Board Game Library and Lending System Database

A. Description: This is a database management system for lending and record-keeping of a library of board games. Users can loan board games in a similar fashion to a book library.

B. Team Members:

1. Rohith Gowda Ranganatha
2. Kavit Mehta

C. MongoDB Implementation:

1. "**user**" ➡️ "**loan**" (1:M) relationship converted into a simple array - "**user_active_loans"**.
2. "**game**" ➡️ "**loan**" (1:M) relationship converted into an array of subdocuments - "**boardgames_with_borrowers**".
3. "**user**" ➡️ "**loan**" (1:M) relationship converted into an embedded subdocument - "**users_with_loans"**.

D. Entity Relationship Diagram:

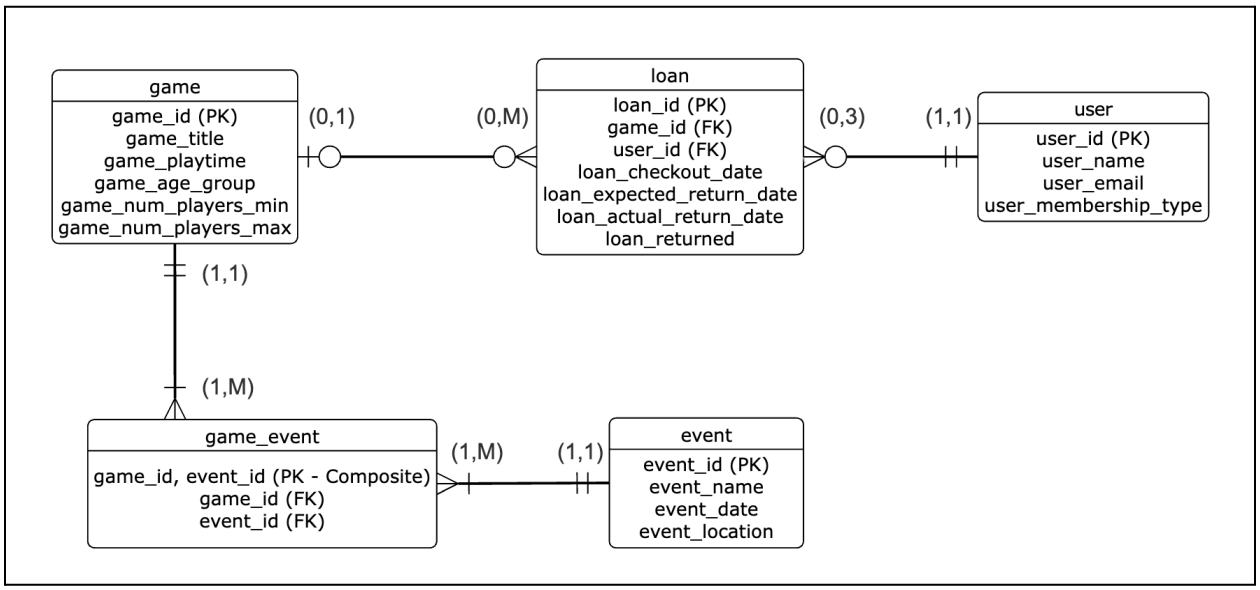

Fig. 1: Boardgames Library System ERD

E. Design 1: user_active_loans (Array)

- The document stores a user's active loan data.
- Users "David Lee" and "Grace Miller" each have two active loans, and their data is inserted into the MongoDB array "user_active_loans".
- The infinite array issue is handled by Business Rule 2: A user can't borrow more than 3 games at a time, i.e., a maximum of 3 active loans allowed.

