

app/services/request_service.py: 85%

60

11

0



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

app/services/request_service.py: 85%

```
35     for r in rows:
36         if r["ISBN"] == isbn:
37             new_count = int(r["Total Requested"]) - 1
38             if new_count<=0: rows.remove(r)
39             else:
40                 r["Total Requested"] = str(new_count)
41                 break
42
43             self.repo.write_all(path, self.total_fields, rows)
44
45
46     def __update_total_requested(self, isbn: str):
47         path = Path(__file__).resolve().parents[1] / "data" / "Total_Requested.csv"
48         rows = self.repo.read_all(path)
49         found = False
50
51         for r in rows:
52             if r["ISBN"] == isbn:
53                 r["Total Requested"] = str(int(r["Total Requested"]) + 1)
54                 found = True
55                 break
56
57         if not found:
58             rows.append({"ISBN": isbn, "Total Requested": "1"})
59
60         self.repo.write_all(path, self.total_fields, rows)
61
62     def get_all_requests(self) -> list[RequestRead]:
63         """
64             Retrieve all requests from the requests.csv file.
65         """
66         rows = self.repo.read_all(self.path)
67         return [RequestRead(**Request.from_dict(r).to_api_dict()) for r in rows]
68
69     def create_request(self, user_id: int, data: RequestCreate) -> RequestRead:
```

app/services/request_service.py: 85%

```
72     """
73     if self.__already_requested(user_id, data.isbn):
74         raise ValueError("This user has already requested this book.")
75
76     new_id = self.__generate_next_id()
77     request = Request(
78         request_id=new_id,
79         user_id=user_id,
80         book_title=data.book_title,
81         author=data.author,
82         isbn=data.isbn,
83     )
84
85     self.repo.append_row(self.path, self.fields, request.to_csv_dict())
86     self.__update_total_requested(data.isbn)
87
88     return RequestRead(**request.to_api_dict())
89
90 def delete_request(self, request_id: int) -> bool:
91     """
92     Delete a specific request and reindex the remaining IDs.
93     """
94     rows = self.repo.read_all(self.path)
95     original_count = len(rows)
96
97
98     isbn_to_decrement = None
99     for r in rows:
100         if int(r["RequestID"]) == request_id:
101             isbn_to_decrement = r["ISBN"]
102             break
103
104     # If the request ID was not found, nothing to delete
105     if isbn_to_decrement is None:
106         return False
```

app/services/request_service.py: 85%

```
109
110     if len(updated_rows) == original_count:
111         return False
112
113     for i, row in enumerate(updated_rows, start=1):
114         row["RequestID"] = str(i)
115
116     self.repo.write_all(self.path, self.fields, updated_rows)
117     self.__decrease_count(isbn_to_decrement)
118
119     return True
```

[« prev](#) [^ index](#) [» next](#) coverage.py v7.11.3, created at 2025-11-19 02:12 +0000