sonarqube

Executive Report

Project Name
17_Media_player_git

Branch
main

Analysis Date
2023-01-04

Size Rating
XS

Lines of Code
677

Version
not provided

Quality Gate
PASSED

Reliability

A

Bugs
0

Reliability on New Code
–

New Bugs

Security

A

Vulnerabilities
0

Security Hotspots
0

Security on New Code
–

New Vulnerabilities

New Security Hotspots

Maintainability

A

Code Smells
61

Debt Ratio
1%

Maintainability on New Code
–

New Code Smells

Debt Ratio on New Code
0

Issues By Severity

	🐞	🔓	⚙
❗	0	0	0
⬆	0	0	17
🔺	0	0	9
🔽	0	0	35
ℹ	0	0	0

- ❗ Blocker
- 🔴 Critical
- 🔺 Major
- 🔽 Minor
- ℹ Info

- 🐞 Bug
- 🔓 Vulnerability
- ⚙ Code Smell

Coverage

0%

Unit Tests
0

Coverage on New Code
0

Duplications

2.1%

Duplicated Blocks
2

Duplications on New Code
0
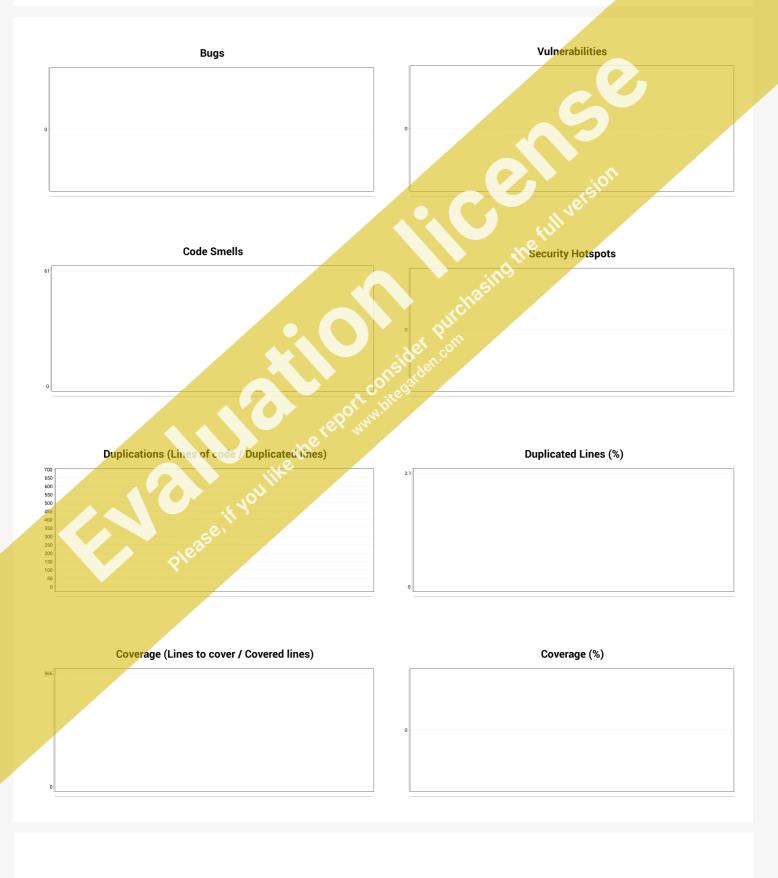
sonarqube

Activity

Project Name
17_Media_player_git

Branch main
Version not provided

Analysis Date
2023-01-04

Bugs

0

Vulnerabilities

0

Code Smells

61

0

Security Hotspots

0

Duplications (Lines of code / Duplicated lines)

700
650
600
550
500
450
400
350
300
250
200
150
100
50
0

Duplicated Lines (%)

2.1

0

Coverage (Lines to cover / Covered lines)

566

0

Coverage (%)