Below is the structure breakdown:

Relationship Side	Field	Type	Rules
One	user_id	integer	Minimum value = 1
	user_name	string	Minimum length = 3
	user_email	string	
	user_membership_type	string	
Many	loan_id	integer array	Maximum length = 3

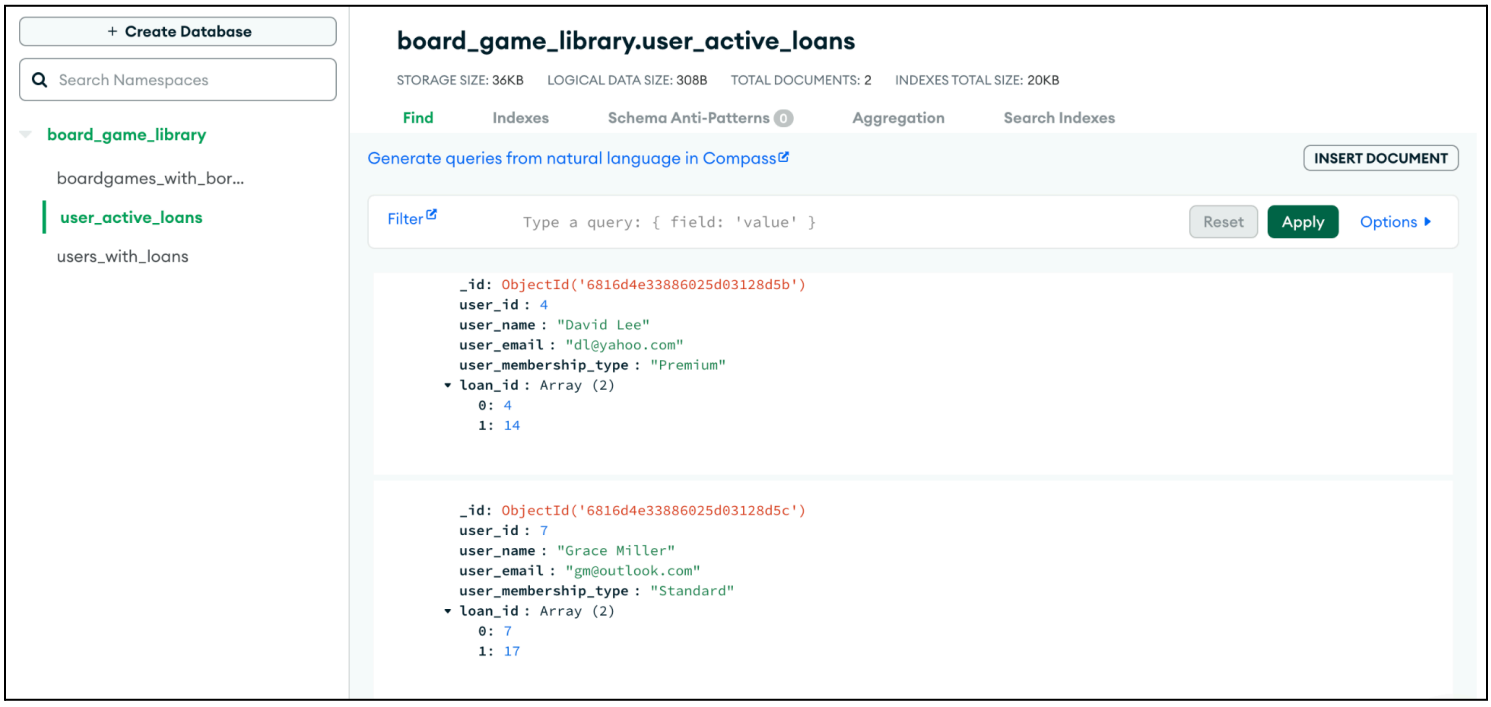

Fig. 2: user_active_loans collection

F. Design 2: users_with_loans (Embedded Subdocuments)

- The document stores a user's active loan data.
- Users "David Lee" and "Bob Smith" each have two active loans, and their data is inserted into the subdocument structure "users_with_loans".
- Infinite document issue is handled by Business Rule 2: A user can't borrow more than 3 games at a time, i.e., a maximum of 3 active loans allowed. This is also ensured by the attribute rule "loan_returned" = false.

