**Управление качеством программных систем.**

**Супрунов Матвей, ПИ21-3**

**ЛАБОРАТОРНАЯ РАБОТА №1 (7).**

Тема лабораторной работы: тестирование документации.

**Документация к микросервисному приложению по потоковой обработке данных**

Цель работы – анализ документации, описывающей требования к проектированию или разработке программного обеспечения или структуре базы данных. На основе проведенного анализа будет составлен отчет, включающий оценку качества документации, выявленные несоответствия и рекомендации по улучшению.

**Описание тестируемой документации**

Для анализа была выбрана документация проекта микробатчевой обработки данных ресторана. Документация состоит из нескольких разделов, каждый из которых играет ключевую роль в описании функциональности, архитектуры и требований к проекту.

1. *Требования бизнеса*: Этот раздел включает в себя описание глобальных целей и потребностей проекта с точки зрения бизнеса. Он объясняет, зачем проект нужен, какие задачи он должен решать и какие выгоды принесет.
2. *Функциональные и нефункциональные требования:* В этом разделе описаны ключевые операции стриминг-сервиса с данными, такие как десериализация, фильтрация, дедупликация, трансформация, обогащение, сериализация и таргетирование. Также указаны технические особенности решения, включая допустимую временную задержку, пропускную способность системы, масштабируемость, отказоустойчивость и допустимую потерю данных.
3. *Пользовательские требования*: Этот раздел содержит рамочно сформулированные задачи по настройке системы для достижения бизнес-целей. Включены инструкции по использованию системы и взаимодействию с ней.
4. *Структура данных*: Включает описание формата входных данных (JSON), структуры слоев хранилища данных (STG, DDS, CDM) и атрибутов, обозначающих значимые бизнес-метрики.
5. *Описание микросервисов и их взаимодействия*: В этом разделе описаны микросервисы, их функции и взаимодействие между собой. Указано, как микросервисы обмениваются данными и какие задачи решают.
6. *Описание инфраструктуры и процессов контейнеризации и оркестрации*: Включает описание процессов контейнеризации (создание Dockerfile, построение и тестирование Docker-образов) и оркестрации (развертывание микросервисов в Kubernetes, настройка авто-масштабирования и мониторинга).

**Описание критериев качества тестируемой документации**

Для оценки качества документации были определены следующие критерии:

* Полнота: Документация должна полностью охватывать все аспекты проекта, обеспечивая полное понимание системы.
* Точность: Информация в документации должна быть точной и достоверной, без двусмысленностей и противоречий.
* Четкость и ясность: Документация должна быть написана понятным языком, с четкой структурой и логичным изложением.
* Соответствие стандартам: Документация должна соответствовать общепринятым стандартам в области разработки программного обеспечения и баз данных.
* Обоснованность требований: Все требования должны быть четко обоснованы, с указанием причин и бизнес-логики.
* Актуальность: Документация должна быть актуальной и соответствовать текущему состоянию проекта.

**Описание и обоснование метода тестирования документации**

Метод тестирования документации включал несколько этапов, каждый из которых был направлен на оценку различных аспектов документации.

Первым этапом было тщательное чтение и понимание документации. Этот шаг позволил ознакомиться с содержанием и структурой документации, выявить основные разделы и понять, какие аспекты проекта она охватывает.

На втором этапе проводился анализ документации на соответствие установленным критериям качества. Проверялись полнота информации, точность описаний, четкость и ясность изложения, соответствие стандартам, обоснованность требований и актуальность данных.

Третий этап включал идентификацию несоответствий. Были выявлены части документации, которые не соответствовали критериям качества. Для этого проводилось сравнение содержимого документации с критериями и стандартами, а также выявлялись двусмысленные или противоречивые утверждения.

На четвертом этапе разрабатывались рекомендации по улучшению документации. Формулировались конкретные предложения по исправлению выявленных несоответствий, включая уточнение описаний, добавление недостающей информации, улучшение структуры и актуализацию данных.

В результате проведенного анализа была составлена подробная картина текущего состояния документации, выявлены основные проблемы и предложены пути их решения. Это позволит значительно улучшить качество документации и, соответственно, повысить эффективность разработки и сопровождения проекта.

**Список несоответствий в документации критериям качества**

**Полнота информации**

**Номер пункта**: Описание слоев хранилища данных (STG, DDS, CDM)  
**Несоответствие**: Описание слоев DDS и CDM недостаточно подробное. В документации отсутствуют примеры данных и схемы хранения, что затрудняет понимание процесса обработки данных.  
**Рекомендация к исправлению**: Добавить детальное описание каждого слоя, включая примеры данных и схемы хранения. Например, в DDS слой можно добавить описание структуры таблиц, используемых для обогащения данных, и пример набора данных до и после обработки.

