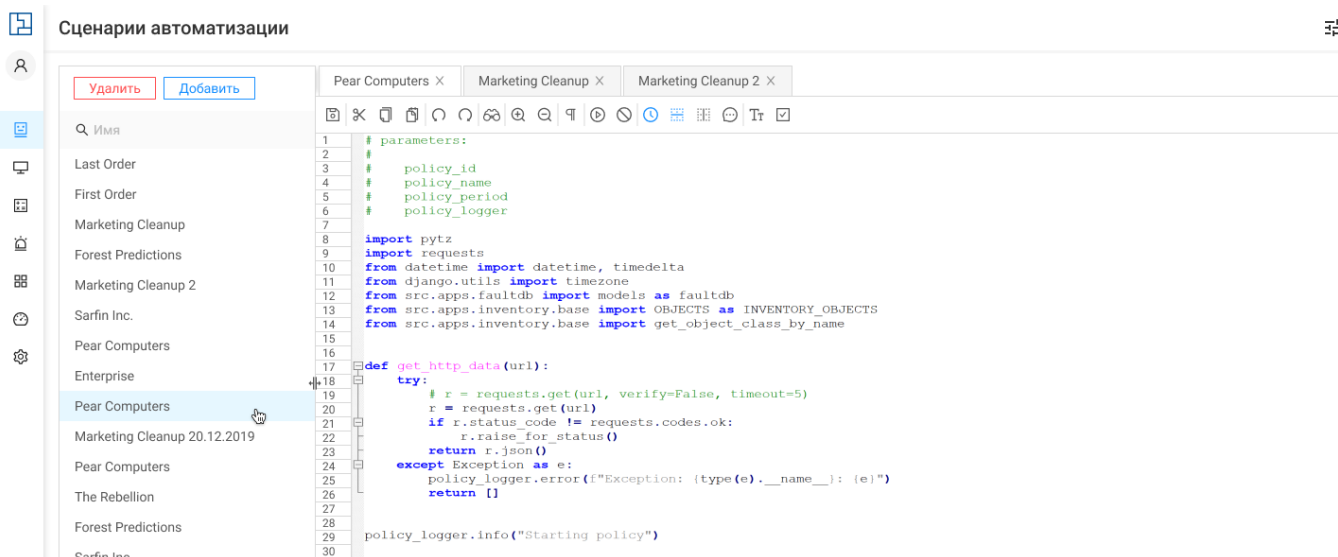


Рисунок 3 – Форма «Управление сценариями» создание сценария

Примечание: Помимо стандартных инструментов работы с текстом блок создания и редактирования кода содержит ряд специальных кнопок управления:



- сохранить изменения;



- выполнить код;



- прервать выполнение кода;



- отобразить журнал выполнения сценария;



- отобразить журнал под блоком ввода кода;



- отобразить журнал справа от блока ввода кода.

- 3) Нажмите на кнопку «Выполнить код».
- 4) Нажмите на кнопку «Отобразить журнал». После нажатия на кнопку будет открыт журнал с логом выполнения написанного кода.
- 5) Найдите в журнале сценария автоматизации ошибку и устраните ее.

Сценарии автоматизации

Удалить Добавить

Имя

Last Order

First Order

Marketing Cleanup

Forest Predictions

Marketing Cleanup 2

Sarfin Inc.

Pear Computers

Enterprise

Pear Computers

Marketing Cleanup 20.12.2019

Pear Computers

The Rebellion

Forest Predictions

Sarfin Inc.

Enterprise

Pear Computers

The Rebellion

Enterprise

Forest Predictions

The Rebellion

Forest Predictions

```
1 # parameters:
2 #
3 # policy_id
4 # policy_name
5 # policy_period
6 # policy_logger
7
8 import pytz
9 import requests
10 from datetime import datetime, timedelta
11 from django.utils import timezone
12 from src.apps.faultdb import models as faultdb
13 from src.apps.inventory.base import OBJECTS as INVENTORY_OBJECTS
14 from src.apps.inventory.base import get_object_class_by_name
15
16
17 def get_http_data(url):
18     try:
19         # r = requests.get(url, verify=False, timeout=5)
20         r = requests.get(url)
21         if r.status_code != requests.codes.ok:
22             r.raise_for_status()
23         return r.json()
24     except Exception as e:
25         policy_logger.error(f"Exception: (type(e).__name__): {e}")
26         return {}
27
28
29 policy_logger.info("Starting policy")
30
31
32 url_address = "http://93.188.43.212:4180/backend/api/"
```

