Walkthrough is an informal meeting for evaluation, usually no preparation is required for this. While inspection is a formal way of meeting where in the moderator plans the meeting with fixed roles assigned to all members involved. But walkthrough and inspection are static unit testing, where the evaluators will look at the code itself, analyze, understand and find defects or errors.

For me it is better to use the inspection code review methodology because it is a formal review type of review, which has the main purpose to find the defect and it can also spot any process improvement if any. It is led by a moderator and not an author of the code. It bases its review on rules and checklist and makes use of entry and exit criteria. The inspection record-keeper list the findings, which include metrics that can be used to aid improvements to the process as well as correcting defects in the document under review. So I think that inspections is better than walkthrough because it has goals and limits.

Maximum of 5 and minimum of 3 person, because it is the first test and this 5 people are the best among the rest. The head of the team is the moderator, which is in charge of the meeting schedule, the pace of the evaluation process, the reviewers to participate. A record keeper, that documents the problems found during the review and the follow-up actions suggested. Reviewer which is an expert in the subject area of the code under review. The group that should go is those with the reviewers and the record keeper and moderator because all of them are needed to have a successful review.

-Does the code do what has been specified in the design specification?

- Is the code portable? The source code is likely to execute on multiple processor architectures and on different operating systems over its lifetime. It must be implemented in a manner that does not preclude this kind of a variety of execution environments.

-Does the procedure used in the module solve the problem correctly?

-Is the code efficient? In general, clarity, readability, or correctness should not be sacrificed for efficiency. Code review is intended to detect implementation choices that have adverse effects on system performance.

-Does a software module duplicate another existing module which could be reused?

-Are there computations using variables with inconsistent data types? Is overflow or underflow a possibility?

-Are error codes and condition messages produced by accessing a common table of messages? Each error code should have a meaning, and all of the meanings should be available at one place in a table rather than scattered all over the program code.

5.) Aside from using the IBM concept which is the inspection, walkthrough can be used but, it is most preferable to use the IBM concept, because it is more reliable and efficient for it has limitation and scopes that the product should have.

6.The special role that the record keeper, he or she is the one that takes down notes during the evaluation, about the error and defects found in the product, and writes down the follow up suggestion from the viewers.

7.) Code review rework and validation is important in making a product, because you need to make a quality and reliable product that satisfy the needs of the client. Code review rework is the evaluation of the program where in it is still in the process of closing the project. It is needed for the author of the code can make sure that there are few errors and defects. And so they can be able to fix it. While validation is needed because it is the actual testing of the product to ensure the compliance with software requirements at the end of the software development.