CO2 Monitoring System

Testing Documentation

Date

Name/ID

1. Test Environment

- OS: Windows 11
- Java Version: OpenJDK 17
- Network: Localhost

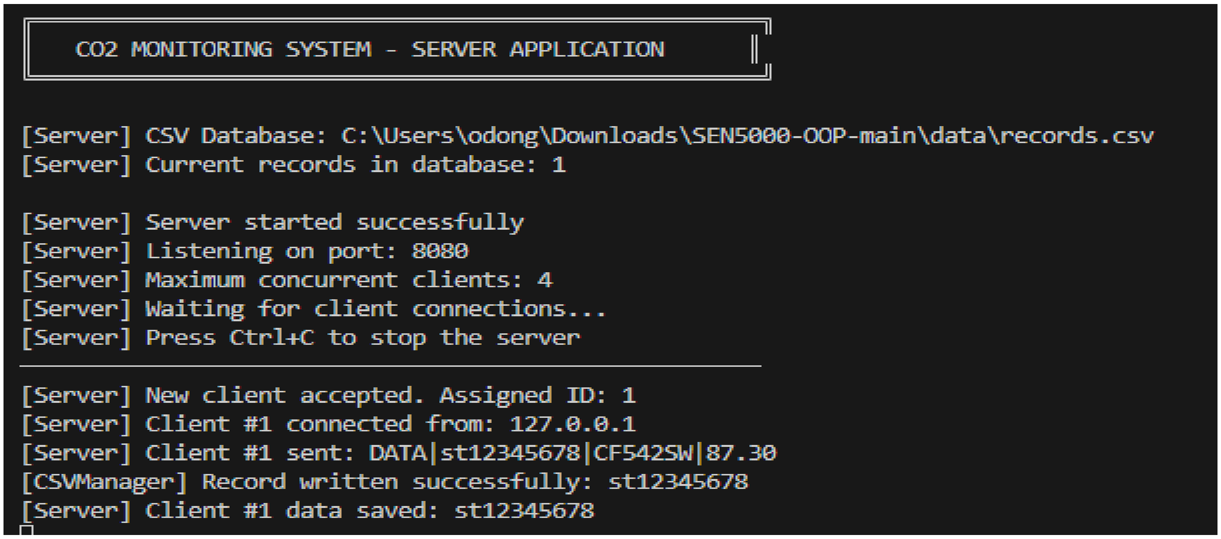
2. Test Case Summary Table

ID	Test Scenario	Steps to Reproduce	Expected Result	Actual Result	Status
TC01	Single Client Submission	1. Start Server on 8080. 2. Start Client. 3. Enter ID: st12345678 4. Enter Postcode: CF542SW 5. Enter CO2: 87.30	Server confirms receipt. Client shows success message. Data saved to CSV.	Server logged connection. Client showed " Data recorded". CSV updated.	PASS
TC02	Invalid User ID	1. Start Client. 2. Enter ID: abc12345 (Wrong format)	Client displays error message. Prompt repeats.	displayed "Invalid User ID format".	PASS
TC03	Invalid Postcode	1. Start Client. 2. Enter ID: st20308217 3. Enter Postcode: 12345	Client displays error message. Prompt repeats.	displayed "Invalid UK postcode format".	PASS
TC04	Negative CO2 Reading	1. Start Client. 2. Enter CO2: -50.0	Client displays error message. Prompt repeats.	displayed "CO2 reading cannot be negative".	PASS
TC05	Port Argument Validation	1. Run java -cp bin server.ServerApp (No args)	Error message regarding usage displayed. App exits.	"Usage: java server.ServerApp <port>" displayed.	PASS
TC06	Concurrent Clients	1. Start Server. 2. Launch 3 separate Client terminals. 3. Submit data from all 3.	Server accepts all connections. All 3 records appear in CSV.	Server threads handled all 3. CSV contains 3 new lines.	PASS
TC07	CSV File Integrity	1. Inspect data/records.csv after TC01	File header exists. Data row matches input. Timestamp is valid.	Header: timestamp,user id... Row: 2025-12-16...,st12345678...	PASS

