

ИНСТРУМЕНТЫ И ОСОБЕННОСТИ ВИЗУАЛИЗАЦИИ БАЗ ДАННЫХ ДЛЯ ВЕБ-ПРИЛОЖЕНИЯ

TOOLS AND FEATURES OF DATABASE VISUALIZATION FOR A WEB APPLICATION

И.Р. Лукьянович¹⁾, В.С. Мирончук²⁾, В.В. Рабец³⁾

I. Lukyanovich¹⁾, U. Miranchuk²⁾, U. Rabets³⁾

^{1),2),3)}Белорусский государственный университет

Минск, Беларусь

^{1),2),3)}Belarusian State University

Minsk, Belarus

e-mail: ¹⁾*lukianinna12345@gmail.com*, ²⁾*vlad.mironchuk.02@gmail.com*,

³⁾*vladrabets46@gmail.com*

Рассмотрены программные средства и особенности реализации и визуального представления баз данных реляционного и объектно-ориентированного типа на примере проектов, предусматривающих их совместное использование.

The software and features of the implementation and visual representation of relational and object-oriented databases are considered on the example of sharing projects are considered.

Ключевые слова: Представление данных; СУБД; MongoDB; PostgreSQL; формат JSON.

Keywords: Data representation; DBMS; MongoDB; PostgreSQL; JSON format.

В качестве баз данных для реализации проекта были выбраны системы управления базами данных (СУБД) MongoDB и PostgreSQL. Эти СУБД применяются в веб-разработке, в частности, в рамках JavaScript-ориентированного стека MEAN/MERN.

MongoDB – не реляционная, документо-ориентированная система управления базами данных, написанная на языке C++ – является примером NoSQL-систем [1]. Модель организации данных можно описать как дерево, узлами которого являются коллекции, содержащие документы.

```
_id: ObjectId("61af212c76ab25307069a905")
displayName: "Overwatch"
totalRating: 8
price: 23.99
categories: Array
  0: ObjectId("6270f935b52979288a054192")
  1: ObjectId("618120e51134a9cfae9b1e73")
ratings: Array
  createdAt: 2021-12-07T08:54:04.600+00:00
  updatedAt: 2021-12-14T16:11:19.014+00:00
  __v: 0
```

Рис. 1. Пары ключ-значение в документе MongoDB

Каждый документ, с одной стороны, подобен коллекции, поскольку содержит не только строки, но и данные любого другого типа, который поддерживает СУБД. Документ представляет набор пар ключ-значение, что, с другой стороны, уподобляет его строкам в реляционных таблицах, где строки хранят информацию об отдельном элементе. Ключи представляют собой строки [2], значения же могут различаться по типу данных (рис. 1).

Для каждого документа в MongoDB определен уникальный идентификатор **id**, который выполняет ту же функцию, что и первичный ключ в реляционных СУБД. Разработчик задает **id** явным образом или его генерирует СУБД. Модель построения **id** гарантирует с высокой долей вероятности, что он будет иметь уникальное значение [3] (рис. 2).



Рис. 2. Идентификатор, сгенерированный СУБД

Связь между документами и коллекциями осуществляется через объект Ref (Reference), который ссылается на **id** другой сущности.

Данные в MongoDB хранятся как документы в формате binary JSON (BSON) [4], который позволяет быстрее выполнять их поиск и обработку. Следует отметить, что данные в JSON-формате занимают гораздо меньше места, чем в формате BSON. Этот недостаток, впрочем, вполне компенсируется скоростью обработки [3]. К недостаткам MongoDB следует также отнести более слабые, чем у реляционных СУБД показатели по обеспечению целостности, сокращению избыточности, разграничению доступа, долговечности, что в настоящее время именуется ACID (Atomicity-consistency-isolation-durability). Кроме того, транзакции с использованием MongoDB являются сложными. В MongoDB также нет возможности реализовать и хранить процедуры или функции, описывающие бизнес-логику на уровне базы данных, что реализуемо во всех реляционных СУБД.

Схема данных в MongoDB генерируется по мере добавления документов автоматически и может быть визуализирована вручную, посредством бесплатного веб-ресурса для моделирования NoSQL баз данных [4] (рис. 3), обладающего минимумом инструментов.

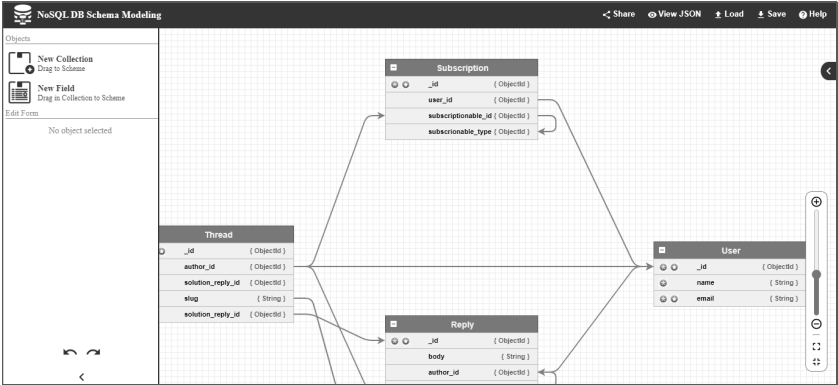


Рис. 3. Моделирование схемы NoSQL базы данных [4]

