Project Name	**Branch**	**Analysis Date**
5_long_banking_system	⑂ main	2022-12-25

Size Rating	**Lines of Code**		**Version**	**Quality Gate**
S	1,211		not provided	PASSED

Reliability

E

Bugs

1

Reliability on New Code

–

New Bugs

Security

A

Vulnerabilities

0

Security Hotspots

1

Security on New Code

–

New Vulnerabilities

New Security Hotspots

Maintainability

A

Code Smells

31

Debt Ratio

0.7%

Maintainability on New Code

–

New Code Smells

Debt Ratio on New Code

0

Issues By Severity

	🐞	🔓	⚙
❗	1	0	0
⬆	0	0	19
🔼	0	0	10
🔽	0	0	2
ℹ	0	0	0

❗ Blocker   🐞 Bug
🔴 Critical   🔓 Vulnerability
🔼 Major   ⚙ Code Smell
🔽 Minor
ℹ Info

Coverage

0%

Unit Tests

0

Coverage on New Code

0

Duplications

0%

Duplicated Blocks

0

Duplications on New Code

0

Project Name
5_long_banking_system

Branch main
Version not provided

Analysis Date
2022-12-25

Bugs

Vulnerabilities

Code Smells

Security Hotspots

Duplications (Lines of code / Duplicated lines)

Duplicated Lines (%)

Coverage (Lines to cover / Covered lines)

Coverage (%)

Project Name
5_long_banking_system

Branch main
Version not provided

Analysis Date
2022-12-25

Top Common Issues

Severity	Rule	Type	# Issues
⬆	String literals should not be duplicated (py)	☢	9
⬆	Wildcard imports should not be used (py)	☢	7
⌃	Sections of code should not be commented out (py)	☢	6
⬆	Cognitive Complexity of functions should not be too high (py)	☢	3
⌃	Builtins should not be shadowed by local variables (py)	☢	2
⬇	Unused local variables should be removed (py)	☢	2
⌃	Unused function parameters should be removed (py)	☢	1
❗	Variables, classes and functions should be defined before being used (py)	🐛	1
⌃	Unused assignments should be removed (py)	☢	1

Severity Levels
- ❗ Blocker
- ⬆ Critical
- ⌃ Major
- ⬇ Minor
- ℹ Info

🔗 Multipe severities found

Issue Types
- 🐛 Bug
- 🔓 Vulnerability
- 🛡 Security Hotspot
- ☢ Code Smell

Top Issues By Severity

Severity	Rule	Type	# Issues
🛑	Variables, classes and functions should be defined before being used (py)		1
🔴	String literals should not be duplicated (py)	⚙	9
🔴	Wildcard imports should not be used (py)	⚙	7
🔴	Cognitive Complexity of functions should not be too high (py)	⚙	3
🔺	Sections of code should not be commented out (py)	⚙	6
🔺	Builtins should not be shadowed by local variables (py)	⚙	2
🔺	Unused assignments should be removed (py)	⚙	1
🔺	Unused function parameters should be removed (py)	⚙	1
🟢	Unused local variables should be removed (py)	⚙	2

Severity Levels
- 🛑 Blocker
- 🔴 Critical
- 🔺 Major
- 🟢 Minor
- ⓘ Info
- 🔗 Multipe severities found

Issue Types
- 🐛 Bug
- 🔓 Vulnerability
- 🛡 Security Hotspot
- ⚙ Code Smell

Project Name
5_long_banking_system

Branch main
Version not provided

Analysis Date
2022-12-25

5

Top Security HotSpots to Review

Severity	Rule	Type	# Issues
⬆	Using pseudorandom number generators (PRNGs) is security-sensitive (py)	🛡	1

Severity Levels
- ❗ Blocker
- ⬆ Critical
- 🔺 Major
- 🔻 Minor
- ℹ Info

🔗 Multipe severities found

Issue Types
- 🐞 Bug
- 🔓 Vulnerability
- 🛡 Security Hotspot
- ⚙ Code Smell

sonarqube

Project Name
5_long_banking_system

Branch main
Version not provided

Analysis Date
2022-12-25

Rule

⊗ **Variables, classes and functions should be defined before being used**

Language py

Issues 1

Variables, Classes and functions should be defined before they are used, otherwise the code will fail.

Issues Breakdown

📄 services.py
field is used before it is defined. Move the definition before.

L320

sonarqube

Rule

String literals should not be duplicated

Language
py

Issues
9

Duplicated string literals make the process of refactoring error-prone, since you must be sure to update all occurrences.

