**Техническая архитектура сервиса информации в составе инструмента проектирования и архитектурного анализа**

1. **Тема**

В связи с необходимостью наличия в составе инструмента проектирования и архитектурного анализа автономного сервиса информации (см. документ “Функциональная архитектура инструмента проектирования и архитектурного анализа”) представляется необходимым создание проекта технической архитектуры такого сервиса в целях:

* определения потребностей в характере, составе и особенностях взаимодействия механизмов, необходимых для реализации сервиса информации
* определения технических средств, необходимых для реализации таких механизмов
* оценки пригодности существующих технических средств для использования в качестве или в составе механизмов сервиса информации
* выбора конкретных технических средств и характера их применения для реализации сервиса информации
* разработки концептуального прототипа сервиса в случае признания невозможности использования в качестве такового существующих технических средств
* дальнейшей разработки полноценного технического решения в случае признания принципиальной возможности создания такого сервиса, удовлетворительности издержек на разработку такого сервиса по итогам разработки концептуального прототипа

Целями настоящего документа являются:

* формулирование и описание бизнес-требований, функциональных требований, технических требования к сервису информации
* рассмотрение существующих технических средств на предмет соответствия таким требованиям
* анализ возможных решений с учетом полного либо частичного использования существующих инструментов либо без использования таковых
* выбор наиболее подходящего технического решения по результатам анализа
* описание технической архитектуры выбранного решения, включая описание составляющих его компонентов (механизмов), их взаимосвязей между собой и с внешней средой (пользователями, внешними сервисами)
* описание рекомендаций по непосредственной технической реализации механизма, включая рекомендации по использованию конкретных технических средств, использованию конкретной модели данных

1. **Глоссарий**

| **Термин** | **Определение** | **Примечание** |
| --- | --- | --- |
| ПО | Программное обеспечение |  |
| СА | Средство автоматизации |  |
| Проектное решение/Архитектура | Структура разрабатываемого программного обеспечения, включающая его компоненты, внутренние (между компонентами) и внешние (с пользователями и/или прочим ПО) взаимосвязи, а также описание таковых |  |
| Проектирование | Разработка и/или модификация проектного решения | подразделяется на функциональное и техническое (см. ниже) |
| Функциональная архитектура | Архитектура ПО с точки зрения реализуемых им функций безотносительно их технической реализации |  |
| Техническая архитектура | Архитектура ПО с точки зрения используемых в его составе технических средств |  |
| Целевая архитектура | Архитектура ПО согласно проектному решению | может быть как функциональной, так и технической |
| Фактическая архитектура | Архитектура ПО, реализованная на практике |  |
| Архитектурный анализ | Процесс выявления и описания фактической архитектуры, её отличий от целевой архитектуры (в случае наличия таковой), а также результат такого процесса |  |

1. **Требования**

На основании задач сервиса информации (см. документ “Функциональная архитектура инструмента проектирования и архитектурного анализа”, раздел 6) представляется необходимым выдвинуть к сервису информации следующие требования (таблица 1):

