МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИЙ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

Рыбинский государственный авиационный технический университет   
имени «П.А. СОЛОВЬЕВА»

Факультет радиоэлектроники и информатики

Кафедра математического и программного обеспечения электронных вычислительных средств

Выпускная квалификационная работа

На тему

«Разработка быстроразвёртвываемого объектного хранилища»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Студент группы ИПБ-17 Ащеулов М.Р.

Руководитель Задорина Н.А.

Рыбинск 2021

**Содержание**

[Введение 2](#_Toc72247798)

[1. Описание проблемы 3](#_Toc72247799)

[2. Обзор аналогов 5](#_Toc72247800)

[3. Программная документация 6](#_Toc72247801)

[**a.** **Техническое задание на программное обеспечение** 7](#_Toc72247802)

[**b.** **Пояснительная записка к программному обеспечению** 8](#_Toc72247803)

[**c.** **Описание программы** 9](#_Toc72247804)

[**d.** **Программа и методика испытаний** 10](#_Toc72247805)

[4. Эксплуатационная документация на программный продукт 12](#_Toc72247806)

[5. Акт испытаний программного продукта 13](#_Toc72247807)

[6. Экономическое обоснование 14](#_Toc72247808)

[7. Заключение 15](#_Toc72247809)

[8. Список литературы 16](#_Toc72247810)

[9. Приложения 17](#_Toc72247811)

# Введение

С каждым годом, количество цифровых данных неумолимо растёт. Пользователи всё больше делают фотографий, записывают видео, создают документов, и т.д. Только в 2020 году во всём мире было создано и скопировано более 59 зетабайта или 59 триллионов гигабайтов данных. 44 из них являются неструктурированными.

Для решения проблемы организации и хранения такого большого количества неструктурированных данных появилась концепция объектного хранилища. В отличие от файлов в файловой системе, объекты хранятся в плоской структуре. Есть просто пул объектов: никаких папок, каталогов или иерархий. Для получения объекта необходимо просто сделать запрос к объектному хранилищу, предоставив идентификатор этого объекта.

В современном мире каждый человек где-то хранит персональные данные: в облаке, на флеш-накопителе или жестком диске компьютера. Сейчас, в эпоху развития компьютерных технологий, практически в любом виде деятельности необходимо передавать данные другим людям или машинам и хранить эти данные. Системы хранения и передачи информации должны быть удобны как для пользователя, так и для обслуживающего персонала или машины. Есть несколько вариантов решения этих проблем:

1. Самый безопасный, передать данные на флеш-накопителе или защищённом диске лично в руки получателю
2. Самый простой и удобный, разместить файлы в облачном хранилище, предлагаемом различными провайдерами

Быстроразвёртываемое объектное хранилище в свою очередь совмещает в себе безопасность физических накопителей и удобство использования облачных систем хранения.

Для того чтобы начать работу необходимо будет просто запустить приложение на локальном устройстве и указать путь до корневой директории, где будет развёрнуто объектное хранилище. Пользователь взаимодействует с объектным хранилищем через специальный интерфейс. Интерфейс взаимодействует с объектным хранилищем посредством API.

# Описание проблемы

Неструктурированные данные – данные, которые не соответствуют заранее определённой модели данных и не имеют легко идентифицируемой структуры. Вследствие этого их нелегко использовать в компьютерных программах. Неструктурированные данные не организованны заранее определённым образом или не имеют заранее определённой модели данных, что делает их неудобными для размещения в реляционной базе данных. Сегодня крупнейшие и не только предприятия осознают ценность и важность данных на всех этапах деятельности и жизненного цикла. В различных областях – в маркетинге, в медицине и здравоохранении, в медиа-сервисах и т.д. – неструктурированные данные крайне ценны тем, что находятся новые способы их анализа, изменения и форматирования. К примеру:

1. Медицинские учреждения применяют различные методы анализа архивных данных для прогнозирования хода болезни
2. Автоконцерны исследуют особые данные с датчиков автомобилей, чтобы предугадать вероятные неисправности
3. Телекомпании повторно используют передачи, фильмы, сериалы для показа. Чтобы их повторно не обрабатывать под требуемый формат их необходимо хранить в надёжном месте

Но ценность такие данные имеют только при условии экономного хранения, удобного доступа и безопасности от «утечек».

Уже существует механизм хранения, который люди используют с появления современных компьютеров − файловая система. Но когда речь идёт о неструктурированных данных, важно понимать, что система, используемая для хранения данных, должна очень хорошо масштабироваться. Однако масштабировать файловую систему это сложная задача. Вам требуется не только управлять ненужными метаданными и иерархией, которые навязывают файловые системы, но также необходимо учитывать такие аспекты обслуживания, как управление резервными копиями.

Недостаточно просто собрать и хранить неструктурированные данные. Также необходимо применить некоторый уровень организации, чтобы разобраться в них. Такие методы, как анализ текста, автоматическая категоризация и автоматическая пометка, имеют решающее значение для получения бизнес-смысла для всех неструктурированных данных, которые вы собираете. Этого сложно добиться с файловыми системами, потому что они имеют фиксированные слои или расположение.

Для решения этой проблемы появились объектные хранилища.

Объектное хранилище представляет собой стратегию управления и манипуляции хранилищем данных как отдельными единицами, объектами. Эти объекты хранятся в одном хранилище и не интегрируются в файлы, находящиеся в других папках. Вместо этого хранилище объектов объединяет фрагменты данных, из которых состоит файл, добавляет в него все соответствующие метаданные и прикрепляет пользовательский идентификатор.

