

УДК 004.01

***ПРОЕКТИРОВАНИЕ МОДЕЛИ СКРИПТА ДЛЯ СБОРА И АНАЛИЗА
ПРОДУКТОВЫХ МЕТРИК ИЗ СЕРВИСА УПРАВЛЕНИЯ ПРОЕКТАМИ***

Терехин Д.Е.,

магистрант,

Калужский государственный университет им. К.Э. Циолковского,

Калуга, Россия

Кряжева Е. В.

к.псих.н., доцент,

Калужский государственный университет им. К.Э. Циолковского,

Калуга, Россия

Раевский В.А.,

к.тех.н.,

Калужский государственный университет им. К.Э. Циолковского,

Калуга, Россия

Аннотация.

Статья рассматривает актуальные вопросы автоматизации процессов управления проектами в организации с применением специализированного программного обеспечения. Приводится обоснование выбора среды разработки PyCharm и библиотеки Psycopg2 для взаимодействия с выбранной реляционной базой данных PostgreSQL. Отмечаются плюсы формата YAML для конфигурирования параметров системы и возможные ограничения, связанные с особенностями синтаксиса. Описываются технические аспекты интеграции скрипта с существующей инфраструктурой, включая использование языка запросов JQL и организацию взаимодействия через API с повышенной защитой данных. Построена структурная диаграмма разработки скрипта и проведена ее

декомпозиция. В заключение сделан вывод о практической значимости предлагаемого решения для повышения эффективности управления рабочими процессами в рамках организации.

Ключевые слова: Jira, Scrum-мастер, скрипт, JQL (Jira Query Language), PyCharm, СУБД PostgreSQL.

DESIGNING A SCRIPT MODEL FOR COLLECTING AND ANALYZING PRODUCT METRICS FROM A PROJECT MANAGEMENT SERVICE

Terekhin D.E.

Undergraduate,

Kaluga State University named after K.E. Tsiolkovsky,

Kaluga, Russia

Kryazheva E. V.,

Candidate of Psychological Sciences, Associate Professor,

Kaluga State University named after K.E. Tsiolkovsky,

Kaluga, Russia

Raevsky V.A.,

Candidate of Technical Sciences,

Kaluga State University named after K.E. Tsiolkovsky,

Kaluga, Russia

Annotation.

The article examines the current issues of automation of project management processes in an organization using specialized software. The rationale for choosing the PyCharm development environment and the Psycpg2 library for interacting with the selected PostgreSQL relational database is provided. The advantages of the YAML format for configuring system parameters and possible limitations related to syntax features are

noted. The technical aspects of integrating the script with the existing infrastructure are described, including the use of the JQL query language and the organization of interaction through an API with enhanced data protection. A structural diagram of the script development is constructed and its decomposition is performed. In conclusion, it is concluded that the proposed solution is of practical importance for improving the efficiency of workflow management within the organization.

Keywords: Jira, Scrum Master, script, JQL (Jira Query Language), PyCharm, PostgreSQL database management system.

В настоящее время многие организации пользуются инструментами для управления проектами. Среди огромного множества систем наиболее популярными являются Comindware, Asana, Trello, Jira. Именно Jira разработанная компанией Atlassian используется в рассматриваемой организации. Данный продукт позволяет отслеживать рабочие процессы внутри фирмы. Scrum-мастера - проводят совещания, следят за соблюдением всех принципов Scrum, разрешают противоречия и защищают команду от отвлекающих факторов, проводят групповые обсуждения, митинги, дейли. Сбор большого количества информации для статистики, анализа и визуализации является сложным и затратным процессом, в котором нельзя допускать ошибки. Очень тяжело отслеживать огромное количество постоянно обновляющейся информации.

Разработка скрипта для сбора информации о задачах является актуальной проблемой. Главными требованиями к разрабатываемому скрипту являются автономная работа, мониторинг метрик, сбор метрик в базу данных:

Архитектура скрипта должна быть максимально масштабируемой и универсальной, что позволит просто и быстро добавлять нововведения и дополнительный функционал, если такой потребуется. Основная часть будет располагаться на сервере и обрабатывать JQL запросы, информация из которых

будет структурироваться и заноситься в базу данных, web - составляющая будет формировать таблицы и графики на основе собранных метрик.

Для взаимодействия скрипта с серверной частью Jira, помимо библиотеки, будут использоваться JQL запросы. JQL (*Jira Query Language*) - это язык запросов в Jira, предназначенный для поиска задач по заданным условиям. Взаимодействие с пользователем будет проходить по средствам web составляющей и независимо от операционной системы.