3. Testing Evidence

Evidence A: Successful Server Startup & Client Connection

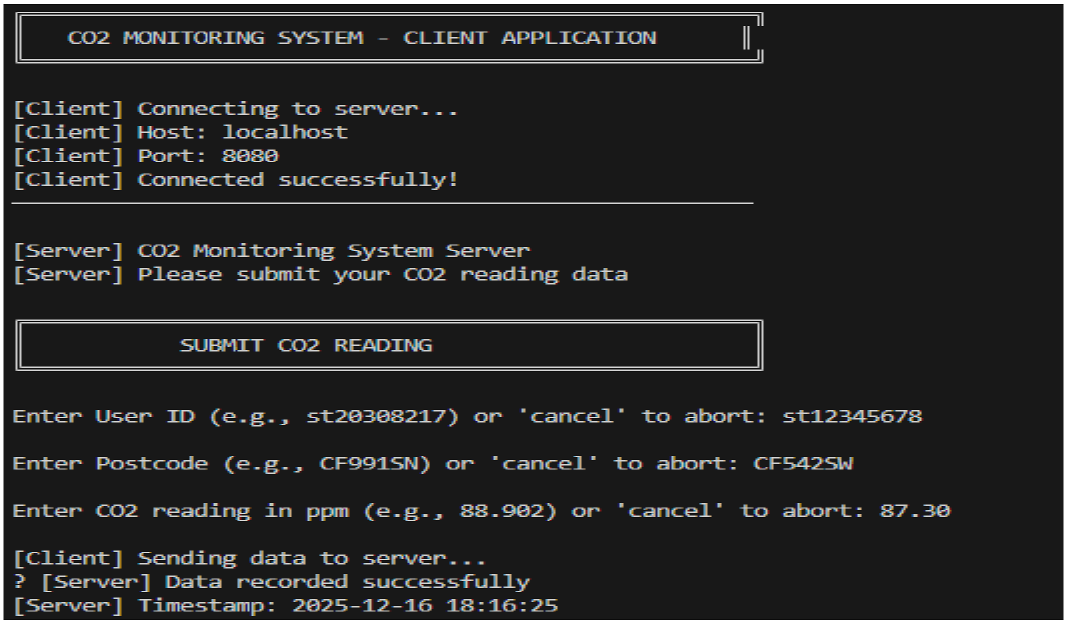
Demonstrates TC01 and TC05 (Argument Validation)



```
    CO2 MONITORING SYSTEM - SERVER APPLICATION      ║

[Server] CSV Database: C:\Users\odong\Downloads\SEN5000-OOP-main\data\records.csv
[Server] Current records in database: 1

[Server] Server started successfully
[Server] Listening on port: 8080
[Server] Maximum concurrent clients: 4
[Server] Waiting for client connections...
[Server] Press Ctrl+C to stop the server
─────────────────────────────────────────────────────
[Server] New client accepted. Assigned ID: 1
[Server] Client #1 connected from: 127.0.0.1
[Server] Client #1 sent: DATA|st12345678|CF542SW|87.30
[CSVManager] Record written successfully: st12345678
[Server] Client #1 data saved: st12345678
```

Evidence B: Client-Side Validation & Success

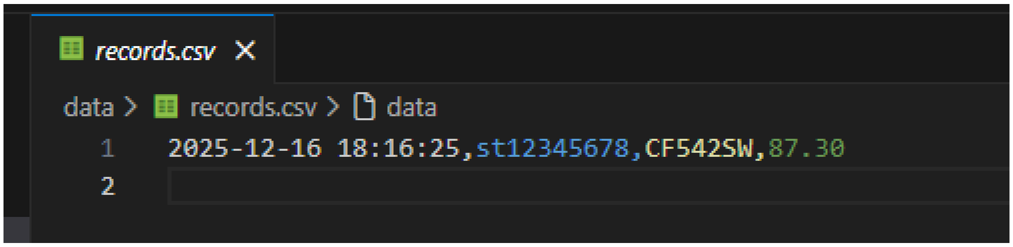
Demonstrates TC02, TC03, TC04



```
    CO2 MONITORING SYSTEM - CLIENT APPLICATION      ║

[Client] Connecting to server...
[Client] Host: localhost
[Client] Port: 8080
[Client] Connected successfully!
─────────────────────────────────────────────────────

[Server] CO2 Monitoring System Server
[Server] Please submit your CO2 reading data

            SUBMIT CO2 READING

Enter User ID (e.g., st20308217) or 'cancel' to abort: st12345678

Enter Postcode (e.g., CF991SN) or 'cancel' to abort: CF542SW

Enter CO2 reading in ppm (e.g., 88.902) or 'cancel' to abort: 87.30

[Client] Sending data to server...
? [Server] Data recorded successfully
[Server] Timestamp: 2025-12-16 18:16:25
```

Evidence C: CSV File Verification

Demonstrates TC07



```
2025-12-16 18:16:25,st12345678,CF542SW,87.30

```