<u>Strategic Approach to Software Testing</u>

1. Understand the Requirements

- Clearly understand what the software is supposed to do.

- Identify the key features and functions.

- Talk to users or stakeholders to ensure the test covers everything important.

2. Plan Your Testing

- Make a plan that includes:

  o **Scope**: What will you test?

  o **Objectives**: What are your goals (e.g., finding bugs, ensuring usability)?

  o **Resources**: Who will test, what tools are needed?

  o **Schedule**: Set time for each testing phase.

3. Choose Testing Methods

- **Manual Testing**: Test the software by hand.

- **Automated Testing**: Use scripts or tools to automate repetitive tests.

- **Functional Testing**: Ensure every feature works as expected.

- **Non-Functional Testing**: Check performance, security, and usability.

4. Prepare Test Cases

- Write down detailed steps on how to test each feature.

- Include input data, expected outcomes, and actual results.

5. Execute the Tests

- Run the test cases.

- Report any issues or bugs found during testing.

6. Track and Fix Bugs

- Use a bug tracking tool to document bugs.

- Developers will fix them, and testers will retest the software to make sure it works.

7. Review and Improve

- After testing, review the process.

- Make changes to improve the next round of testing.