Разработка достаточно трудоёмкая, поэтому все этапы будут фиксироваться в репозитории GitLab (система управления версиями), для предотвращения потери кода. В последствии из репозитория будет производиться деплой (перенос проекта из системы управления версиями на сервер в определённую директорию) скрипта на сервер.

Скрипт должен производить авторизацию при отправке JQL запроса в Jira по средствам API токена для повышения безопасности и минимизации утечек данных во время передачи ответов.

Следующим шагом будет происходить проектирование базы данных и таблиц, в которые будут записываться метрики, также следует наладить подключение для возможности отправки запросов к базе по средствам библиотеки. Основой любого скрипта или приложения на Python является интерпретатор и подходящая IDE (интегрированная среда разработки). Также необходимо создать виртуальное окружение под проект, так как будет использоваться большое количество библиотек, которые понадобятся для реализации функционала [5].

В качестве среды разработки был выбран PyCharm. PyCharm предоставляет специальные функции автодополнения и анализа кода. При обнаружении ошибки она подсветится и будет предложен перечень возможных путей исправления. Автоматический рефакторинг поможет эффективно редактировать код. Также PyCharm имеет удобную и простую навигацию, которая позволит беспрепятственно перемещаться по проекту.

PyCharm предоставляет широкие возможности исправления и улучшения кода с помощью рефакторингов Rename и Delete, Extract Method, Inline Variable и многих других. Рефакторинги учитывают особенности конкретного языка или фреймворка, помогая вносить изменения по всему проекту. Также PyCharm обладает огромным инструментарием: встроенный отладчик и инструмент запуска тестов, полноценный встроенный терминал, инструменты для работы с базами данных. IDE интегрирована с популярными системами контроля версий, содержит встроенный SSH-терминал, поддерживает возможности удаленной разработки и удаленные интерпретаторы, а также интеграцию с Docker и другими.

Огромным плюсом является режим debug в котором можно построчно отслеживать действия кода, следить за объявлением переменных, результатами отработанных функций и использовать во многих других сценариях, которые понадобятся разработчику.

У Atlassian есть собственная документация по использованию библиотеки для работы с Jira по средствам API. Модуль называется Jira.

Для подключения к серверу будет использоваться специальный API ключ, который будет передаваться в запросе при создании клиента. После прохождения авторизации будет получен клиент, к которому можно обращаться.

Далее с помощью методов библиотеки скрипт будет вытаскивать список задач с определённой даты, структурировать данные и рассортировывать их по таблицам в базе данных.

На сегодняшний день существует множество систем управления базами данных - СУБД, от открытых до коммерческих, от реляционных до новомодных NoSQL и других. Одной из самых популярных реляционных систем баз данных является PostgreSQL [4].

Основой работы СУБД PostgreSQL является серверный процесс базы данных, работающий на одном сервере. Доступ из приложения к данным базы PostgreSQL будет производиться по средствам специального процесса базы

данных. То есть программа-клиент не может получать самостоятельный доступ к данным даже в том случае, если она развёрнута на том же ПК, где осуществляется серверный процесс. У каждого процесса есть своя локальная память. В ней располагается кэш каталога (информация, которая часто используется в базе данных), планы запросов, рабочее пространство для выполнения запросов и другое. Чтобы научить скрипт общаться с базой данных понадобится библиотека Psycopg2. Для её работы необходима установленная база данных PostgreSQL в облаке Яндекса.

Стандартные библиотеки Python не предоставляют встроенного функционала для работы с PostgreSQL, однако есть большое количество сторонних библиотек [3, 5]. Наиболее популярной из них является Psycopg2. Данная библиотека разработана на языке C, что позволяет ей быстро работать.

В качестве вспомогательного модуля, для получения конфигурационных параметров из специального файла с расширением YAML, будет задействована библиотека PyYAML.

YAML (*Yet Another Markup Language*) – текстовый формат для записи данных не являющийся языком разметки. Как и в Python, в YAML используются отступы для указания структуры документа. YAML использует пробелы и не использует знаки табуляции. Так же схожесть с Python прослеживается при добавлении комментария так как используется символ # и продолжаются до конца строки.

У YAML есть много плюсов и минусов. Его легко читать, можно легко научиться определять необходимые конструкции, но сложная спецификация может повлечь большое количество непредвиденных ошибок. Данная библиотека делает реализацию на разных языках несовместимыми друг с другом во многих тривиальных аспектах. Несмотря на эти особенности, YAML остается лучшим языком конфигурации.

