

# **Defect Triage**

## **Loan Management System (Waterfall)**

**Version: 1.0**

**Release Model: Sequential SDLC (Waterfall)**

**Document Type: Defect Triage Document**

**Prepared By: QA Analyst**

**Reviewed By: Product Owner**

**Approval Status: Approved for System Testing**

## Revision History

Version	Date	Author	Description
1.0	2026-02-18	Sanyogita Herwathe (QA Owner)	Initial Waterfall Enterprise Release Version

## Table of Contents

Revision History .....	2
Approval & Sign-Off .....	2
1. Purpose .....	3
2. Triage Objectives.....	3
3. Triage Participants .....	3
4. Triage Frequency .....	3
5. Defect Classification Model.....	3
6. Triage Decision Outcomes.....	4
7. Real Examples from Waterfall Cycle .....	4
8. Governance Controls .....	5
9. Triage Flow Diagram (Textual).....	5
10. Outcome .....	5

## Approval & Sign-Off

<b>Prepared By:</b> QA – Sanyogita Herwathe
<b>Reviewed By:</b> Product Owner
<b>Approved By:</b> Engineering Manager
<b>Approval Status:</b> Approved for Release Governance

## 1. Purpose

This document defines the structured defect triage process followed during the WebLoan LMS release cycle to ensure:

- Accurate severity classification
- Business impact assessment
- Prioritized resolution
- Controlled build progression
- Risk-based release decisions

## 2. Triage Objectives

The triage process ensures:

- ✓ Critical defects are resolved immediately
- ✓ High-impact business issues are prioritized
- ✓ Duplicate or invalid defects are filtered
- ✓ Risk-based deferrals are documented
- ✓ Stakeholder alignment before release

## 3. Triage Participants

Role	Responsibility
QA Lead	Facilitates triage, validates severity
Developer	Root cause analysis & fix estimation
Business Analyst	Validates business impact
Product Owner	Approves deferrals
QA Engineer	Provides reproduction details

## 4. Triage Frequency

Conducted per build cycle
Emergency triage for Critical defects
Final triage before release sign-off

## 5. Defect Classification Model

### Severity Definitions

Critical	System unusable, core workflow blocked
High	Major feature impacted but workaround exists
Medium	Functional gap, partial impact
Low	UI, cosmetic, minor validation issue

Priority Guidelines
Priority is determined by:
Business impact
Release timeline
Customer visibility
Compliance risk
Severity ≠ Priority (Example: Cosmetic issue on homepage may be Low severity but High priority.)

## 6. Triage Decision Outcomes

Decision	Description
Fix Immediately	Critical defects
Fix in Current Build	High severity
Schedule Next Sprint	Medium defects
Defer	Business-approved non-blockers
Reject	Not reproducible / enhancement

## 7. Real Examples from Waterfall Cycle

<b>Critical Defects</b>
<ul style="list-style-type: none"> <li>• Duplicate Loan Application creation</li> <li>• Login failure for registered users</li> <li>• 500 Internal Server error on Prequalification save</li> </ul>
<b>Action:</b>
<ul style="list-style-type: none"> <li>• Immediate fix in next build</li> <li>• Regression validation performed</li> <li>• Deferred Examples</li> <li>• Search field partial matching behavior</li> <li>• Minor mobile orientation lag</li> <li>• UI label inconsistencies</li> </ul>
<b>Action:</b>
<ul style="list-style-type: none"> <li>• Business risk accepted</li> <li>• Logged for future enhancement cycle</li> </ul>

## 8. Governance Controls

- |   |
|---|
| • No Critical defects allowed for release     |
| • No unresolved High defects without approval |
| • All deferrals documented with justification |
| • Build tagging maintained                    |

## 9. Triage Flow Diagram (Textual)

- |                                 |
|---------------------------------|
| 1. Defect Logged                |
| 2. QA Validates Reproducibility |
| 3. Severity Assigned            |
| 4. Developer Root Cause Review  |
| 5. Triage Meeting Decision      |
| 6. Fix / Defer / Reject         |
| 7. Regression Validation        |
| 8. Closure Confirmation         |

## 10. Outcome

The triage process ensured:
-----------------------------

- |   |
|---|
| • Controlled release stability          |
| • Zero critical leakage                 |
| • Transparent stakeholder communication |
| • Risk-based delivery alignment         |