Below is the structure breakdown:

Relationship Side	Field	Type	Rules
One	user_id	integer	Minimum value = 1
	user_name	string	Minimum length = 3
Many	load_id	integer	Minimum value = 1
	game_id	integer	Minimum value = 1
	loan_checkout_date	date	
	loan_expected_return_date	date	
	loan_returned	boolean	Value = false

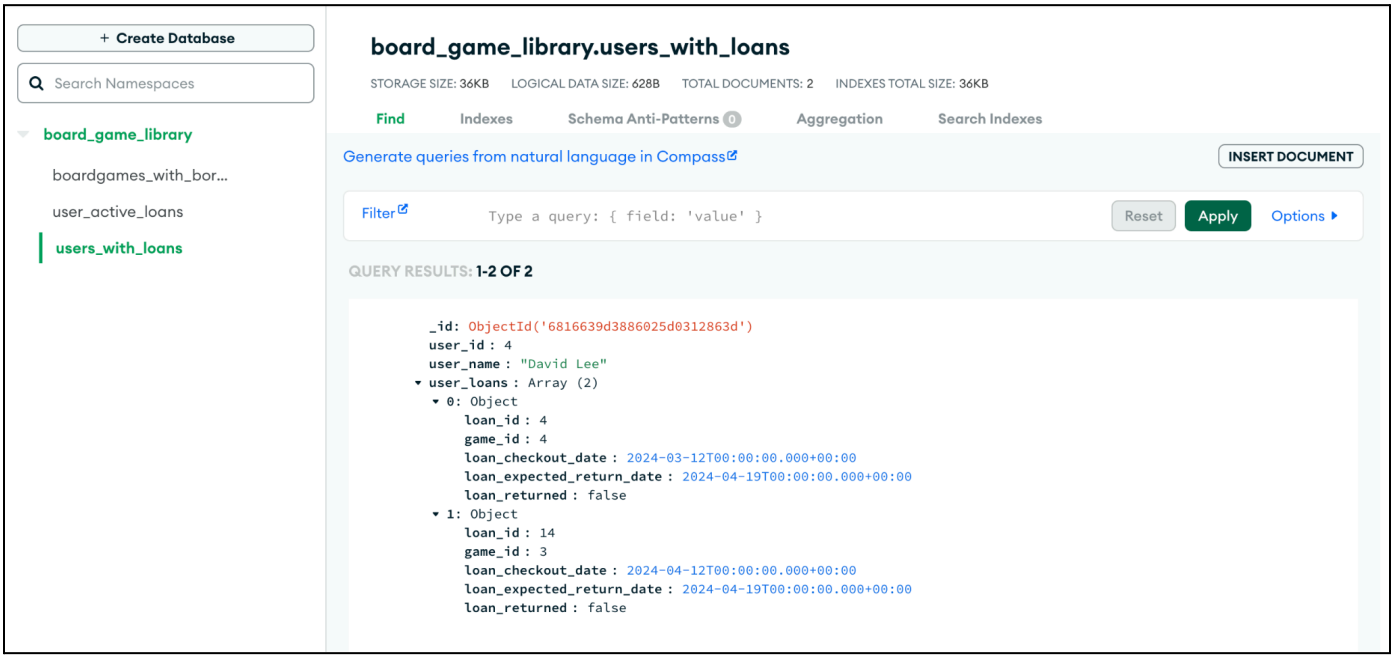

Fig. 3: users_with_loans collection

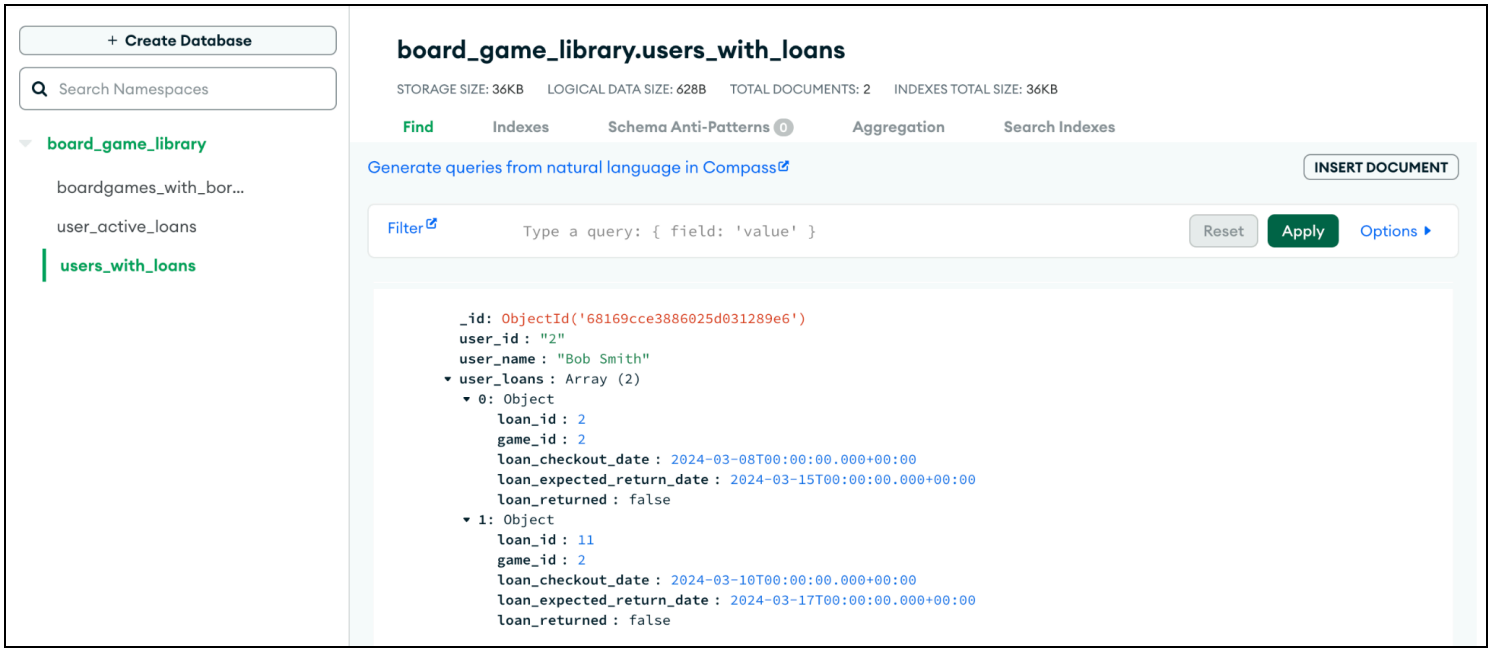

Fig. 4: users_with_loans collection

G. Design 3: boardgames_with_borrowers (Array of Subdocuments)

- The document stores information about all active loans for a particular game.
- The game “Chess” is loaned by users “David Lee” and “Eve Adams”. Similarly, “Exploding Kittens” is loaned by users “Frank White” and “Hank Green”.
- Infinite document issue is handled by only inserting games with corresponding active loans. Once a loan is returned, the record is removed. This is ensured by the attribute rule “loan_returned” = false.

Below is the structure breakdown:

Relationship Side	Field	Type	Rules
One	game_id	integer	Minimum value = 1
	game_title	string	
Many	user_id	integer	Minimum value = 1
	user_name	string	Minimum Length = 3
	loan_checkout_date	date	
	loan_returned	boolean	Value = false
	load_id	integer	Minimum value = 1

Fig. 5: boardgames_with_borrowers collection

Fig. 6: boardgames_with_borrowers collection

H. JSON Schema for Design 1 “user_active_loans”:

- **Title**: board_game_library_schema
- **Description**: This schema defines the “user_active_loans”
- **Required Fields**: user_id; user_name; loan_id
- **Additional rules**:
  - Rule 1: user_id must be positive
  - Rule 2: user_name must be at least 3 characters
  - Rule 3: loan_id items must be positive
  - Rule 4: loan_id max length = 3
  - Rule 5: no duplicate loan_id items