Далее рассмотрим схематичное представление разработки скрипта на диаграмме IDEF0 (Рисунок 1).

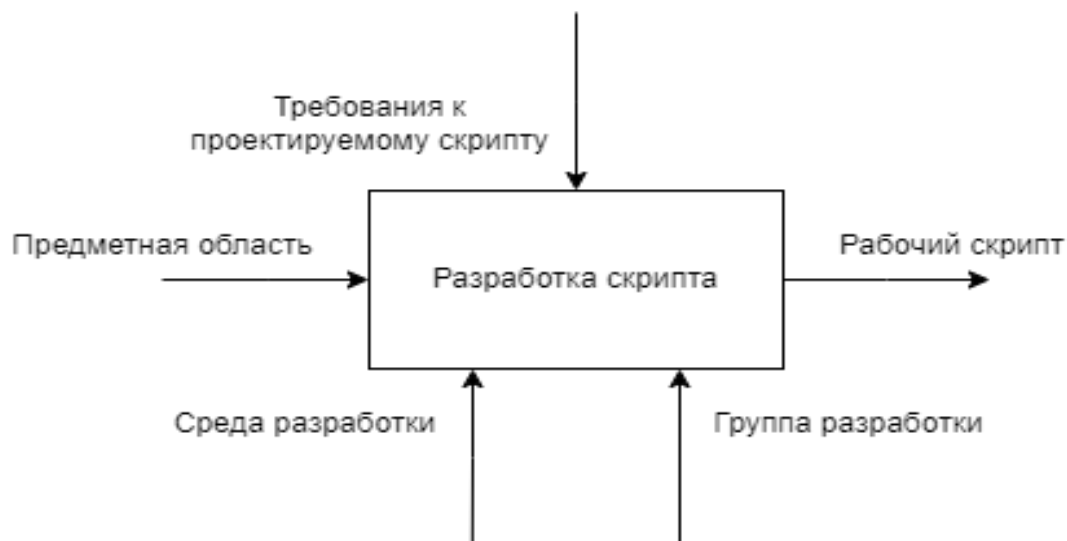


Рис.1 - Структурная диаграмма разработки скрипта (составлено авторами)

Вход:

- Предметная область

Управление

- Требование к проектируемому скрипту

Выход

- Рабочий скрипт

Механизм

- Среда разработки
- Группа разработки

С помощью декомпозиции рассмотрим более подробную диаграмму разработки. На рисунке 2 изображена декомпозиция разработки скрипта. Основные этапы: Запрос на создание скрипта для сбора метрик из jira, разработка ПО для сбора метрик из jira, создание виртуального окружения и написание кода, выбор, настройка и создание базы данных, подготовка и настройка сервера для разворачивания и работы скрипта, тестирование скрипта на локальной машине с

использованием тестовых таблиц, демонстрация функционала постановщику задачи, коммит в репозиторий и деплой наработки на сервер и проверка работоспособности.

