

Coverage for app/services/request_service.py: 81%



69 statements 56 run 13 missing 0 excluded

« prev ^ index » next coverage.py v7.11.3, created at 2025-11-21 23:38 -0800

```
1 from pathlib import Path
2 from app.models.request import Request
3 from app.schemas.request import RequestCreate, RequestRead
4 from app.repositories.csv_repository import CSVRepository
5
6 class RequestService:
7     def __init__(self):
8         self.repo = CSVRepository()
9         self.path = Path(__file__).resolve().parents[1] / "data" / "Requests.csv"
10        self.totalpath = Path(__file__).resolve().parents[1] / "data" / "Total_Requested.csv"
11        self.fields = ["RequestID", "UserID", "Book Title", "Author", "ISBN"]
12        self.total_fields = ["ISBN", "Total Requested"]
13
14
15    def __generate_next_id(self) -> int:
16        """Generate the next RequestID number."""
17        rows = self.repo.read_all(self.path)
18        if not rows:
19            return 1
20        ids = [int(r["RequestID"]) for r in rows if r["RequestID"].isdigit()]
21        return max(ids, default=0) + 1
22
23    def __already_requested(self, user_id: int, isbn: str) -> bool:
24        """Checks if this user has already requested the same book"""
25        rows = self.repo.read_all(self.path)
26        return any(r["UserID"] == str(user_id) and r["ISBN"] == isbn for r in rows)
27
28    def __decrease_count(self, isbn: str):
29        rows = self.repo.read_all(self.totalpath)
30        for r in rows:
31            if r["ISBN"] == isbn:
```

```
32         new_count = int(r["Total Requested"]) - 1
33         if new_count<=0: rows.remove(r)
34     else:
35         r["Total Requested"] = str(new_count)
36     break
37
38     self.repo.write_all(self.totalpath, self.total_fields, rows)
39
40
41     def __update_total_requested(self, isbn: str):
42         rows = self.repo.read_all(self.totalpath)
43         found = False
44
45         for r in rows:
46             if r["ISBN"] == isbn:
47                 r["Total Requested"] = str(int(r["Total Requested"]) + 1)
48                 found = True
49             break
50
51         if not found:
52             rows.append({"ISBN": isbn, "Total Requested": "1"})
53
54         self.repo.write_all(self.totalpath, self.total_fields, rows)
55
56     def get_all_requests(self) -> list[RequestRead]:
57         """Retrieve all requests from the requests.csv file."""
58         rows = self.repo.read_all(self.path)
59         return [RequestRead(**Request.from_dict(r).to_api_dict()) for r in rows]
56
60
61     def create_request(self, user_id: int, data: RequestCreate) -> RequestRead:
62         """Create a new book request and save it to the requests.csv file."""
63         if self.__already_requested(user_id, data.isbn):
64             raise ValueError("This user has already requested this book.")
65
66         new_id = self.__generate_next_id()
67         request = Request(
68             request_id = new_id,
```

```
69         user_id = user_id,
70         book_title = data.book_title,
71         author = data.author,
72         isbn = data.isbn,
73     )
74
75     self.repo.append_row(self.path, self.fields, request.to_csv_dict())
76     self._update_total_requested(data.isbn)
77
78     return RequestRead(**request.to_api_dict())
79
80     def delete_request(self, request_id: int) -> bool:
81         """Delete a specific request and reindex the remaining IDs."""
82         rows = self.repo.read_all(self.path)
83         original_count = len(rows)
84
85
86         isbn_to_decrement = None
87         for r in rows:
88             if int(r["RequestID"]) == request_id:
89                 isbn_to_decrement = r["ISBN"]
90                 break
91
92         if isbn_to_decrement is None:
93             return False
94
95         updated_rows = [r for r in rows if r["RequestID"] != str(request_id)]
96
97         if len(updated_rows) == original_count:
98             return False
99
100        for i, row in enumerate(updated_rows, start=1):
101            row["RequestID"] = str(i)
102
103        self.repo.write_all(self.path, self.fields, updated_rows)
104        self._decrease_count(isbn_to_decrement)
105
```

106

```
    return True
```

[« prev](#) [^ index](#) [» next](#) coverage.py v7.11.3, created at 2025-11-21 23:38 -0800