| **Требование** | **Описание** | **Примечание** |
| --- | --- | --- |
| **3.1 Бизнес-требования** | | |
| **3.1.1 Стоимость\*** | Стоимость поиска или разработки, внедрения, эксплуатации средства автоматизации не должна быть избыточной | избыточной предлагается считать стоимость, превышающую ожидаемые издержки, связанные с последствиями проблем проектирования, с учетом возможности тиражирования и продажи |
| **3.1.2 Безопасность\*** | Защита хранимой информации от несанкционированного доступа | информация о собственных разработках, хранимая в базе знаний, представляет коммерческую тайну и требует защиты от утечек |
| **3.1.3 Гибкость\*** | возможность использования инструмента в различных проектах без необходимости модификации под конкретный проект |  |
| **3.1.4 Простота эксплуатации** | Средство проектирования не должно вызывать избыточных затруднений при использовании | следует учесть, что функции хранения, поиска и предоставления доступа к информации предполагаются к использованию не только архитекторами |
| **3.1.5 Тиражируемость** | возможность распространения и продажи | в случае собственной разработки в целях извлечения прибыли и покрытия издержек |
| **3.1.6 Доступность технических средств** | отсутствие значительных ограничений нетехнического характера на использование конкретных технических средств и/или технологий в качестве или в составе сервиса | к таким ограничениям могут относиться юридические, финансовые, кадровые (наличие необходимых специалистов) и т.д. |
| **3.2 Функциональные требования** | | |
| **3.2.1 Хранение информации\*** | возможность сбора и централизованного хранения формализованной архитектурной информации | описание моделей сервисов, механизмов и компонентов механизмов, их взаимосвязей между собой в рамках других различных сервисов и механизмов, документация к вышеперечисленному, также информация о взаимосвязях моделей, документации и готовых технических решений |
| **3.2.2 Поиск и предоставление доступа к информации\*** | быстрый поиск необходимой информации по различным критериям, предоставление удобного доступа ко всей необходимой информации в соответствии с ролевой моделью | опционально поиск может включать автоподсказки |
| **3.3 Технические требования** | | |
| **3.3.1 Отказоустойчивость\*** | бесперебойная работа сервиса в рамках предполагаемых нагрузок | стандартное требование |
| **3.3.2 Масштабируемость\*** | возможность увеличения объема хранимой и используемой информации без необходимости изменения архитектуры, значительного изменения состава механизмов сервиса |  |
| **3.3.3 Быстродействие\*** | обеспечение удовлетворительного быстродействия всех процессов работы с информацией | в первую очередь - поиска |
| **3.3.4 Отсутствие избыточной функциональности** | отсутствие чрезмерного количества избыточных функций | стандартное требование - избыточная функциональность усложняет использование технического средства как в процессе разработки, так и в процессе эксплуатации |
| **3.3.5 Универсальность средств работы с информацией\*** | возможность хранить, управлять, использовать в поиске информацию различных типов |  |
| **3.3.6 Гибкость работы с информацией** | возможность внесения изменений в модель данных, инструменты управления данными, средства поиска данных без изменения архитектуры сервиса, с минимальным изменением технической реализации сервиса |  |
| **3.3.7 Формализуемость хранимой информации\*** | возможность хранения информации в формализованном виде в целях упрощения упорядочения и автоматической обработки информации |  |
| **3.3.8 Технологическая гибкость\*** | возможность использования в составе сервиса различных механизмов и технологий с потенциальной возможностью механизмов в процессе эксплуатации (в т.ч. с заменой технологий, использованных для реализации механизмов) |  |
| **3.3.9 Универсальность средств интеграции\*** | использование средств интеграции, позволяющих осуществлять взаимодействие с внешними сервисами вне зависимости от технологий и технических средств, использованных для реализации внешних сервисов |  |
| **3.3.10 Гибкость пользовательских интерфейсов** | возможность изменения пользовательских интерфейсов с минимальными затратами | связано с 3.3.6 |

Таблица 1. Требования к сервису информации

\* - обязательное требование

1. **Существующие технические решения**

В рамках изучения существующих проектов и готовых решений были выявлены и проанализированы на предмет соответствия требованиям следующие решения (таблицы 2, 3):

| **Решение** | **Описание** | **Степень соответствия требованиям** | **Заключение** |
| --- | --- | --- | --- |
| **4.1 Средство управления проектами YouTrack** | база знаний в составе средства управления проектами YouTrack | **0.56** | много избыточной функциональности, мало технологической гибкости, хуже формализуемость информации ,платное стороннее решение |
| **4.2 Atlassian Confluence** | средство для ведения базы знаний | **0.53** | аналогично 4.4 |
| **4.3 Технологическая платформа 1С:Предприятие** | фреймворк для создания ERP-решений, включая хранение и управление большими объемами данных | **0.73** | ограниченная технологическая гибкость, ограниченное соответствие 3.1.5, 3.1.6, платное стороннее решение |

Таблица 2. Существующие технические решения. Общая информация. Степень соответствия требованиям оценивается по 10-балльной шкале на основе средней оценки по таблице 3

| **Требование/Решение** | **4.1** | **4.2** | **4.3** |
| --- | --- | --- | --- |
| **3.1.1** | 1 | 1 | 1 |
| **3.1.2** | 2 | 2 | 2 |
| **3.1.3** | 1 | 1 | 1 |
| **3.1.4** | 1 | 1 | 1 |
| **3.1.5** | 0 | 0 | 0 |
| **3.1.6** | 1 | 1 | 0 |
| **3.2.1** | 2 | 2 | 2 |
| **3.2.2** | 2 | 2 | 2 |
| **3.3.1** | 2 | 2 | 2 |
| **3.3.2** | 2 | 2 | 2 |
| **3.3.3** | 2 | 2 | 2 |
| **3.3.4** | 0 | 0 | 0 |
| **3.3.5** | 1 | 1 | 2 |
| **3.3.6** | 2 | 2 | 2 |
| **3.3.7** | 0 | 0 | 2 |
| **3.3.8** | 0 | 0 | 1 |
| **3.3.9** | 0 | 0 | 1 |
| **3.3.10** | 1 | 0 | 1 |
| **Итого (средняя от 0 до 1)** | **0.56** | **0.53** | **0.67** |