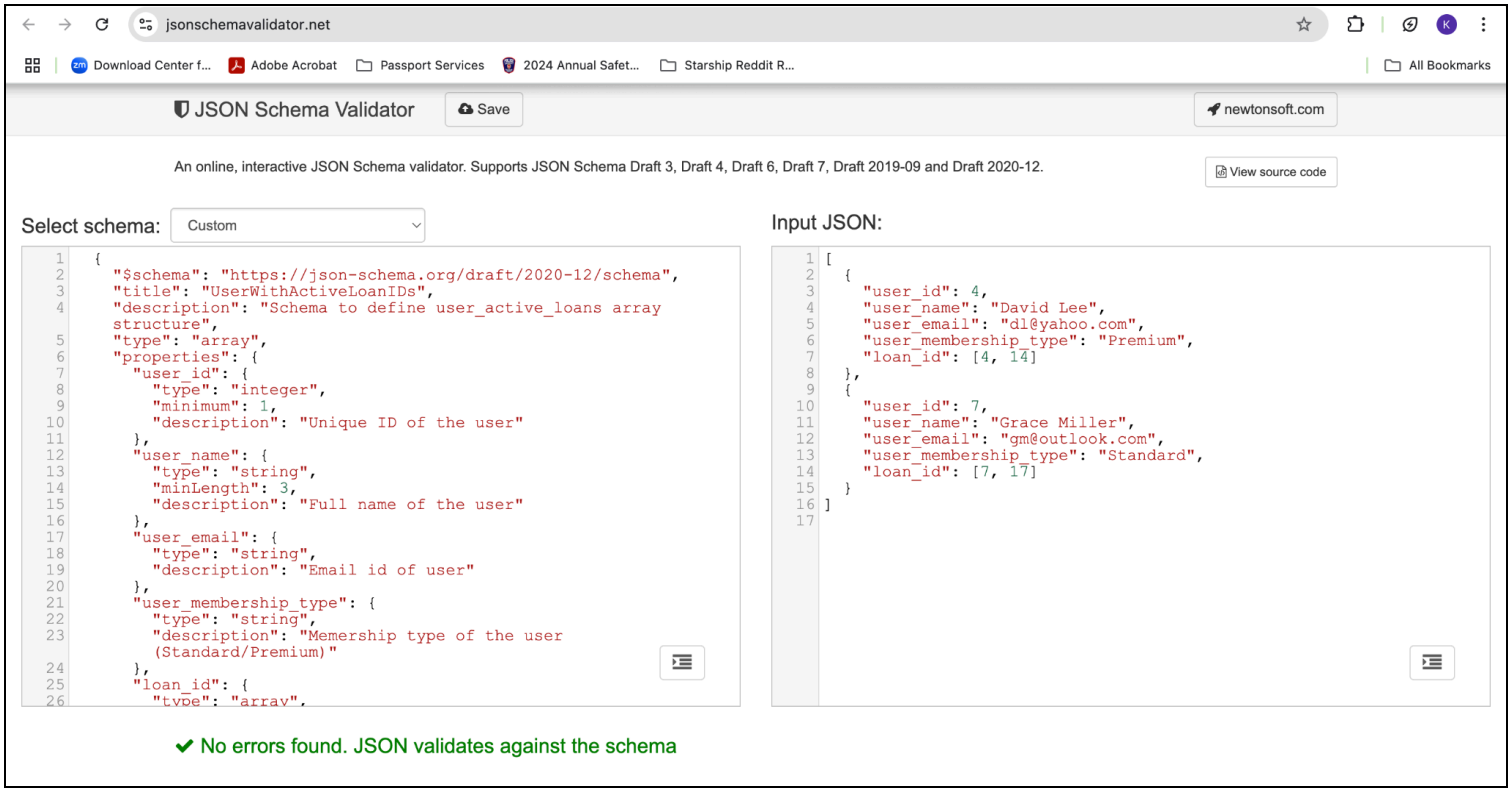

Fig. 7: user_active_loans JSON Schema Validator

I. Queries:

1. **Within the “boardgames_with_borrowers” collection, find games where any borrower has loan_id > 15.**

```
{
  "game_borrowers.loan_id": { "$gt": 15 }
}
```

