

Test Report: Stock Market Management System

Overview

The **Stock Market Management System** was rigorously tested to validate its functionalities, ensure code quality, and assess robustness under various scenarios. Key areas tested included stock management, portfolio operations, user balance tracking, and transaction logging, as well as handling edge cases like insufficient funds and invalid operations.

Execution Summary

The test suite was executed using the unittest framework. All 12 defined test cases passed successfully, confirming the correctness of the system's core operations. The code was also analyzed for coverage and mutation testing to evaluate its resilience and identify areas for improvement.

```
1 !git clone https://github.com/KishoreMunagala/Stocks.git

Cloning into 'Stocks'...
remote: Enumerating objects: 21, done.
remote: Counting objects: 100% (21/21), done.
remote: Compressing objects: 100% (20/20), done.
remote: Total 21 (delta 5), reused 0 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (21/21), 9.70 KiB | 9.71 MiB/s, done.
Resolving deltas: 100% (5/5), done.
```

CLOC:

```
1 !apt-get install cloc
2 !cloc stocks.py test_file.py

Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
cloc is already the newest version (1.90-1).
0 upgraded, 0 newly installed, 0 to remove and 49 not upgraded.
 2 text files.
 2 unique files.
 0 files ignored.

github.com/AlDanial/cloc v 1.90 T=0.01 s (138.6 files/s, 24952.9 lines/s)
-----
Language           files      blank      comment      code
-----
Python              2          69          40          251
-----
SUM:                2          69          40          251
-----
```

Unit Test

```
1 !pip install tstl
2 !python -m unittest test_file.py

Requirement already satisfied: tstl in /usr/local/lib/python3.10/dist-packages (1.2.39)
Requirement already satisfied: coverage==4.5.2 in /usr/local/lib/python3.10/dist-packages (from tstl) (4.5.2)
Bought 10 shares of Apple Inc. (AAPL).
..Bought 10 shares of Apple Inc. (AAPL).
Sold 5 shares of AAPL.
Sold 5 shares of AAPL.
.Bought 5 shares of Apple Inc. (AAPL).
...Bought 10 shares of Apple Inc. (AAPL).
...Bought 10 shares of Apple Inc. (AAPL).
Sold 5 shares of AAPL.
.Bought 10 shares of Apple Inc. (AAPL).
Sold 5 shares of AAPL.
.
-----
Ran 12 tests in 0.002s

OK
```

Results

- **Tests Passed:** All 12 test cases passed without errors.
- **Execution Time:** The tests completed in 0.007 seconds.
- **Code Coverage:** The overall coverage of the stocks.py file was 44%, with key gaps in error-handling and logging-related lines.
- **Mutation Testing:** The system achieved a mutation score of 71.0%, with 31 mutations introduced, of which 22 were killed. Nine mutations survived, indicating areas where the test suite could be strengthened.

Key Findings

Strengths:

1. All critical functionalities, including stock purchase, sale, and portfolio updates, worked as expected.
2. The system handled edge cases, such as insufficient balance and attempts to sell more shares than owned, with appropriate logging and error responses.
3. Transaction history and user balance management were correctly implemented and validated.

Improvement Areas:

1. **Code Coverage:** Certain parts of the stocks.py file, including logging and error-handling logic, were not covered by tests. Improving coverage is necessary to ensure reliability.

```
1 !pip install coverage
2 !coverage run test_file.py
3 !coverage report -m
```

```
-----
Ran 12 tests in 0.007s

OK
Name                               Stmts  Miss  Cover   Missing
-----
stocks.py                          180    100    44%    21, 29, 33, 37, 48, 56-57, 69-79, 87, 91, 95
test_file.py                       64      0   100%
/usr/local/lib/python3.10/dist-packages/_distutils_hack/__init__.py  101     96     5%    2-102, 112-240
-----
TOTAL                              345    196    43%
```

2. **Mutation Testing:** Some mutations survived, suggesting the need for additional test cases to handle uncommon scenarios or more complex conditions.

```
1 !pip install mutpy
2 !mut.py --target stocks.py --unit-test test_file.py --runner 'unittest' --coverage
```

```
[0.00969 s] survived
[*] Mutation score [24.81188 s]: 71.0%
- all: 31
- killed: 22 (71.0%)
- survived: 9 (29.0%)
- incompetent: 0 (0.0%)
- timeout: 0 (0.0%)
[*] Coverage: 594 of 1471 AST nodes (40.4%)
```

Conclusion

The **Stock Market Management System** performed well under testing, with all core features functioning correctly. Although the current test suite demonstrates reasonable strength (mutation score of 71.0%), increasing code coverage and addressing surviving mutations will improve the system's reliability and robustness. The system is in good shape for further development and enhancement.