```python
import csv
import os
import time
from classes import *

def setup_test_files():
    """Create initial test CSV files with sample data"""
    # Create staff_details.csv if it doesn't exist
    if not os.path.exists('staff_details.csv'):
        with open('staff_details.csv', 'w', newline='') as f:
            writer = csv.writer(f)
            writer.writerow(['staff_id', 'first_name', 'last_name', 'status', 'time_elapsed'])
            writer.writerow(['101', 'John', 'Doe', 'Out of Office', '0'])
            writer.writerow(['102', 'Jane', 'Smith', 'Out of Office', '0'])

    # Create call_details.csv if it doesn't exist
    if not os.path.exists('call_details.csv'):
        with open('call_details.csv', 'w', newline='') as f:
            writer = csv.writer(f)
            writer.writerow(['call_id', 'status', 'time_elapsed', 'sat_score', 'handler_id'])
            writer.writerow(['1001', 'Pending', '0', '0.8', '0'])
            writer.writerow(['1002', 'Pending', '0', '0.6', '0'])


def test_manager_functions():
    """Test all Manager class functions"""
    print("\n=== TESTING MANAGER FUNCTIONS ===")

    # Initialize manager with some staff
    manager = Manager(id=1, first_name="Alice", last_name="Johnson", staff_list=[101, 102])

    # Test view staff details
    print("\nViewing staff details:")
    manager.view_staff_detail(101)
    manager.view_staff_detail_selected(102, ['first_name', 'last_name'])

    # Test add staff
    print("\nAdding new staff:")
    manager.add_staff(103, "Bob", "Williams")
    manager.add_staff(103, "Bob", "Williams")  # Should show exists message

    # Test edit staff name
    print("\nEditing staff name:")
    manager.edit_staff_name(103, "Robert", "Williams")
    manager.view_staff_detail_selected(103, ['first_name', 'last_name'])

    # Test remove staff
    print("\nRemoving staff:")
    manager.remove_staff(103)
    manager.view_staff_detail(103)


def test_staff_functions():
    print("\n=== TESTING STAFF FUNCTIONS ===")

    # Initialize staff member
    staff = Staff(
        id=101,
        first_name="John",
        last_name="Doe",
        manager_id=1,
        calls_taken=0,
        successful_calls=0,
        failed_calls=0,
        target_successful_calls=10,
        working_time_elapsed=0,
        avg_sat_score=0,
        call_status="Free"
    )

    # Initialize test calls
    call1 = Call(id=1001, status="Pending")
    call2 = Call(id=1002, status="Pending")

    # Test workday functions
    print("\nTesting workday functions:")
    staff.start_workday()
    time.sleep(10)  # Simulate working

    # Test call handling
    print("\nAccepting and ending calls:")
    staff.accept_call(call1)
    print(f"Call 1 status: {call1.status}, Handler: {call1.handler_id}")
    time.sleep(20)
    staff.end_call(call1, 0.56)
    print(f"Call 1 status: {call1.status}, Duration: {call1.time_elapsed:.2f}s")

    staff.accept_call(call2)
    time.sleep(10)
    staff.end_call(call2, 0.9)
    print(f"Call 1 status: {call1.status}, Duration: {call1.time_elapsed:.2f}s")


    # Test call history
    print("\nViewing call history:")
    staff.see_call_history()

    # End workday
    staff.end_workday()
    print(f"\nTotal calls taken: {staff.calls_taken}")
    print(f"Successful calls: {staff.successful_calls}")
    print(f"Failed calls: {staff.failed_calls}")


def main():
    setup_test_files()

    try:
        test_manager_functions()
        test_staff_functions()

        print("\n=== FINAL STAFF DETAILS ===")
        with open('staff_details.csv', 'r') as f:
            print(f.read())

        print("\n=== FINAL CALL DETAILS ===")
        with open('call_details.csv', 'r') as f:
            print(f.read())

    except Exception as e:
        print(e)

    finally:
```

```
        pass

if __name__ == "__main__":
    main()
```