**Постановка задачи:**

1. Разработка коннектора для NoSQL СУБД Tarantool для .net core
2. Реализация протокола IProto для обеспечения взаимодействия между Tarantool и.net core

**Спецификация основных проектных требований**

**Функциональные требования**

1. Должен быть реализован бинарный протокол iproto
2. Коннектор должен предоставлять методы для взаимодействия со спэйсами и индексами
3. Коннектор должен предоставлять следующие методы при работе с данными (insert, replace, upsert, delete, update, select)
4. Коннектор должен предоставлять возможность для запуска процедур, хранящихся в Tarantool

**Нефункциональные требования**

1. Время обработки запроса не должно превышать 500мс при условии нахождения приложения, использующего коннектор и сервера Tarantool на одном сервере при малой загруженности сервера
2. Метода коннектора для получения/отправки данных должны быть доступны, как в синхронном варианте, так и в асинхронном

**Глоссарий понятий предметной области**

1. Коннектор к СУБД: программный объект, который предоставляет набор абстракций, обеспечивающий доступ ЯП к api СУБД
2. Спэйс: ‘контейнер’ используемый Tarantool’ом для хранения данных
3. Кортеж: запись, хранящая данные в Tarantool'е
4. Индекс: совокупность значений ключей и указателей

**Модель предметной области**

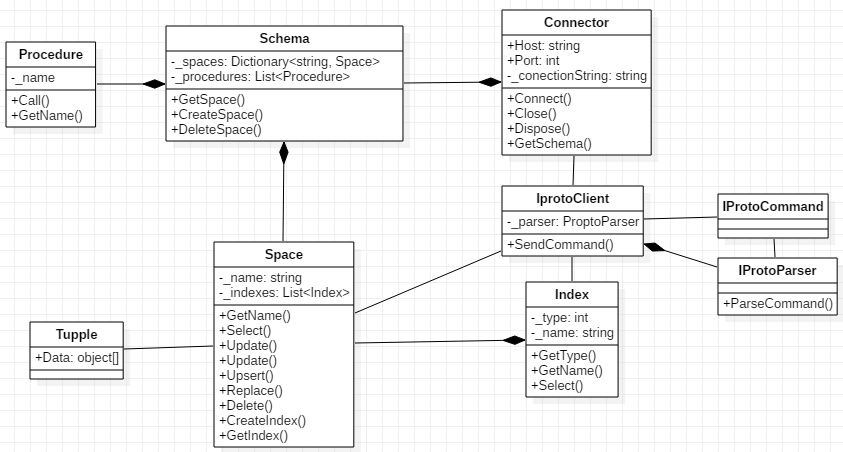


Рисунок 1.Диаграмма классов предметной области

**Выявленные актеры:**

1. пользователькое приложение
2. субд tarantool
3. коннектор

**Прецеденты:**

1. подключение к базе данных (подключение к субд для получение доступа к отправке запросов)
2. отправка запроса upsert (update с insert)
3. отправка запроса select (получение кортежей)
4. отправка запроса insert (добавление кортежа)
5. отправка запроса delete (удвление кортежа)
6. отправка запроса update (изменение кортежа)
7. отправка запроса replace (замена кортежа)
8. вызов процедуры (вызов процедуры, хранимой на стороне субд)
9. получение схемы (получение списка существующих спэйсов)
10. получение спэйса (получение спэйса для отправки запросов к конкретносу спэйсу)
11. получение индекса (получение индекса для отправки запросов к конкретному индексу)
12. создание индекса (создание нового индекса)
13. создание спэйса (создание нового спэйса)
14. сформировать запрос к бд по протоколу iproto (преобразование комманд приложения к конектору в вид воспринимаемый сервером субд)
15. обработать ответ от субд (преобразовать полученный ответ от сервера в вид, который может воспринять программа)
16. обработать запрос от коннектора

|  |  |
| --- | --- |
| Прецедент | Приоритет |
| 1 | Высокий |
| 2 | Нормальный |
| 3 | Нормальный |
| 4 | Нормальный |
| 5 | Нормальный |
| 6 | Нормальный |
| 7 | Нормальный |
| 8 | Низкий |
| 9 | Низкий |
| 10 | Высокий |
| 11 | Высокий |
| 12 | Низкий |
| 13 | Низкий |
| 14 | Высокий |
| 15 | Высокий |
| 16 | Высокий |

Таблица 1. Приоритеты прецедентов

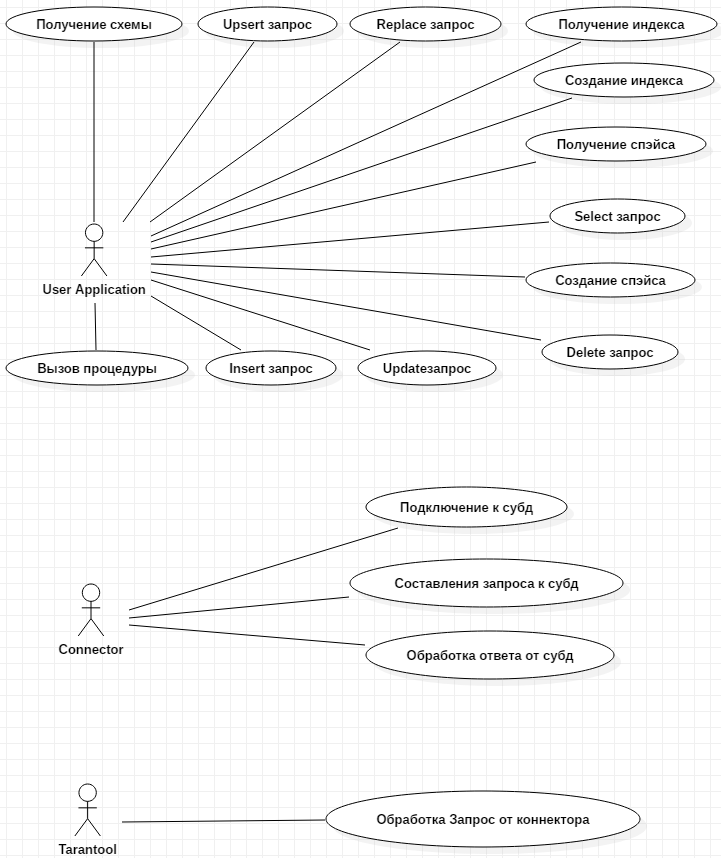


Рисунок 2. Диаграмма основных прецедентов

**Риски**

1. неоптимальный выбор структур данных (предварительное планирование поможет избежать данного риска)
2. неоптимальный выбор архитектуры (предварительно планирование поможет избежать данного риска)
3. срыв сроков (предварительная оценка сроков поможет избежать данного риска)
4. ошибки в логике работы программы (написание тестов для проверки корректности работы)
5. риск появления новых требований (применение гибкой архитектуры поможет избежать данного риска)
6. риск использование нестабильных технологий (проедение иследование по выбору технологий поможет избежать данного риска)

**Оценка по модели COCOMO-II**

0