0

sonarqube

Project Name	
17_Media_player_git	

Branch	main
Version	not provided

Analysis Date
2023-01-04

Top Common Issues

Severity	Rule	Type	# Issues
Minor	Jump statements should not be redundant (py)		22
Critical	"SystemExit" should be re-raised (py)	Code Smell	11
Minor	"pass" should not be used needlessly (py)	Code Smell	8
Critical	Wildcard imports should not be used (py)	Code Smell	5
Major	Nested blocks of code should not be left empty (py)	Code Smell	4
Minor	Unused local variables should be removed (py)	Code Smell	4
Major	Sections of code should not be commented out (py)	Code Smell	3
Critical	String literals should not be duplicated (py)	Code Smell	1
Major	Functions and methods should not have identical implementations (py)	Code Smell	1
Major	The "print" statement should not be used (py)	Code Smell	1
Minor	Method names should comply with a naming convention (py)	Code Smell	1

Severity Levels
- Blocker
- Critical
- Major
- Minor
- Info
- Multipe severities found

Issue Types
- Bug
- Vulnerability
- Security Hotspot
- Code Smell

sonarqube

Project Name
17_Media_player_git

Branch main
Version not provided

Analysis Date
2023-01-04

Top Issues By Severity

Severity	Rule	Type	# Issues
⭱	"SystemExit" should be re-raised (py)		11
⭱	Wildcard imports should not be used (py)		5
⭱	String literals should not be duplicated (py)		1
⭡	Nested blocks of code should not be left empty (py)		4
⭡	Sections of code should not be commented out (py)		3
⭡	Functions and methods should not have identical implementations (py)		1
⭡	The "print" statement should not be used (py)		1
⭣	Jump statements should not be redundant (py)		22
⭣	"pass" should not be used needlessly (py)		8
⭣	Unused local variables should be removed (py)		4
⭣	Method names should comply with a naming convention (py)		1

Severity Levels
- 🅾 Blocker
- ⭱ Critical
- ⭡ Major
- ⭣ Minor
- ⓘ Info

🔍 Multipe severities found

Issue Types
- 🐛 Bug
- 🔓 Vulnerability
- ⛨ Security Hotspot
- ⚛ Code Smell

bitegarden

Project Name
17_Media_player_git

Branch main
Version not provided

Analysis Date
2023-01-04

Top Security HotSpots to Review

Severity	Rule	Type	# Issues

No Security HotSpots found

Severity Levels
- 🛑 Blocker
- ⭕ Critical
- 🔺 Major
- 🔽 Minor
- ℹ️ Info

⚙️ Multipe severities found

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

sonarqube

Project Name
17_Media_player_git

Branch main
Version not provided

Analysis Date
2023-01-04

Rule
"SystemExit" should be re-raised

Language
py

Issues
11

SystemExit is raised when sys.exit() is called.

`bad-practice`   `error-handling`   `suspicious`

Issues Breakdown

Python Media Player Version 0.0.1/GUI.py Specify an exception class to catch or reraise the exception	**L37**
Python Media Player Version 0.0.1/Tools/Controls.py Specify an exception class to catch or reraise the exception	**L23**
Python Media Player Version 0.0.1/Tools/Controls.py Specify an exception class to catch or reraise the exception	**L176**
Python Media Player Version 0.0.1/Tools/Controls.py Specify an exception class to catch or reraise the exception	**L191**
Python Media Player Version 0.0.1/Tools/DisplayPanel.py Specify an exception class to catch or reraise the exception	**L43**
Python Media Player Version 0.0.1/Tools/Gui.py Specify an exception class to catch or reraise the exception	**L34**
Python Media Player Version 0.0.1/Tools/ListPanel.py Specify an exception class to catch or reraise the exception	**L51**
Python Media Player Version 0.0.1/Tools/player.py Specify an exception class to catch or reraise the exception	**L71**
Python Media Player Version 0.0.1/Tools/player.py Specify an exception class to catch or reraise the exception	**L103**
Python Media Player Version 0.0.1/Tools/player.py Specify an exception class to catch or reraise the exception	**L128**
Python Media Player Version 0.0.1/Tools/player.py Specify an exception class to catch or reraise the exception	**L179**

sonarqube

Rule

Wildcard imports should not be used

Language py

Issues 5

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

📄 Python Media Player Version 0.0.1/Tools/Controls.py
Import only needed names or import the module and then use its members.