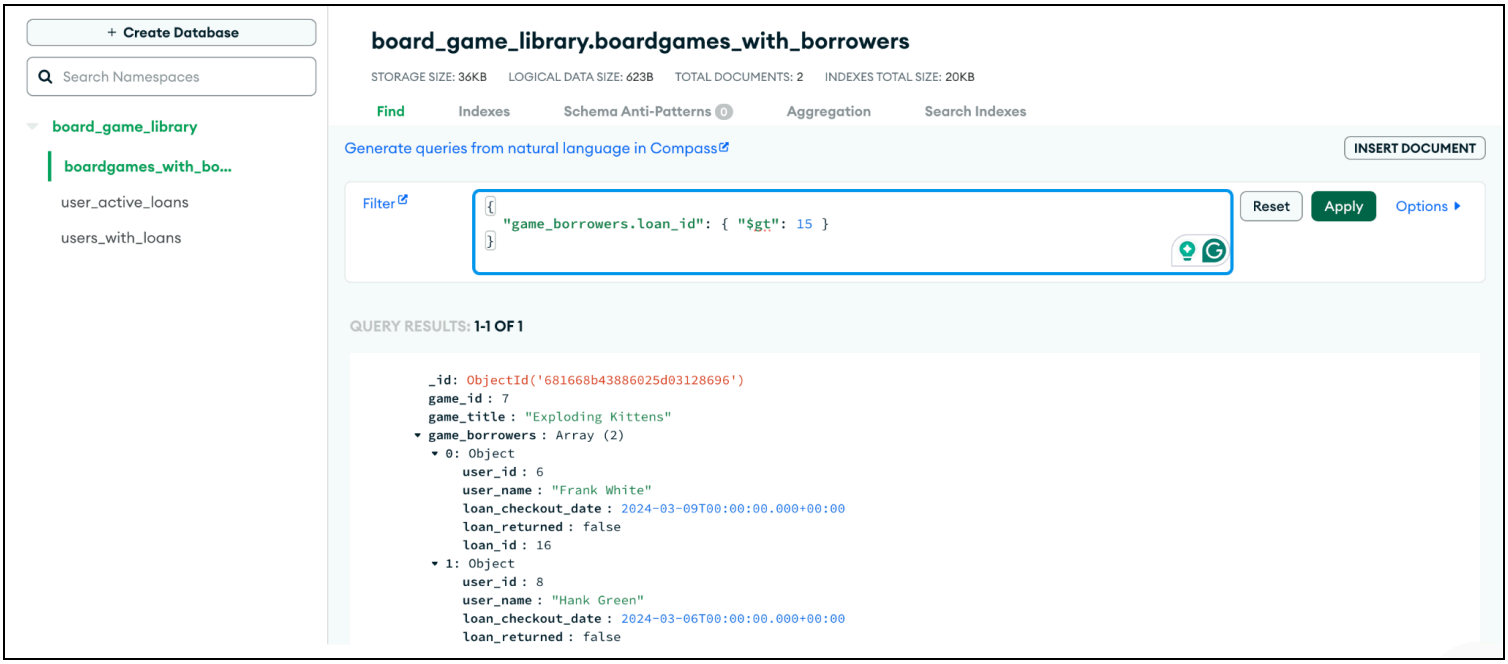

Fig. 8: "Greater than or less than" query results

2. **Within the "users_with_loans" collection, find users who have exactly 2 loan records in the user_loans array.**