🏷 **design**

Issues Breakdown

📄 account.py   **L40**
Define a constant instead of duplicating this literal "You have %d try left!!!" 10 times.

📄 account.py   **L191**
Define a constant instead of duplicating this literal "You enter invalid choice many times, please wait a f...

📄 services.py   **L39**

📄 services.py   **L43**
Define a constant instead of duplicating this literal "You have %d try left!!!" 6 times.

📄 services.py   **L48**
Define a constant instead of duplicating this literal "You enter wrong choice many times, please wait few m...

📄 transactions.py   **L31**
Define a constant instead of duplicating this literal "You have %d try left!!!" 6 times.

📄 transactions.py   **L53**
Define a constant instead of duplicating this literal "Finish action!!!" 14 times.

📄 utils.py   **L61**
Define a constant instead of duplicating this literal "You have %d try left!!!" 3 times.

📄 utils.py   **L112**
Define a constant instead of duplicating this literal "You need to be 18 years old to create a new account"...

Project Name
5_long_banking_system

Branch main
Version not provided

Analysis Date
2022-12-25

Rule
Wildcard imports should not be used

Language
py

Issues
7

Importing every public name from a module using a wildcard (from mymodule import *) is a bad idea because:
It could lead to conflicts between names defined locally and the ones imported.

🏷 **pitfall**

Issues Breakdown

account.py
Import only needed names or import the module and then use its members. **L8**

messages.py
Import only needed names or import the module and then use its members. **L5**

models.py
Import only needed names or import the module and then use its members. **L7**

services.py
Import only needed names or import the module and then use its members. **L5**

test_utils.py
Import only needed names or import the module and then use its members. **L3**

transactions.py
Import only needed names or import the module and then use its members. **L4**

utils.py
Import only needed names or import the module and then use its members. **L9**

sonarqube

Rule

Cognitive Complexity of functions should not be too high

Language
py

Issues
3

Cognitive Complexity is a measure of how hard the control flow of a function is to understand.

🏷 **brain-overload**

Issues Breakdown

📄 account.py **L14**
Refactor this function to reduce its Cognitive Complexity from 40 to the 15 allowed.

📄 services.py **L75**
Refactor this function to reduce its Cognitive Complexity from 17 to the 15 allowed.

📄 services.py **L172**
Refactor this function to reduce its Cognitive Complexity from 27 to the 15 allowed.

sonarqube

Rule

Using pseudorandom number generators (PRNGs) is security-sensitive

Language py

Issues 1

Using pseudorandom number generators (PRNGs) is security-sensitive.

cwe   owasp-a3   owasp-m5

Issues Breakdown

models.py **L61**
Make sure that using this pseudorandom number generator is safe here.

Project Name	
5_long_banking_system	

Branch main
Version not provided

Analysis Date
2022-12-25

Rule

Language
py

Issues
6

Sections of code should not be commented out

Programmers should not comment out code as it bloats programs and reduces readability.

🏷 **unused**

Issues Breakdown

📄 messages.py Remove this commented out code.	**L55**
📄 test_utils.py Remove this commented out code.	**L8**
📄 test_utils.py Remove this commented out code.	**L33**
📄 test_utils.py Remove this commented out code.	**L51**
📄 test_utils.py Remove this commented out code.	**L77**
📄 utils.py Remove this commented out code.	**L257**

sonarqube

Rule

Builtins should not be shadowed by local variables

Language
py

Issues
2

Shadowing a builtin makes your code more difficult to read and maintain.

🏷 **bad-practice** **confusing** **pitfall**

Issues Breakdown

📄 utils.py
Rename this variable; it shadows a builtin.
L239

📄 utils.py
Rename this variable; it shadows a builtin.
L250

Rule
Unused assignments should be removed

Language
py

Issues
1

A dead store happens when a local variable is assigned a value that is not read by any subsequent instruction.

🏷 **cwe**   **unused**

Issues Breakdown

📄 utils.py   **L73**
Remove this assignment to local variable 'result'; the value is never used.

Rule

Unused function parameters should be removed

Language
py

Issues
1

Unused parameters are misleading.

🏷 **unused**

Issues Breakdown

📄 utils.py   **L147**
Remove the unused function parameter "email".

sonarqube

Project Name
5_long_banking_system

Branch main
Version not provided

Analysis Date
2022-12-25

Rule
Unused local variables should be removed

Language
py

Issues
2

If a local variable is declared but not used, it is dead code and should be removed.

🏷 unused

Issues Breakdown

services.py
Remove the unused local variable "phone_number".
L300

services.py
Remove the unused local variable "phone_number".
L327