L19

📄 Python Media Player Version 0.0.1/Tools/DisplayPanel.py
Import only needed names or import the module and then use its members.

L47

📄 Python Media Player Version 0.0.1/Tools/Gui.py
Import only needed names or import the module and then use its members.

L38

📄 Python Media Player Version 0.0.1/Tools/ListPanel.py
Import only needed names or import the module and then use its members.

L58

📄 Python Media Player Version 0.0.1/Tools/player.py
Import only needed names or import the module and then use its members.

L38

bitegarden

Rule
String literals should not be duplicated

Language
py

Issues
1

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

design

Issues Breakdown

Python Media Player Version 0.0.1/Tools/ListPanel.py   **L127**
Define a constant instead of duplicating this literal '[+] Song Variable Update Path : {}' 3 times.

Project Name
17_Media_player_git

Branch   main
Version   not provided

Analysis Date
2023-01-04

Rule
Nested blocks of code should not be left empty

Language
py

Issues
4

Most of the time a block of code is empty when a piece of code is really missing.

🏷 **suspicious**

Issues Breakdown

📄 Python Media Player Version 0.0.1/Tools/Controls.py   **L53**
Either remove or fill this block of code.

📄 Python Media Player Version 0.0.1/Tools/ListPanel.py   **L116**
Either remove or fill this block of code.

📄 Python Media Player Version 0.0.1/Tools/ListPanel.py   **L120**
Either remove or fill this block of code.

📄 Python Media Player Version 0.0.1/Tools/ListPanel.py   **L142**
Either remove or fill this block of code.

sonarqube

Project Name
17_Media_player_git

Branch main
Version not provided

Analysis Date
2023-01-04

Rule

Sections of code should not be commented out

Language
py

Issues
3

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

🏷 unused

Issues Breakdown

📄 Python Media Player Version 0.0.1/Tools/Controls.py **L41**
Remove this commented out code.

📄 Python Media Player Version 0.0.1/Tools/DisplayPanel.py **L68**
Remove this commented out code.

📄 Python Media Player Version 0.0.1/Tools/player.py **L32**
Remove this commented out code.

sonarqube

Project Name
17_Media_player_git

Branch main
Version not provided

Analysis Date
2023-01-04

Rule

Functions and methods should not have identical implementations

Language
py

Issues
1

When two functions or methods have the same implementation, either it was a mistake - something else was intended - or the duplication was intentional, but may be confusing to maintainers.

🏷️ confusing   duplicate   suspicious

Issues Breakdown

📄 Python Media Player Version 0.0.1/Tools/Controls.py   **L181**
Update this function so that its implementation is not identical to previous on line 166.

sonarqube

Project Name
17_Media_player_git

Branch main
Version not provided

Analysis Date
2023-01-04

Rule

The "print" statement should not be used

Language
py

Issues
1

The print statement was removed in Python 3.0.

🏷️ **obsolete   python3**

Issues Breakdown

📄 Python Media Player Version 0.0.1/Tools/player.py **L120**
Replace print statement by built-in function.

Project Name
17_Media_player_git

Branch   main
Version   not provided

Analysis Date
2023-01-04

Rule
Jump statements should not be redundant

Language
py

Issues
22

Jump statements, such as return, break and continue let you change the default flow of program execution, but jump statements that direct the control flow to the original direction are just a waste of keystrokes.

🏷 clumsy   redundant

Issues Breakdown

📄 Python Media Player Version 0.0.1/Tools/Controls.py
Remove this redundant return.   **L46**

📄 Python Media Player Version 0.0.1/Tools/Controls.py
Remove this redundant return.   **L99**

📄 Python Media Player Version 0.0.1/Tools/Controls.py
Remove this redundant return.   **L113**

📄 Python Media Player Version 0.0.1/Tools/Controls.py
Remove this redundant return.   **L160**

📄 Python Media Player Version 0.0.1/Tools/Controls.py
Remove this redundant return.   **L164**