   { "user_loans": { "$size": 2 } }

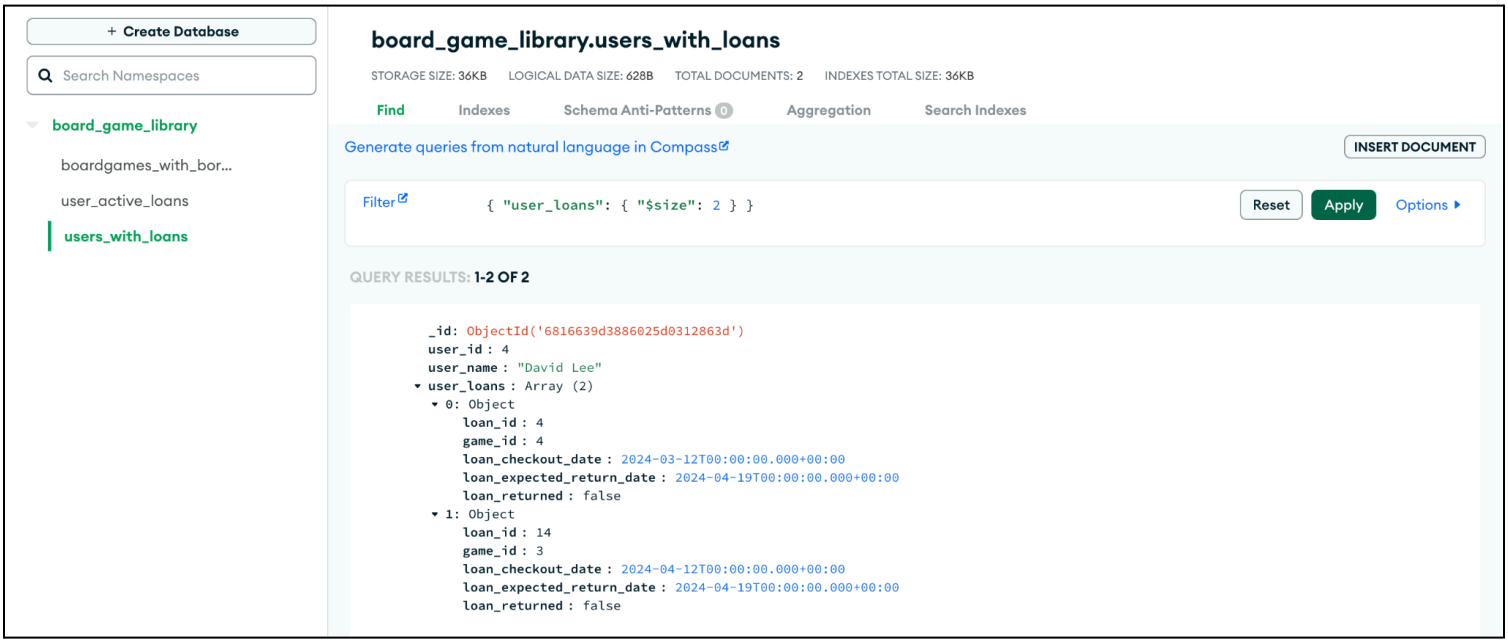

Fig. 9: "Query within an array" query results

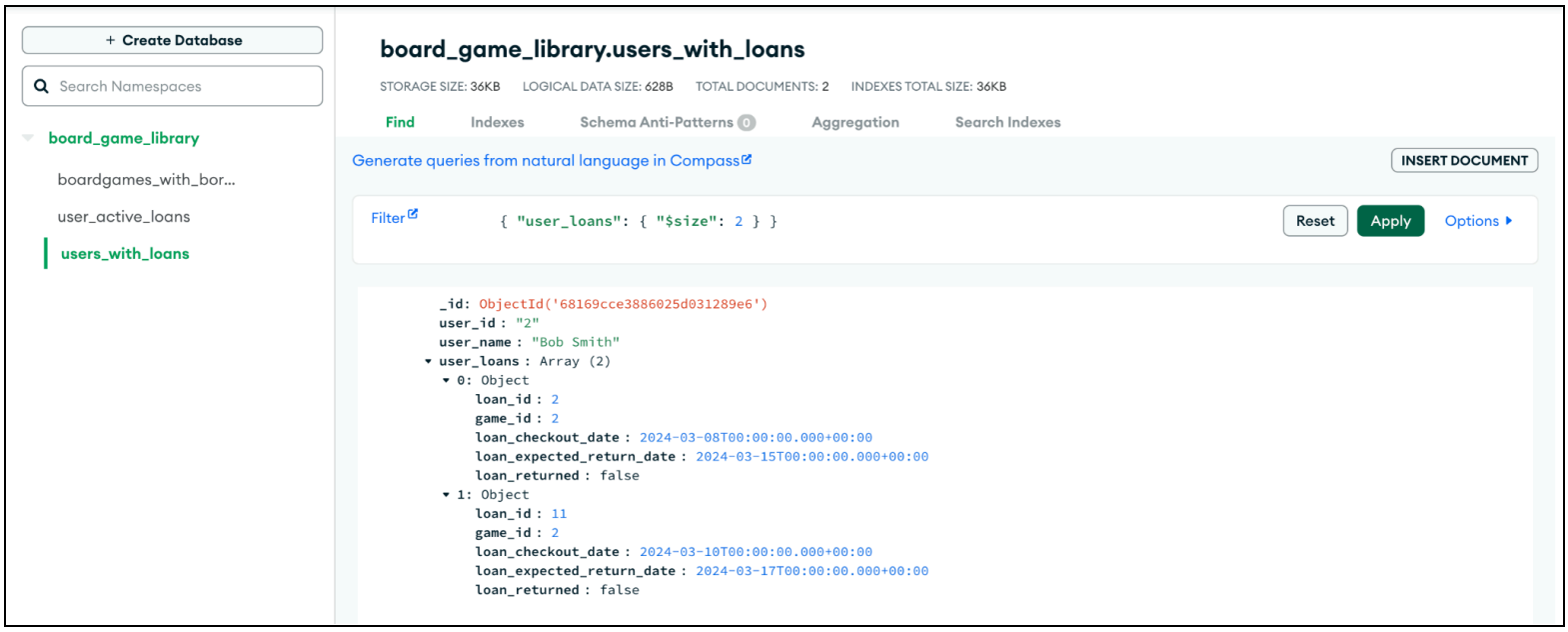

Fig. 10: "Query within an array" query results

3. **Within the "users_with_loans" collection, find users who have a loan where loan_returned is true.**