Журнал выполнения сценария

```
2020-08-05 16:26:24,019 [INFO] === START POLICY (pid: 1509) ===
2020-08-05 16:26:24,298 [INFO] idb_sandbox_pgsqsl:
2020-08-05 16:26:24,298 [INFO] idb_sandbox_pgsqsl.jdbc_class = org.postgresql.Driver
2020-08-05 16:26:24,298 [INFO] idb_sandbox_pgsqsl.url = jdbc:postgresql://192.168.88.12:5432/srt_monitoring?
2020-08-05 16:26:24,298 [INFO] idb_sandbox_pgsqsl.login = srt
2020-08-05 16:26:24,299 [INFO] idb_sandbox_pgsqsl.password = Vjybnjh123
2020-08-05 16:26:24,299 [INFO] idb_sandbox_pgsqsl.jdbc_path =
2020-08-05 16:26:24,299 [INFO]
2020-08-05 16:26:24,316 [INFO] select id, name, port as db_port from automation_datasource
2020-08-05 16:26:24,316 [INFO] state = True
2020-08-05 16:26:24,316 [INFO] res1 = [{'id': 52, 'name': 'Inventory', 'db_port': 5432}, {'id': 53, 'name': 'Finance',
'db_port': 1234}, {'id': 54, 'name': 'idb_sandbox_pgsqsl', 'db_port': 5432}, {'id': 29, 'name': 'leroymlrin', 'db_port':
```

Рисунок 4 – Форма «Управление сценариями» журнал выполнения сценария

- 6) Нажмите на кнопку «Выполнить код».
- 7) Убедитесь, что в Журнале выполнения сценария ошибка устранена.
- 8) Нажмите кнопку «Сохранить изменения».



2.4 ОБНОВЛЕНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Для выполнения обновления Системы необходимо выполнение следующих условий:

- Все команды, приведенные в данном руководстве, должны быть выполнены пользователем ОС с правами суперпользователя;
 - Дистрибутив обновления Системы должен быть скачен на сервер.
1. Запустить командную строку.
 2. Выполнить запуск установочного файла `install.sh` указав в параметрах установки порт, на котором развернут действующий веб клиент Системы. (В примере ниже приведен порт 8080 может быть выбран любой другой не "занятый" порт.)

```
./install.nl --port8080
```

3. После запуска команды будет выполнено автоматизированное развертывание Системы и всех ее компонент.

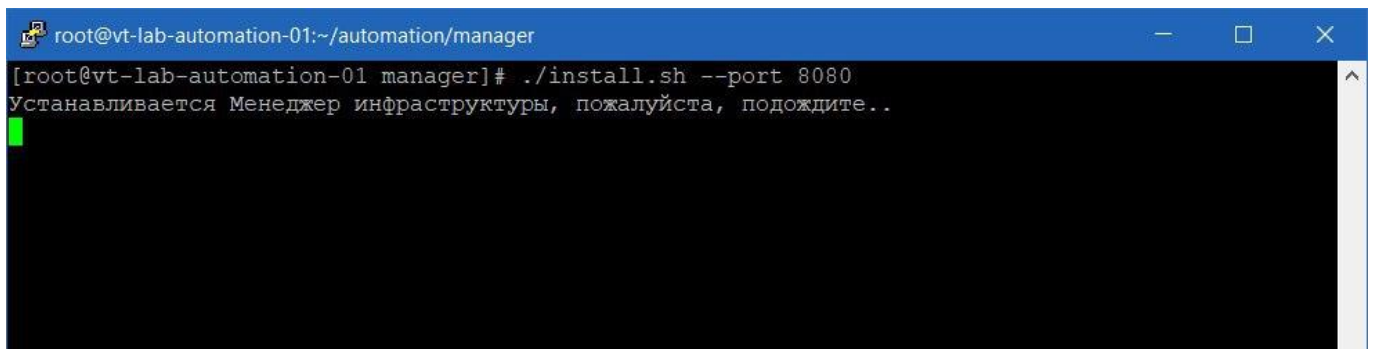


Рисунок 5 – Процесс обновления GIMS

4. После завершения установки в консоли будет отображена ссылка для перехода к веб клиенту Системы.