📄 Python Media Player Version 0.0.1/Tools/DisplayPanel.py
Remove this redundant return.   **L64**

📄 Python Media Player Version 0.0.1/Tools/DisplayPanel.py
Remove this redundant return.   **L77**

📄 Python Media Player Version 0.0.1/Tools/DisplayPanel.py
Remove this redundant return.   **L92**

📄 Python Media Player Version 0.0.1/Tools/DisplayPanel.py
Remove this redundant return.   **L97**

📄 Python Media Player Version 0.0.1/Tools/DisplayPanel.py
Remove this redundant return.   **L103**

📄 Python Media Player Version 0.0.1/Tools/ListPanel.py
Remove this redundant return.   **L121**

📄 Python Media Player Version 0.0.1/Tools/ListPanel.py
Remove this redundant return.   **L128**

📄 Python Media Player Version 0.0.1/Tools/ListPanel.py
Remove this redundant return.   **L133**

📄 Python Media Player Version 0.0.1/Tools/ListPanel.py
Remove this redundant return.   **L145**

📄 Python Media Player Version 0.0.1/Tools/ListPanel.py
Remove this redundant return.   **L153**

📄 Python Media Player Version 0.0.1/Tools/player.py
Remove this redundant return.   **L89**

Issues Breakdown

Python Media Player Version 0.0.1/Tools/player.py   **L93**
Remove this redundant return.

Python Media Player Version 0.0.1/Tools/player.py   **L97**
Remove this redundant return.

Python Media Player Version 0.0.1/Tools/player.py   **L105**
Remove this redundant return.

Python Media Player Version 0.0.1/Tools/player.py   **L109**
Remove this redundant return.

Python Media Player Version 0.0.1/Tools/player.py   **L139**
Remove this redundant return.

Python Media Player Version 0.0.1/Tools/player.py   **L163**
Remove this redundant return.

sonarqube

Rule

Language | | Issues
py | | **8**

"pass" should not be used needlessly

The use of a pass statement where it's not required by the syntax is pure cruft and should be removed.

🏷️ unused

Issues Breakdown

Python Media Player Version 0.0.1/Tools/Controls.py
Remove this unneeded "pass".   **L58**

Python Media Player Version 0.0.1/Tools/Controls.py
Remove this unneeded "pass".   **L61**

Python Media Player Version 0.0.1/Tools/Controls.py
Remove this unneeded "pass".   **L175**

Python Media Player Version 0.0.1/Tools/Controls.py
Remove this unneeded "pass".   **L190**

Python Media Player Version 0.0.1/Tools/DisplayPanel.py
Remove this unneeded "pass".   **L82**

Python Media Player Version 0.0.1/Tools/DisplayPanel.py
Remove this unneeded "pass".   **L85**

Python Media Player Version 0.0.1/Tools/player.py
Remove this unneeded "pass".   **L117**

Python Media Player Version 0.0.1/Tools/player.py
Remove this unneeded "pass".   **L121**

sonarqube

Rule

Unused local variables should be removed

Language
py

Issues
4

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

🏷 unused

Issues Breakdown

📄 Python Media Player Version 0.0.1/Tools/Controls.py   **L67**
Remove the unused local variable "add_fileicon".

📄 Python Media Player Version 0.0.1/Tools/Controls.py   **L68**
Remove the unused local variable "add_directoryicon".

📄 Python Media Player Version 0.0.1/Tools/Controls.py   **L69**
Remove the unused local variable "exiticon".

📄 Python Media Player Version 0.0.1/Tools/Controls.py   **L79**
Remove the unused local variable "delete_selectedicon".

Code Smell
S100

Project Name
17_Media_player_git

Branch   main
Version   not provided

Analysis Date
2023-01-04

Rule
Method names should comply with a naming convention

Language
py

Issues
1

Sharing some naming conventions is a key point to make it possible for a team to efficiently collaborate.

🏷 **convention**

Issues Breakdown

📄 Python Media Player Version 0.0.1/Tools/Controls.py   **L181**
Rename method "Next" to match the regular expression ^[a-z_][a-z0-9_]*$.