Таблица 3. Соответствие существующих решений требованиям. Степень соответствия требованиям оценивается по 3-балльной шкале (0 - не соответствует, 1- соответствует частично, 2 - соответствует полностью)

Заключение: ни одно из рассмотренных существующих решений не соответствует требованиям (включая обязательные) в удовлетворительной степени.

Примечание: голые СУБД в этом разделе не рассматриваем, иначе он раздуется до бесконечности

1. **Анализ возможных решений**

Исходя из выдвинутых бизнес-требований, функциональных требований и технических требований, результатов анализа готовых решений представляется необходимым рассмотреть следующие возможные решения:

**5.1 Существующее решение с готовыми пользовательским интерфейсом и моделью данных.** Поиск и использование существующего решения с готовыми интерфейсом и моделью данных

**5.2 Существующее решение с готовыми модифируемыми пользовательским интерфейсом и моделью данных.** Поиск и использование существующего решения с готовыми интерфейсом и моделью данных с возможностью их модификации

**5.3 Разработка решения с готовым пользовательским интерфейсом и моделью данных с использованием готового средства управления данными.** Разработка собственного решения с готовыми интерфейсом и моделью данных с использованием готового средства управления данными

**5.4 Разработка решения с готовыми модифируемыми пользовательским интерфейсом и моделью данных с использованием готового средства управления данными.** Разработка собственного решения с готовыми интерфейсом и моделью данных с возможностью их модификации с использованием готового средства управления данными

**5.5 Разработка решения с готовым пользовательским интерфейсом и моделью данных с использованием собственного средства управления данными.** Разработка собственного решения с готовыми интерфейсом и моделью данных, включая разработку средства управления данными

**5.6 Разработка решения с готовыми модифируемыми пользовательским интерфейсом и моделью данных с использованием собственного средства управления данными.** Разработка собственного решения с готовыми интерфейсом и моделью данных с возможностью их модификации, включая разработку средства управления данными

Примечание: механизмы разработки интерфейсов - готовый или собственный, рассматривать не будет. Готовый очевидно удобнее. Можно для красоты рассмотреть в презентабельной версии документа

Сравнительный анализ возможных решений (таблица 4):

| **Решение/Требование** | **5.1** | **5.2** | **5.3** | **5.4** | **5.5** | **5.6** |
| --- | --- | --- | --- | --- | --- | --- |
| **3.1.1** | 5 | 3 | 6 | 4 | 1 | 1 |
| **3.1.2** | 6 | 5 | 9 | 8 | 10 | 9 |
| **3.1.3** | 5 | 5 | 5 | 9 | 7 | 10 |
| **3.1.4** | 5 | 5 | 10 | 10 | 10 | 10 |
| **3.1.5** | 2 | 2 | 8 | 8 | 9 | 9 |
| **3.1.6** | 5 | 5 | 10 | 10 | 10 | 10 |
| **3.2.1** | 7 | 7 | 10 | 10 | 10 | 10 |
| **3.2.2** | 7 | 7 | 10 | 10 | 10 | 10 |
| **3.3.1** | 5 | 5 | 9 | 9 | 7 | 7 |
| **3.3.2** | 5 | 5 | 9 | 9 | 7 | 7 |
| **3.3.3** | 5 | 4 | 9 | 8 | 7 | 6 |
| **3.3.4** | 9 | 8 | 10 | 9 | 10 | 9 |
| **3.3.5** | 5 | 6 | 7 | 9 | 5 | 7 |
| **3.3.6** | 4 | 6 | 4 | 9 | 4 | 9 |
| **3.3.7** | 5 | 5 | 9 | 9 | 9 | 9 |
| **3.3.8** | 3 | 3 | 5 | 9 | 5 | 10 |
| **3.3.9** | 5 | 5 | 10 | 10 | 10 | 10 |
| **3.3.10** | 5 | 6 | 6 | 9 | 6 | 9 |
| **Итого** | **93** | **92** | **146** | **159** | **137** | **152** |