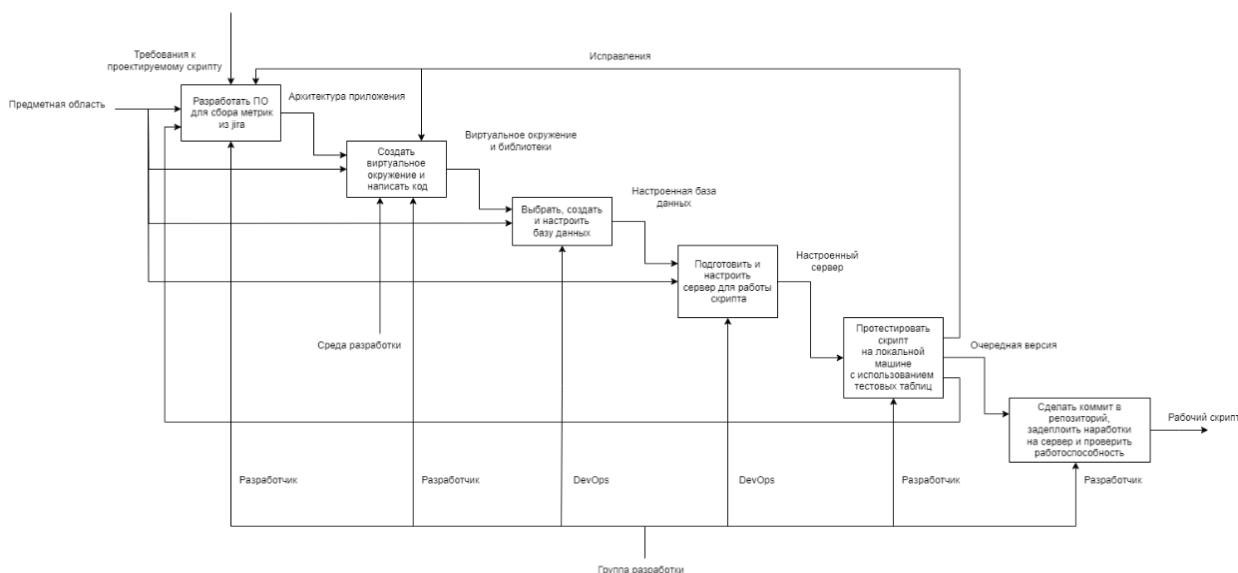


Рис.2 - Декомпозиция разработки скрипта (составлено авторами)

Проведем декомпозицию контекстной диаграммы и опишем последовательность разработки [1]:

Разработать ПО для сбора метрик из jira

- Управление: Требования к проектируемому скрипту
- Вход: Предметная область
- Механизм: Разработчик
- Выход: Архитектура приложения

Создать виртуальное окружение и написать код

- Управление: Исправления
- Вход: Предметная область, архитектура приложения
- Механизм: Разработчик, среда разработки
- Выход: Виртуальное окружение и библиотеки

Выбрать, создать и настроить базу данных

- Управление: -
- Вход: Предметная область, виртуальное окружение и библиотеки
- Механизм: DevOps
- Выход: Настроенная база данных

Подготовить и настроить сервер для работы скрипта

- Управление: -
- Вход: Предметная область, настроенная база данных
- Механизм: DevOps
- Выход: Настроенный сервер

Протестировать скрипт на локальной машине с использованием тестовых таблиц

- Управление: -
- Вход: Настроенный сервер
- Механизм: Разработчик
- Выход: Очередная версия

Сделать коммит в репозиторий, задеплоить наработки на сервер и проверить работоспособность

- Управление: -
- Вход: Очередная версия
- Механизм: Разработчик
- Выход: Рабочий скрипт

Таким образом, нами был рассмотрен опыт применения системы Jira от компании Atlassian, выделены сильные и слабые стороны используемого инструмента, проанализированы роли и обязанности Scrum-мастеров в процессе поддержания эффективных коммуникаций и координации действий команды.

Особый акцент сделан на сложностях, связанных с обработкой больших объемов постоянно обновляемой информации, необходимой для анализа и представления результатов рабочего процесса. Разработана модель скрипта,

предназначенного для автономного сбора и хранения метрик, формируемых в результате работы Jira.

Проведён отбор необходимых технических средств и инструментов, среди которых выделяются среда разработки PyCharm, база данных PostgreSQL и дополнительные библиотеки, обеспечивающие удобное взаимодействие с проектом и хранение данных. Показаны достоинства выбранного формата YAML для конфигурирования параметров системы, несмотря на некоторые потенциальные риски и ограничения. Предлагаемое решение представляет собой комплексный подход к управлению рабочими процессами и обеспечению аналитики на высоком уровне.

Библиографический список:

1. Григорьев, М. В. Проектирование информационных систем : учеб. пособие для вузов / М. В. Григорьев, И. И. Григорьева. - М. : Юрайт, - 2021. — 318 с. — (Высшее образование). — ISBN 978-5-534-01305-4.
2. Лаврищева, Е. М. Программная инженерия и технологии программирования сложных систем: учебник для вузов / Е. М. Лаврищева. // 2-е изд., испр. и доп. — М.: Юрайт, - 2021. — 432 с. — (Высшее образование). — ISBN 978-5-534-07604-2.
3. Любанович, Б. Простой Python. Современный стиль программирования = Introducing Python: Modern Computing in Simple Packages / Б. Любанович ; пер. с англ. - СПб. : Питер - 2016. — 480 с. — Серия «Бестселлеры O'Reilly».
4. Проектирование информационных систем: учебник и практикум для вузов / под общ. ред. Д. В. Чистова. - М. : Юрайт. - 2021. — 258 с. — (Высшее образование). — ISBN 978-5-534-00492-2.
5. Прохоренок, Н. А. Python 3 и PyQt: разработка приложений / Н. А. Прохоренок. - СПб.: БХВ-Петербург - 2012. — 703 с.

Оригинальность 76%