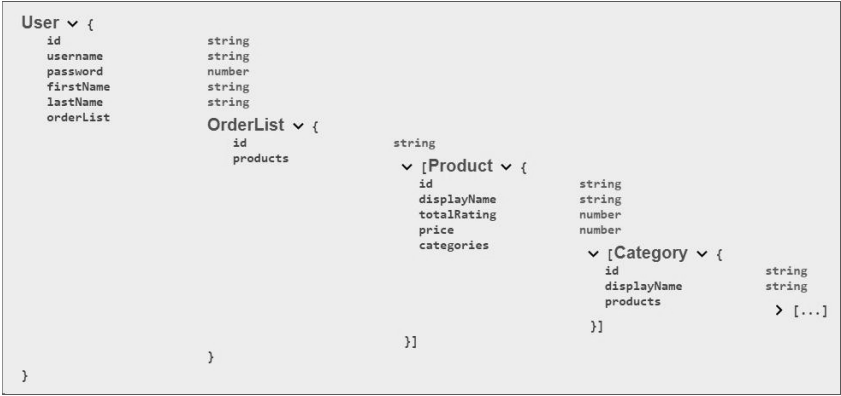


Рис. 4. Схема данных серверной части веб-приложения, сгенерированная Swagger UI

Современным инструментом визуализации NoSQL базы данных является Swagger-UI, который представляет собой фреймворк, позволяющий генерировать документацию на основании программного кода [5]. На основе кода или набора правил Swagger автоматически генерирует документацию в формате JSON-файла. Ее можно встроить на страницу сайта или в приложение, чтобы пользователи или заказчик мог знакомиться с документацией, в том числе в интерактивной форме. Swagger

генерирует на основе файла с расширением YAML (рис. 3) схему данных, создавая графическую документацию backend'а веб-приложения (рис. 4). Swagger включает как бесплатные, так и платные компоненты.

PostgreSQL – реляционная СУБД с открытым исходным кодом. Существует в реализациях для множества UNIX-подобных платформ, включая AIX, различные BSD-системы, HP-UX, IRIX, Linux, macOS, Solaris/OpenSolaris, Tru64, QNX, а также для Microsoft Windows. На рис. 5 представлена реляционная база данных проекта.

The screenshot shows the PostgreSQL Enterprise console with a table named 'products'. The table has columns: id, name, category, status, location, and product. The 'product' column is highlighted with a red circle. The table contains 10 rows of data, including products like 'RURAL MEATS', 'RURAL MEATS', 'RURAL MEATS', 'RURAL MEATS', 'RURAL MEATS', 'RURAL MEATS', 'RURAL MEATS', 'RURAL MEATS', 'RURAL MEATS', and 'RURAL MEATS'.

id	name	category	status	location	product
1	RURAL MEATS	MEATS	AVAILABLE	STALLEN RD	STALLEN RD
2	RURAL MEATS	MEATS	AVAILABLE	STALLEN RD	STALLEN RD
3	RURAL MEATS	MEATS	AVAILABLE	STALLEN RD	STALLEN RD
4	RURAL MEATS	MEATS	AVAILABLE	STALLEN RD	STALLEN RD
5	RURAL MEATS	MEATS	AVAILABLE	STALLEN RD	STALLEN RD
6	RURAL MEATS	MEATS	AVAILABLE	STALLEN RD	STALLEN RD
7	RURAL MEATS	MEATS	AVAILABLE	STALLEN RD	STALLEN RD
8	RURAL MEATS	MEATS	AVAILABLE	STALLEN RD	STALLEN RD
9	RURAL MEATS	MEATS	AVAILABLE	STALLEN RD	STALLEN RD
10	RURAL MEATS	MEATS	AVAILABLE	STALLEN RD	STALLEN RD

Рис. 5. Реляционная база данных проекта, реализованная на PostgreSQL

Рассмотрим связи данных в PostgreSQL и их визуализацию. Всего существуют 4 типа связей: OneToOne, OneToMany, ManyToOne, ManyToMany.

OneToOne. Связь используется для отношения сущностей по типу “Один к одному”, например связь User – OrderList. Заказ относится только к одному пользователю.

OneToMany. Связь использует когда какое-то поле сущности относится ко многим сущностям, например OrderList – Product – в списке заказов может быть несколько продуктов. Вследствие создается таблица отношения. ManyToOne реализуется связь в обратном направлении: Product – OrderList.

ManyToMany. Используется в том случае, когда какое-то поле сущности относится ко многим другим сущностям, и в тоже время другие сущности относятся ко многим исходным сущностям. В этом случае создается 2 таблицы из всех сочетаний отношений. Например, связь Product – Category. У продукта может быть много категорий, в то же время категория относится ко многим продуктам (рис. 6, 7).