Таблица 4. Сравнительный анализ возможных решений. Степень соответствия требованиям оценивается по 10-балльной шкале. Примечание: методику оценки можно где-нибудь в приложении описать

Исходя из результатов сравнительного анализа предпочтительным представляется решение **5.4** - **разработка решения с готовыми модифируемыми пользовательским интерфейсом и моделью данных с использованием готового средства управления данными.** Рекомендуется разработка концептуального прототипа сервиса на базе данного решения. Описание технической архитектуры решения, рекомендаций по технической реализации, включая рекомендации по использованию конкретных технологий и технических средств, построению модели данных будет описано в последующих разделах документа

1. **Предлагаемая техническая архитектура**

Исходя из выдвинутых бизнес-требований, функциональных требований и технических требований , результатов анализа готовых решений, анализа возможных решений предлагается следующая техническая архитектура сервиса информации (рисунок 1):

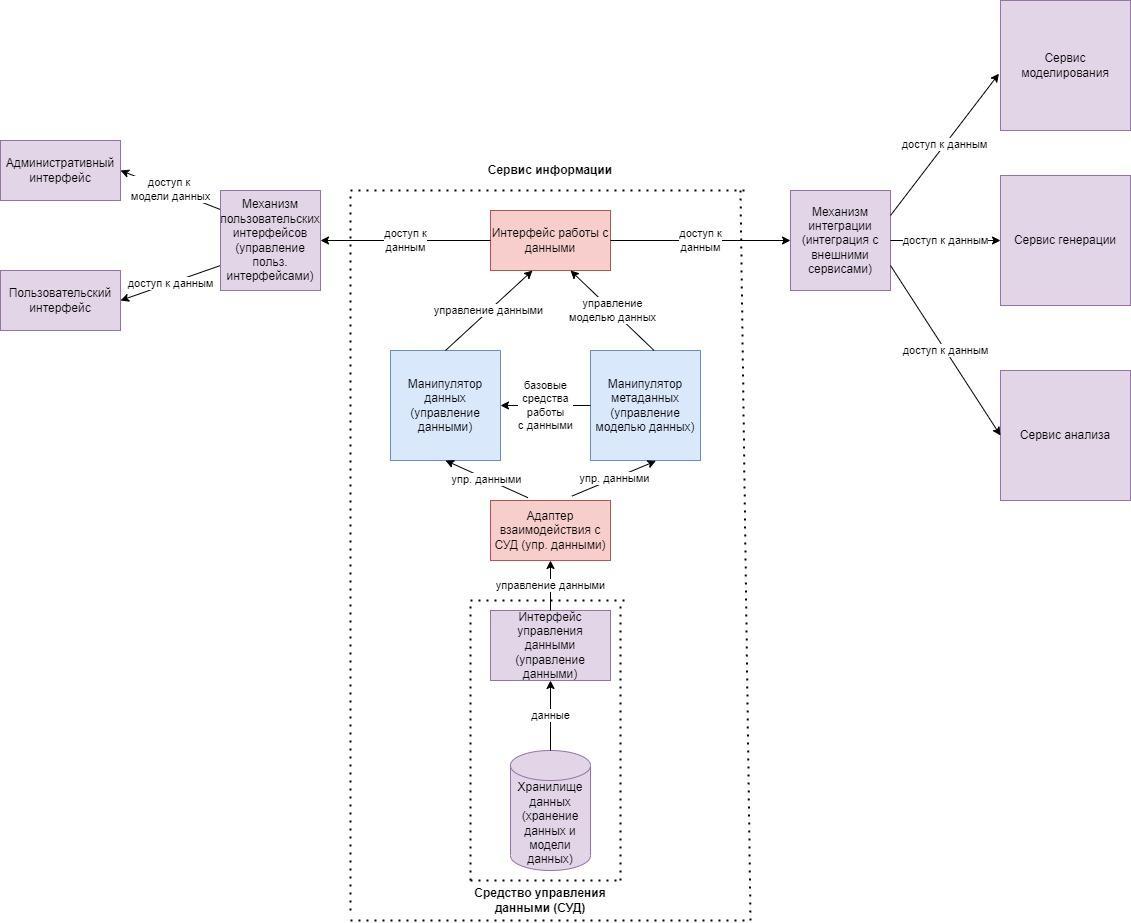


Рисунок 1. Техническая архитектура сервиса информации

Сервис информации предлагается реализовать как совокупность следующих механизмов:

**6.1.1 Хранилище данных.** Механизм хранения информации. Является частью средства управления данными. Используется для упорядоченного хранения формализованной информации. Взаимодействует с:

* Интерфейс управления данными (предоставление данных)

**6.1.2 Интерфейс управления данными.** Совокупность методов и средств непосредственного управления хранимыми данными независимо от модели данных. Является частью средства управления данными. Используется для чтения, записи, структурирования, обслуживания, предоставления доступа к данным. Взаимодействует с:

* Хранилище данных (управление хранимыми данными)
* Адаптер взаимодействия с СУД (предоставление средств управления данными)

**6.1.3 Адаптер взаимодействия с СУД.** Совокупность методов и средств взаимодействия со средствами управления данными. Используется для предоставления механизмам сервиса единого средства взаимодействия с различными видами СУД в целях возможности использования различных СУД в разных экземплярах сервиса, возможности замены СУД в процессе эксплуатации. (Примечание: включает сам адаптер + либы для работы с конкретными СУД). Взаимодействует с:

* Интерфейс управления данными (использование средств управления данными)
* Манипулятор метаданных (предоставление средств управления данными)
* Манипулятор данных (предоставление средств управления данными)

**6.1.4 Манипулятор метаданных.** Механизм управления моделью данных. Используется для создания и модификации модели данных, предоставления базовых средств управления данными. Взаимодействует с:

* Адаптер взаимодействия с СУД (использование средств управления данными)
* Интерфейс работы с данными (предоставление средств управления моделью данных)

**6.1.5 Манипулятор данных.** Механизм управления данными в рамках модели данных. Используется для обработки (чтение, запись, структурирование, поиск, доступ) к данным. (Примечание: манипуляторы описать подробнее, возможно, уточнить схему по результатам прототипирования. Да и всю архитектуру уточнить). Взаимодействует с:

* Адаптер взаимодействия с СУД (использование средств управления данными)
* Манипулятор метаданных (использование базовых средств работы с данными)
* Интерфейс работы с данными (предоставление средств управления и работы с данными)

**6.1.6 Интерфейс работы с данными.** Совокупность методов и средств эксплуатации функциональности сервиса информации. Используется для единообразного взаимодействия с сервисом информации пользовательских интерфейсов, прочих сервисов инструмента, сторонних сервисов и приложений. Взаимодействует с:

* Манипулятор метаданных (использование средств управления метаданными)
* Манипулятор данных (использование средств управления и работы с данными)
* Механизм пользовательских интерфейсов (предоставление средств работы с функциональностью сервиса)
* Механизм интеграции (предоставление средств работы с функциональностью сервиса)

Окружение сервиса информации составляют следующие категории сервисов и технических средств:

**6.2.1 Механизм пользовательских интерфейсов.** Техническое средство построения пользовательских интерфейсов. Используется для построения и эксплуатации пользовательских интерфейсов, предоставления пользователям доступа к функциональности сервиса информации посредством пользовательских интерфейсов. Взаимодействует с:

* Интерфейс работы с данными (использование функциональности сервиса информации)
* Административный интерфейс (предоставление средств использования функциональности сервиса информации в части доступа к метаданным)
* Пользовательский интерфейс (предоставление средств использования функциональности сервиса информации в части доступа к данным)

Примечание: отдельный техпроект на интерфейсы (6.2.1 - 6.2.3)

**6.2.2 Административный интерфейс.** Совокупность визуальных средств взаимодействия администраторов сервиса с моделью данных сервиса. Используется для управления моделью данных сервиса. Взаимодействует с:

* Механизм пользовательских интерфейсов (использование средств доступа к функциональности сервиса информации в части доступа к метаданным)

**6.2.3 Пользовательский интерфейс.** Совокупность визуальных средств взаимодействия пользователей сервиса с данными сервиса. Используется для доступа к данным сервиса. Взаимодействует с:

* Механизм пользовательских интерфейсов (использование средств доступа к функциональности сервиса информации в части доступа к данным)

**6.2.4 Механизм интеграции.** Совокупность средств интеграции составных частей (сервисов) инструмента проектирования и архитектурного анализа, интеграции инструмента проектирования и архитектурного анализа с внешними сервисами и приложениями. Используется для интеграции с внутренними и внешними сервисами и приложениями. Взаимодействует с:

* Интерфейс работы с данными (использование функциональности сервиса информации)

**6.2.5 Сервис моделирования, 6.2.6 Сервис генерации, 6.2.7 Сервис анализа.** Подробнее см. документ “Функциональная архитектура инструмента проектирования и архитектурного анализа”. Взаимодействуют с:

* Механизм интеграции (использование функциональности сервиса информации)

Примечание: если инструмент монолитный (так и будет в прототипе) - в качестве механизма выступает среда разработки, если микросервисный - требуется отдельный техпроект.

1. **Рекомендации по технической реализации**

Исходя из выдвинутых бизнес-требований, функциональных требований и технических требований , результатов анализа готовых решений, анализа возможных решений, предлагаемой технической архитектуры рекомендуется реализовывать механизмы и модель данных сервиса следующим образом:

**7.1 Рекомендации по используемым технологиям и техническим средствам.**

**7.1.1 Средство управления данными (механизмы 6.1.1, 6.1.2).** В качестве средства управления данными рекомендуется использование реляционных СУБДпо причине возможности структурирования, формализации, масштабирования хранимых данных, универсальности работы с данными, наличия в составе готовых реляционных СУБД полного спектра необходимых средств управления данными, работы с данными. В рамках создания концептуального прототипа рекомендуется использование СУБД PostgreSQL как распространенного бесплатного средства управления данными, обладающего всей необходимой функциональностью

**7.1.2 Адаптер взаимодействия с СУД (механизм 6.1.3).** Рекомендуется реализовывать с использованием языка Python по причине его универсальности и доступности (в т.ч. согласно требованию 3.1.6 - кадровые причины). Рекомендуется реализовывать как совокупность библиотек для непосредственной работы с поддерживаемыми сервисом СУД и единого интерфейса интеграции с данными библиотеками. В процессе и по итогам прототипирования разработать, описать и задокументировать архитектуру и спецификацию (состав методов) интерфейса.

**7.1.3 Манипуляторы метаданными и данными (механизмы 6.1.4, 6.1.5).** Рекомендуется реализовывать с использованием языка языка Python по причине его универсальности и доступности (в т.ч. согласно требованию 3.1.6 - кадровые причины). В процессе и по итогам прототипирования разработать, описать и задокументировать архитектуру и характер взаимодействия данных механизмов, уточнить целевую архитектуру сервиса по итогам.

**7.1.4 Интерфейс работы с данными.** Рекомендуется реализовывать с использованием языка Python по причине его универсальности и доступности (в т.ч. согласно требованию 3.1.6 - кадровые причины). В процессе и по итогам прототипирования разработать, описать и задокументировать архитектуру и спецификацию (состав методов) интерфейса.

**7.1.5 Механизм пользовательских интерфейсов.** В качестве механизма пользовательских интерфейсов в рамках прототипирования рекомендуется использовать любое подходящее средство разработки веб-интерфейсов на усмотрение разработчика. Окончательное решение по использованию того или иного механизма, вариативности использования механизмов - по итогам прототипирования

**7.1.7 Механизм интеграции.** В рамках прототипирования рекомендуется создание инструмента проектирования и архитектурного анализа как монолитного сервиса, в связи с чем необходимость в разработке выделенного механизма интеграции отсутствует - его роль играет среда разработки. В случае принятия решения о микросервисной реализации инструмента проектирования и архитектурного анализа по итогам прототипирования и тестовой эксплуатации потребуется отдельное проектирование механизма интеграции

**7.2 Модель данных.** Предлагается использовать следующую модель данных сервиса информации в качестве опорной при прототипировании сервиса (рисунок 2):

