

Coverage for app/services/user_service.py: 95%

39 statements

37 run

2 missing

0 excluded

« prev ^ index » next coverage.py v7.11.3, created at 2025-11-20 19:17 -0800

```
1 import os, csv
2 from typing import Optional, Dict
3 from app.utils.data_manager import CSVRepository
4
5 BASE_DIR = os.path.dirname(os.path.dirname(__file__))
6 DATA_DIR = os.path.join(BASE_DIR, "data")
7 os.makedirs(DATA_DIR, exist_ok=True)
8
9 USER_CSV = os.path.join(DATA_DIR, "Users.csv")
10 FIELDNAMES = ["id", "username", "email", "password_hash"]
11
12 class CSVUserService:
13     def __init__(self, repo: CSVRepository, path: str = USER_CSV):
14         self.repo = repo
15         self.path = path
16         if not os.path.exists(self.path):
17             with open(self.path, "w", newline="", encoding="utf-8") as f:
18                 csv.DictWriter(f, fieldnames=FIELDNAMES).writeheader()
19
20     def _norm(self, s: str) -> str:
21         return s.strip().lower()
22
23     def get_by_username(self, username: str) -> Optional[Dict]:
24         username = self._norm(username)
25         for row in self.repo.read_all(self.path):
26             if self._norm(row["username"]) == username:
27                 row["id"] = int(row["id"])
28                 return row
29         return None
30
31     def create_user(self, *, username: str, email: str, password_hash: str) -> Dict:
32         rows = self.repo.read_all(self.path)
33
34         username = self._norm(username)
```

```
35     email = self._norm(email)
36     rows = self.repo.read_all(self.path)
37
38     if any(self._norm(r["username"]) == username for r in rows):
39         raise ValueError("username_taken")
40     if any(self._norm(r["email"]) == email for r in rows):
41         raise ValueError("email_taken")
42
43     new_id = 1 if not rows else max(int(r["id"]) for r in rows) + 1
44     rec = {
45         "id": str(new_id),
46         "username": username,
47         "email": email,
48         "password_hash": password_hash,
49     }
50     rows.append(rec)
51     self.repo.write_all(self.path, FIELDNAMES, rows)
52     rec["id"] = new_id
53     return rec
```

« prev ^ index » next coverage.py v7.11.3, created at 2025-11-20 19:17 -0800