**Ясность и понятность**

**Номер пункта**: Описание взаимодействия микросервисов  
**Цитата из документации**: "Микросервисы обмениваются данными через тематическую очередь."  
**Несоответствие**: Утверждение неясное и может вызвать недопонимание. Нет информации о типе данных и формате, в котором они передаются между микросервисами.  
**Рекомендация к исправлению**: Уточнить, какие данные передаются между микросервисами и в каком формате. Например: "Микросервисы обмениваются данными в формате JSON через тематическую очередь Kafka. Данные включают идентификаторы заказов, статусы и временные метки."

**Точность и достоверность**

**Номер пункта**: Описание требований к порогам категорий  
**Цитата из документации**: "Для категорий надо завести счётчики с конкретными порогами."  
**Несоответствие**: Не указаны конкретные значения порогов и обоснование их выбора.  
**Рекомендация к исправлению**: Обосновать выбор пороговых значений, указав бизнес-логику или анализ данных. Например: "Для каждой категории установлены пороговые значения на основе анализа исторических данных о заказах и сезонных трендов. Порог для категории 'напитки' составляет 100 заказов в месяц."

**Структурированность и логичность**

**Номер пункта:** Описание обработки данных в микросервисах  
**Цитата из документации:** "Обогащение данных в DDS слое."  
**Несоответствие**: Фраза недостаточно точна и может быть интерпретирована по-разному. Неясно, какие именно данные добавляются и с какой целью.  
**Рекомендация к исправлению:** Уточнить, какие данные добавляются на этапе обогащения и с какой целью. Например: "На этапе обогащения в DDS слой добавляются данные о времени приготовления и доставке, чтобы улучшить точность анализа времени выполнения заказов. Например, добавляется атрибут 'время приготовления' с форматом HH:MM

**Номер пункта**: Описание структуры данных CDM  
**Цитата из документации**: "В CDM нужны две витрины."  
**Несоответствие**: Недостаточно информации о структуре витрин и их содержимом.  
**Рекомендация к исправлению**: Подробно описать структуру каждой витрины, включая ключевые метрики и атрибуты. Например: "Первая витрина в CDM содержит счетчики заказов по блюдам, включая атрибуты: идентификатор блюда, количество заказов, среднее время приготовления. Вторая витрина содержит счетчики заказов по категориям товаров, включая атрибуты: идентификатор категории, количество заказов, среднее время доставки."

**Удобство поддержки и обновлений**

**Номер пункта**: Описание инструментов для контейнеризации и оркестрации  
**Несоответствие**: Описание инструментов контейнеризации и оркестрации устарело и не учитывает последние версии и возможности.  
**Рекомендация к исправлению**: Регулярно обновлять раздел документации, связанный с инструментами контейнеризации и оркестрации, чтобы он соответствовал последним версиям и тенденциям. Например, включить описание использования Docker Compose для локального тестирования и Helm для управления Kubernetes.

**Доступность и удобство использования**

**Номер пункта**: Описание API микросервисов  
**Несоответствие**: Документация имеет неудобный формат представления информации, что затрудняет ориентирование в описании API.  
**Рекомендация к исправлению**: Обеспечить легкий доступ к документации в нескольких удобных форматах, таких как Swagger для интерактивного документации API, PDF-версии и онлайн-ресурс. Также следует включить примеры запросов и ответов для каждого API-метода. Например, для метода получения информации о заказе можно привести пример запроса: GET /api/orders/{id} и соответствующий ответ: {"id": "123", "status": "CLOSED", "items": [{"name": "burger", "quantity": 2}]}.

**Выводы по работе**

В результате проведенного анализа документации проекта микробатчевой обработки данных ресторана были выявлены недочеты, неточности и ошибки, не соответствующие критериям качества документации. Основные проблемы связаны с неполнотой информации, неясностью и отсутствием четкости в описаниях, недостаточной точностью и достоверностью данных, а также устаревшими сведениями об инструментах и подходах. Для улучшения качества документации рекомендуется добавить подробные описания, уточнить формулировки, обосновать требования и регулярно обновлять информацию. Эти меры помогут сделать документацию более понятной, точной и актуальной, что, в свою очередь, повысит эффективность разработки и сопровождения проекта.

**Список использованных источников**

1. IEEE Standard for Software Requirements Specifications (IEEE 830-1998).
2. "Writing Effective Use Cases" by Alistair Cockburn.
3. "The Art of Scalability" by Martin L. Abbott and Michael T. Fisher.
4. Docker and Kubernetes official documentation