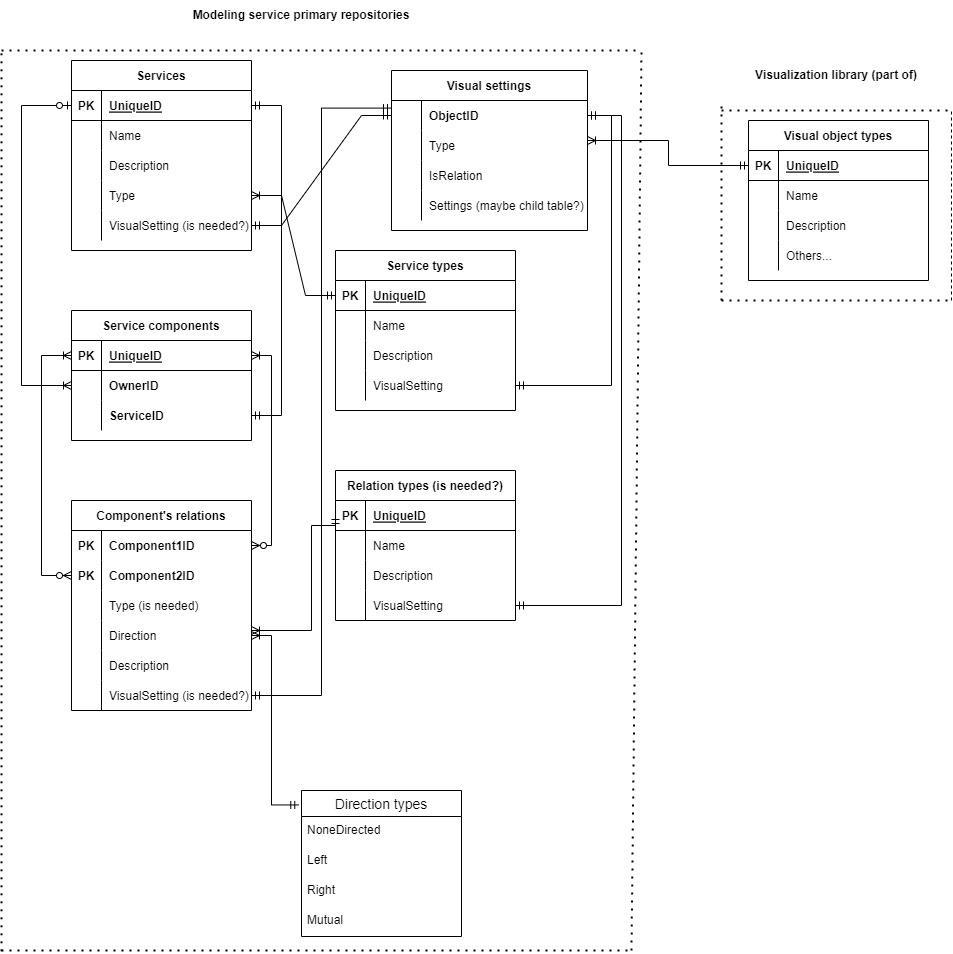


Рисунок 2. Опорная модель данных сервиса информации. Реляционная ER-диаграмма

Окончательное решение о выборе конкретной модели данных в процессе прототипирования - на усмотрение разработчика. Окончательное решение о первичной структуре модели данных рекомендуется принимать по итогам прототипирования. Первичную модель данных необходимо описать и задокументировать отдельно.

Примечание: в презентабельной версии документа добавить описание состава модели данных согласно диаграмме

1. **Заключение**

Исходя из описанного в документе можно сделать следующие выводы:

* В целях успешной эксплуатации и (возможно) тиражирования сервис информации должен соответствовать ряду функциональных, технических и бизнес-требований (см. раздел 3)
* В ходе исследований выявлен ряд существующих инструментов и технических средств, частично соответствующих выдвинутым требованиям. Ни один из исследованных инструментов не соответствует выдвинутым требованиям в достаточной мере (см. раздел 4)
* Возможно несколько вариантов создания инструмента, в достаточной мере соответствующего предъявляемым требованиям, включая как полное или частичное использование существующих инструментов, так и полную или частичную разработку собственного решения с жесткими либо модифицируемым пользовательским интерфейсом и моделью данных. Целесообразной признана разработка собственного решения с использованием готовых инструментов построения интерфейса и модели данных (см. раздел 5)
* В состав сервиса должны входить следующие механизмы - средство управления данными (вариативно), адаптер взаимодействия с СУД, манипуляторы метаданных и данных, интерфейс взаимодействия с данными (см. раздел 6)
* В целях реализации сервиса рекомендуется использование определенных технологий и технических средств. Окончательное решение по их использованию необходимо принять и задокументировать по итогам прототипирования (см. раздел 7.1)
* В качестве основы для построения модели данных сервиса рекомендуется определенная реляционная модель. Окончательное решение по структуре и составу модели данных необходимо принять по итогам прототипирования (см. раздел 7.2)