**Defect Management**

Mismatch in specifications-requirement phase (anything that doesn’t match either the customer specification.

Categories:

Wrong , Missing or ,extra implementation.

Classification:

Defect severity:

1. Critical-show stoppers- **functionality doesn’t work in the main area**.
2. Major – functionality doesn’t **work but not in the main** area(business can be provided)
3. Minor – user interface problems – not in the way of any functionality
4. Cosmetic- misspellings etc.

Priority levels: how to schedule the defect resolving set by project manager/developer

1. Blockers : Resolve immediately
2. High : resolve at the earliest, on priority at the second level(logo of business)
3. Medium: resolve the defect.
4. Low: could be solved at later stages.

Defect examples:

Requirement: Incomplete, ambiguous, and not verifiable

Design : Missing requirements, constraint, Manual design error

Coding : data related error code malfunctioning, implementation errors.

Defect log:

Reported by: tester name to be mentioned – takes significant amount of time – development team want a single persons name with whom they can contact about the defect.

Environment field : needed so as to know the env in which the defect came up and cud be slved – concentrate on that environment.

Bug id: reqd to track – identify it by unique name.

Defect life cycle:

New – when just logd

Assigned: assigned to the developer

Open – check if its duplicated/rejected(developers don’t agree that it’s a defect/deferred – developer accepted it as an issue but taken care in subsequent release if there is time./ not a bug-developers do not accept it as a defect – that’s the way it shud work

Fixed – pending retest-retest-verified-closed-reopened