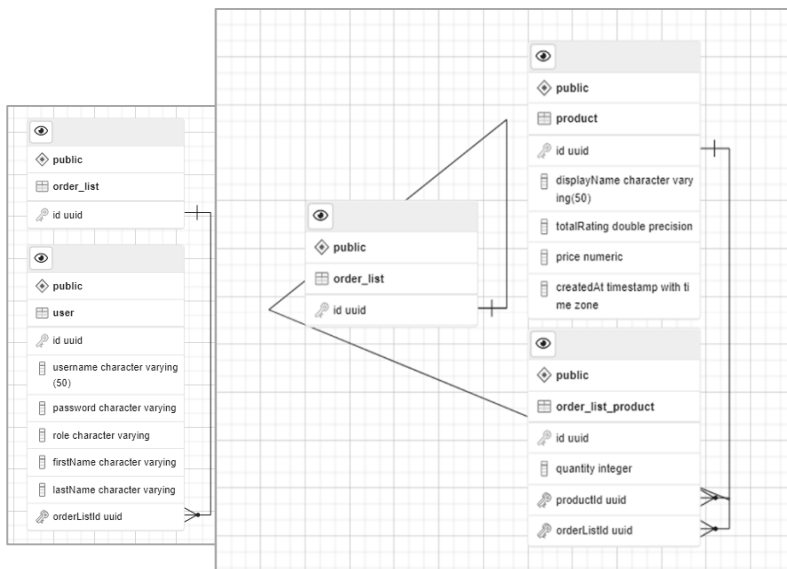


Рис. 6. Связи «Один-к-одному», «Один-ко-многим», «Многие-ко-многим» на диаграмме сущность-связь

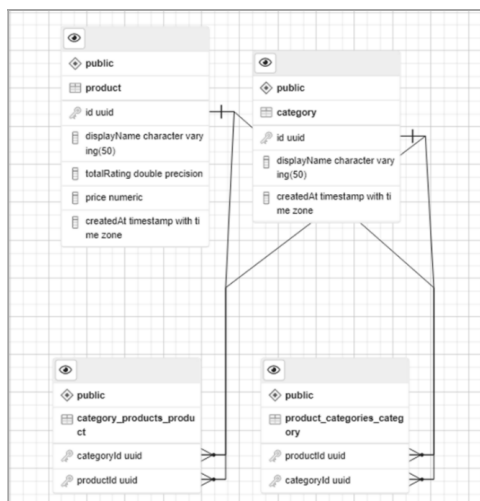


Рис. 7. Схема данных проекта

Реляционные базы данных имеют неоспоримые преимущества, если хранимые данные табличного типа, если безопасность данных является приоритетом и предъявляются высокие требования по ACID.

Если данные плохо структурированы, и их объем потенциально существенно увеличится, следует использовать NoSQL базы данных, например, документо-ориентированную СУБД MongoDB, для которых появляются развитые средства визуализации, что повышает эффективность и качество разработки.

БИБЛИОГРАФИЧЕСКИЕ ССЫЛКИ

1. MongoDB [Электронный ресурс]. URL: <http://softtime.info/view/MongoDB> (дата доступа: 15.02.2022).
2. Введение в JSON [Электронный ресурс]. URL: <http://json.org/json-ru.html> (дата доступа: 17.02.2022).
3. Королева Ю.А., Маслова В.О., Козлов В.К. Разработка концепции миграции данных между реляционными и нереляционными системами БД [Электронный ресурс]. URL: <https://cyberleninka.ru/article/n/razrabotka-kontseptsii-migratsii-dannyh-mezhdu-relyatsionnymi-i-nerelyatsionnymi-sistemami-bd/viewer> (дата доступа: 05.11.2022).
4. NoSQL Database Planning & Modeling Tool Online [NoSQL DB Modeler, Entity Relationship Diagram [Электронный ресурс]. URL: <https://nosqldbm.ru> (дата доступа: 05.11.2022).
5. Руководство Swagger UI [Электронный ресурс]. URL: <https://starkovden.github.io/swagger-ui-tutorial.html> (дата доступа: 05.11.2022).

МАТЕМАТИЧЕСКИЕ АСПЕКТЫ СОВРЕМЕННЫХ РЕЙТИНГОВЫХ СИСТЕМ

MATHEMATICAL ASPECTS OF MODERN RATING SYSTEMS

В.А. Ницагин

V.A. Nifagin

Белорусский государственный университет

Минск, Беларусь

Belarusian State University

Minsk, Belarus

e-mail: vladnifagin@bsu.by

Разнообразные рейтинговые системы и интернет поисковики используют математические алгоритмы, включающие в качестве содержательного основания участника рейтинговой системы, т. е. объект, подлежащий оценке в рассматриваемой рейтинговой системе, оцифрованный параметр – ранг в виде неотрицательного числа, характеризующий один из аспектов (свойств, показателей полезности) участника рейтинговой системы, в заданном числовом диапазоне или их совокупность. А также целевую вектор-функцию (или функцию полезности, функцию предпочтения) – функционал F , в соответствии, со значением которого