

Шаг 4: гибкость и адаптация схемы

Немного теории

Гибкость схемы данных является ключевым аспектом при работе с NoSQL базами данных, включая MongoDB. Это означает возможность адаптации структуры документов к изменяющимся требованиям и потребностям бизнеса.

1. Динамические схемы: в отличие от реляционных баз данных, NoSQL не требует жестко заданной схемы, что позволяет легко добавлять, удалять или изменять поля в документах.
2. Интеграция новых типов данных: гибкость схемы упрощает процесс добавления новых типов данных, что особенно важно в динамично меняющихся условиях рынка и при развитии бизнеса.
3. Проактивная адаптация: гибкость схемы позволяет предвидеть и реагировать на будущие изменения в требованиях к данным, что повышает устойчивость и масштабируемость системы.

В нашем кейсе можно выделить такие нюансы:

1. Расширяемость данных клиентов и бронирований:
 - возможность добавления новых полей в документы

Клиенты

, таких как предпочтения клиентов или история общения, без необходимости изменения всей структуры базы данных.
2. Адаптация данных номеров отеля:

- возможность внесения изменений в документы `Номера` , например, добавление новых удобств или услуг, а также изменение ценовой политики.

3. Управление обновлениями и расширениями:

- способность быстро реагировать на изменения в бизнес-требованиях, например, внедрение новых видов бронирования или специальных предложений.
- Пример (возможный новый сценарий): гостиница решает добавить программу лояльности. Для этого в документы `Клиенты` можно добавить поле `уровень лояльности` и `накопленные баллы` , не затрагивая остальную структуру базы данных.

Гибкость и адаптивность схемы данных в MongoDB позволяет гостиничному сервису оставаться отзывчивым к изменениям бизнес-требований и бизнес-процессов. Это обеспечивает возможность непрерывного развития продукта, поддерживая при этом эффективное и стабильное функционирование системы управления данными.

Что дальше?

У нас в итоге получилась такая схема данных для MongoDB.

Коллекция: `Клиенты`

Эта коллекция содержит документы, представляющие клиентов гостиницы. Каждый документ включает личные данные клиента и информацию о его бронированиях.

Пример документа в `Клиенты` :

```
{
  "ID клиента": "C1001",
  "имя": "Иван Иванов",
  "номер телефона": "1234567890",
  "email": "ivanov@example.com",
  "адрес": "г. Москва, ул. Пушкина, д. 1",
  "бронирования": [
    {
      "ID брони": "B2001",
      "дата заезда": "2023-01-01",
      "дата выезда": "2023-01-05",
      "статус бронирования": "подтверждено",
      "ID номера": "N3001",
      "статус номера": "доступен"
    },
    {
      "ID брони": "B2002",
      "дата заезда": "2023-02-15",
      "дата выезда": "2023-02-20",
      "статус бронирования": "ожидается",
      "ID номера": "N3002",
      "статус номера": "доступен"
    }
  ]
}
```

Коллекция:

Эта коллекция содержит документы, каждый из которых представляет номер в гостинице, включая его тип, стоимость и статус доступности.

Пример документа в :

```
{
  "ID номера": "N3001",
  "ID гостиницы": "H1001",
  "тип номера": "люкс",
```

```
"стоимость": 15000,  
"статус доступности": "доступен"  
}
```

Связи между Коллекциями

- В коллекции `Клиенты`, поля `ID номера` в массиве `бронирования` служат ссылками на соответствующие документы в коллекции `Номера`.

В реляционных базах данных мы использовали DDL для создания физической модели данных в СУБД. Но MongoDB использует динамическую схему, что означает, что вы не обязаны заранее определять структуру (схему) данных. Вы можете создавать документы на лету, и MongoDB автоматически сохранит их в соответствующем формате BSON.

В MongoDB нет необходимости в языке DDL для создания таблиц и определения столбцов, так как это нереляционная база данных. Структура данных определяется динамически, когда вы вставляете документы.

Вместо SQL MongoDB использует свой собственный язык запросов, который является более функциональным и гибким для работы с документо-ориентированными данными. Он позволяет выполнять различные операции, такие как поиск, вставка, обновление и удаление документов.