**CODE REVIEW CHECKLIST FOR EXPENSE TRACKER**

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Code Review Points** | **Status (Yes/No)** |
| 1 | Formatting the code (Refer the below steps with screen shots ) | yes |
| 2 | Remove Unused Import Statements | yes |
| 3 | Add single line comments about the class and methods | yes |
| 4 | Use try-catch block for Exception Handling, Try with resource if required (File, stream) | yes |
| 5 | Don’t provide junk name while creating variables, method name and etc (Follow the camel cases on variable declaration) | yes |
| 6 | Avoid duplicate code | yes |
| 7 | Limit the accessibility of packages, classes, interfaces, methods, and fields based on the requirement and functionality | yes |
| 8 | Don’t hard code the value, use Properties/Constant files | yes |
| 9 | Use Loggers instead of SOP | Not applicable |
| 10 | Always do the Null checks | yes |
| 11 | Try to avoid Nested for loops/for each | Not applicable |
| 12 | ST Take necessary steps to block and avoid any cross-scripting attacks, SQL injection, and other security holes. | Not applicable |
| 13 | Disposing of Unmanaged Resources like File I/O, Network resources, etc. They should to be disposed of once their usage is completed. | Not applicable |
| 14 | Run Sonar before check-in the code | yes |
| 15 | Peer code review | yes |