Объектное хранилище добавляет в файл полные метаданные, устраняя многоуровневую структуру файлового хранилища, и помещает все в плоское адресное пространство — пул хранилища. Именно метаданные являются ключом к успеху объектного хранилища: они обеспечивают глубокий анализ использования и функций данных в пуле.

Очень важно, чтобы данные были доступны через HTTP(S), чтобы обеспечить лёгкий доступ к файлу. Затем его можно подвергнуть анализу или другим методам. Объектное хранилище справляется с этим хорошо. Почти все объектные хранилища имеют REST API, которое позволяет крайне просто получить доступ к файлам через HTTP(S).

REST API полезно не только для получения доступа к объектам, оно позволяет авторизоваться и получать токен сессии, получать свойства файла, управлять разрешениями и т.д., в общем всё, что необходимо будет делать вручную в файловой системе.

Изображение

Важным аспектом любых систем хранения является обработка метаданных. Объектное хранилище обеспечивает большую гибкость, поскольку метаданные объекта изначально не определены. Метаданные не ограничиваются тем, что система хранения считает важным. Имеется возможность добавить любой тип или количество метаданных. Объектное хранилище позволяет назначить тип приложения, с которым связан объект; важность для приложения; уровень защиты данных, который требуется конкретному объекту. И так далее, возможности безграничны.

Объектное хранилище обладает такими свойствами как:

**Масштабируемость**. Объектное хранилище может содержать практически любое количество данных без необходимости в разбиении набора данных на разделы.

**Эффективность.** Отсутствие иерархии означает отсутствие узких мест, возникающих вследствие использования сложных систем каталогов.

**Доступность.** Объектные системы хранения имеют механизмы для сохранения целостности данных, обеспечивают репликацию данных, последовательные обновления и отсутствие простоев.

# Обзор аналогов

Несмотря на то, что реализаций объектных хранилищ существует огромное множество, почти все они являются облачным решением и хранят персональные данные пользователей на собственных серверах. Быстроразвёртываемое объектное хранилище обладает уникальной особенностью – запуск и работа на локальной машине или устройстве, что позволяет вам настроить его в локальной сети без доступа к интернету и ограничить возможный несанкционированный доступ извне. Так же важным преимуществом является цена, а точнее ценообразование. Если при использовании облачных решений вы платите за обслуживание, пользование и хранение данных, то в случае с локальным быстроразвёртываемым хранилищем вы платите исключительно за «железо» или выбранному Вами провайдеру по утверждённому тарифу. Для наглядности я приведу таблицу сравнения цены пользования облачным решением от некоторых компаний за год и стоимость системы с накопителями того же объёма.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Amason s3 | MinIO | Быстроразвёртываемое объектное хранилище |
| Объём накопителей | 8ТБ | 8ТБ | 8ТБ |
| Количество GET запросов | 10000 | Не ограничено | Не ограничено |
| Количество PUT, COPE, POST запросов | 100 | Не ограничено | Не ограничено |
| Обслуживание | Включено в стоимость | Ручное (1 человек) | Ручное (1 человек) |
| Размещение | В облаке | В облаке или локально | В облаке и/или локально |
| Цена | 2329.8 USD/год | 960 USD/год за облачное решение и 500 USD за комплект оборудования | 500 USD за комплект оборудования  Или облачное размещение по цене тарифа провайдера, у которого будет размещено |

Из таблицы видно, что Быстроразвёртываемое объектное хранилище выигрывает в цене в долгосрочной перспективе и возможностях конфигурации системы: всё зависит от вашего бюджета и требований. А за счёт простоты настройки и использования с обслуживанием справится обычный пользователь.

# Программная документация

* Текст

## **Техническое задание на программное обеспечение**

* Текст

## **Пояснительная записка к программному обеспечению**

* Текст

## **Описание программы**

* Текст

## **Программа и методика испытаний**

Для тестирования API будет использовано специализированное программное обеспечение Postman.

Программа должна корректно обрабатывать и отвечать на следующие запросы:

1. GET /api/get/{uid}
2. POST /api/post -F file=”/path/to/file/”
3. PUT /api/put/{uid} -F file=”/path/to/file”
4. DELETE /api/delete/{uid}
5. GET /api/auth -H “email=email” -H “password=password”
6. POST /api/obj/{uid}/set-meta --data ‘{“key”:”value”}’
7. POST /api/obj/{uid}/add-meta --data ‘{“key”:”value”}’
8. GET /api/obj/{uid}/get-all-meta
9. GET /api/obj/{uid}/get-meta/key?={key}

|  |  |  |
| --- | --- | --- |
| Номер запроса | Описание запроса | Ожидаемый результат |
| 1 | Получение объекта по уникальному идентификатору | Объект соответствующий уникальному идентификатору |
| 2 | Загрузка в объектное хранилище нового объекта | Объект загружен и получен ответ об успешном завершении операции |
| 3 | Обновление объекта в объектном хранилище | Старый объект с соответствующим идентификатором удалён, новый загружен |
| 4 | Удаление объекта в объектном хранилище | Требуемый объект удалён |
| 5 | Аутентификация и получение токена доступа | Токен доступа |
| 6 | Установление метаданных для объекта | Для объекта установлены метаданные |
| 7 | Добавление параметра к метаданным | Добавлен параметр в конец списка для соответствующего объекта |
| 8 | Получение всех метаданных | Все метаданные |
| 9 | Получение значения метаданных по ключу | Значение соответствующее определённому ключу |

# Эксплуатационная документация на программный продукт

* Текст

# Акт испытаний программного продукта

* Текст

# Экономическое обоснование

* Текст

# Заключение

* Текст

# Список литературы

* Текст

# Приложения

* Текст