Task 1: Bug Logging Template

Here is a structured Bug Logging Template that can be used to report and track software bugs efficiently.

Bug Report Template:-

Field	Details
Bug ID	(Auto-generated or manually assigned unique ID)
Title	Concise and descriptive title of the bug
Reported By	Name of the tester reporting the bug
Reported Date	Date when the bug was reported
Module/Feature	The specific module or feature affected
Severity	Critical / High / Medium / Low
Priority	High / Medium / Low
Environment	OS, Browser, Device, App version, etc.
Preconditions	Steps or setup required before reproducing the bug
Steps to Reproduce	Clear step-by-step instructions to replicate the issue
Expected Behavior	What should happen if the system works correctly
Actual Behavior	What is happening incorrectly due to the bug
Screenshots/Logs	Attach relevant screenshots or logs
Assigned To	Developer responsible for fixing the bug
Status	Open / In Progress / Fixed / Verified / Closed
Comments	Additional notes or developer/tester remarks

Task 2: Different Types of Testing with Examples

(1) Functional Testing

(a) Definition: Validates that the software functions as per requirements.

(b) Example: Testing a login page to ensure the correct username/password allows access while incorrect credentials show an error.

(2) Non-Functional Testing

(a) Definition: Tests aspects like performance, usability, security, etc.

(b) Example: Checking how fast a webpage loads under different network conditions.

(3) Unit Testing

(a) Definition: Testing individual components or functions in isolation.

(b) Example: Testing a function that calculates the total price in a shopping cart.

(4) Integration Testing

(a) Definition: Ensures different modules work together correctly.

(b) Example: Checking if payment processing works after adding an item to the cart.

(5) System Testing

(a) Definition: Tests the complete application to verify it meets requirements.

(b) For Example: Running tests on a fully developed e-commerce site before launch.

(6)User Acceptance Testing (UAT)

(a) Definition: End-users validate if the software meets business needs.

(b) For Example: Clients testing an HR system before deployment to check if workflows match their process.

(7) Regression Testing

(a) Definition: Ensures new code changes don’t break existing functionality.

(b) For Example: After adding a new feature, previous features like login and checkout are re-tested.

(8) Performance Testing

(a) Definition: Tests the speed, stability, and scalability of the application.

(b) For Example: Checking how many users can access a website before it crashes.

(9) Security Testing

(a) Definition: Ensures the software is protected against threats and vulnerabilities.

(b) For Example: Checking if a banking app encrypts user passwords properly.

(10) Exploratory Testing

(a) Definition: Testers explore the application without predefined test cases.

(b) For Example: Clicking random buttons and inputs to check if unexpected errors occur.