   { "user_loans.loan_returned": true }

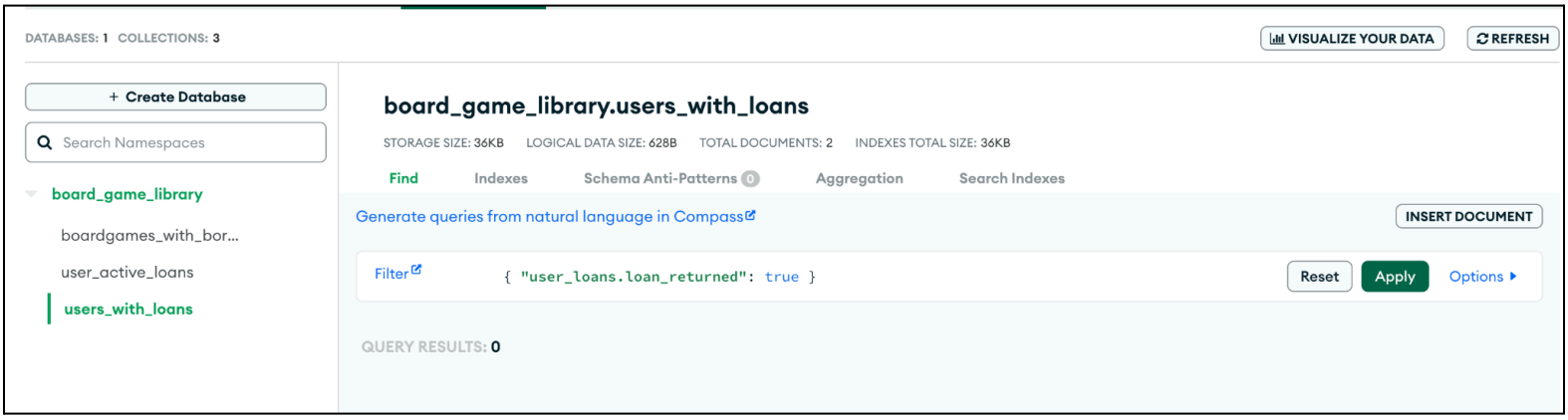

Fig. 11: "Query within a subdocument" query results

Note: All the original entities and queries are saved as tables and views respectively in Orson's user027_DB3 schema.