**Common Rules**

1. Equal rights: all bugs should be submitted to bug tracker with no distinction based on their priority, severity, color of their skin or gender.
2. Bug is a bug: bug describes problem in terms of symptoms (screenshots/logs) - don't make any conclusion about the cause of issue; provide your thoughts in comment section instead.
3. Raise flags: when you find something critical (a showstopper), don't just submit it to Jira - inform QA Lead / Project Manager.

**Workflow overview**

**Step 1: submit**

A reporter submits bug. Bug consists of required (e.g. Summary, Description) and optional (e.g. Attachments) attributes. Please, refer to Bug Attributes (below) for detailed information. There are also specific requirements per product, - refer to Product Specifics (below).  
A bug is assigned to the appropriate Project Manager (or Product Owner / Team Lead), - see Product Specifics section of this article to find assignee associated with a product.

**Step 2: prioritize and assign**

Prioritization is made by editing bug "Fix version" - it can be set to one of the development iterations or backlog.  
Bug is assigned to a 'resolver' (e.g. developer).  
As an option, PM can resolve bug himself (as 'won't fix' or 'not a bug').

**Step 3: resolve**

Bug can be resolved in variety of ways by developer, except as 'won't fix' or 'not a bug' (developer should assign bug back to PM in this case). See table below for list of resolutions and their verifications.

**Step 4: verify**

QA Engineer (usually a reporter) verifies resolution. In case of successful verification, tester closes it. Otherwise - reopens.

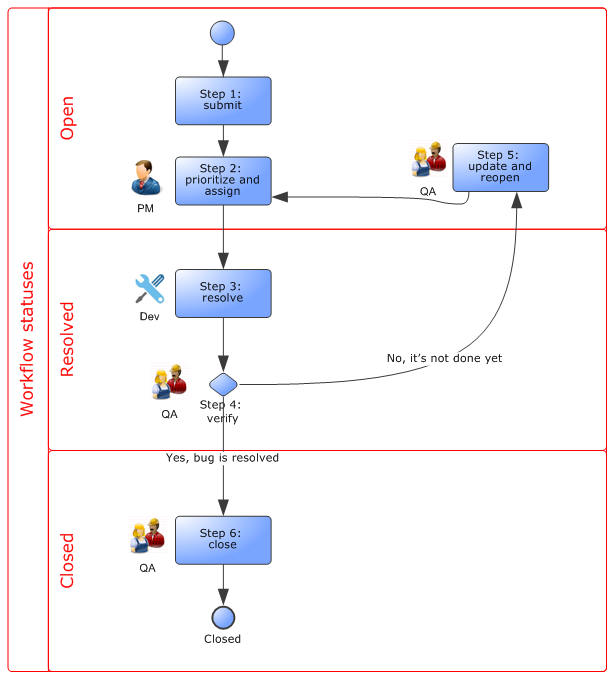
|  |  |  |
| --- | --- | --- |
| **Resolution** | **Description** | **Verification** |
| Fixed | Bug was fixed | Close if bug no longer reproduces; otherwise reopen |
| Won't fix | There's a reason not to fix the bug EVER, e.g. product is no longer supported; major changes are planned - bug won't be applicable | Make sure there's a reason not to fix it EVER; otherwise reopen, suggesting to put to backlog |
| Duplicate | The bug was already submitted | Close, updating the original bug with your steps / logs / screenshots (reopen original bug if it was closed); if you feel like your bug is a separate bug, - reopen it |
| Incomplete | Some information is missing (e.g. steps are not clear, missing logs/screenshots) | Update the bug and reopen |
| Cannot reproduce | Bug doesn't reproduce for resolver | This resolution should be processed with care.  a) If it's a random bug (repro rate is <100%), closing means it doesn't reproduce AT ALL now. E.g. you didn't reproduce it in reasonable timeframe / number of efforts.  b) If it is environment related (developer can't repro on his environment, but it does repro on QA environment), - escalate.  c) Don't close a bug, that used to reproduce, but now does not within one build. This is likely an environment/configuration issue, - escalate. |
| Not a bug | The submitted issue doesn't make sense to resolver | If it is an improvement - reopen, it should be put into backlog (note: a "bad" improvement should be resolved as "won't fix").  If it is a bug - update with more information and reopen.  If it is nor a bug, nor an improvement - why was it submitted? |

**Step 5: reopen**

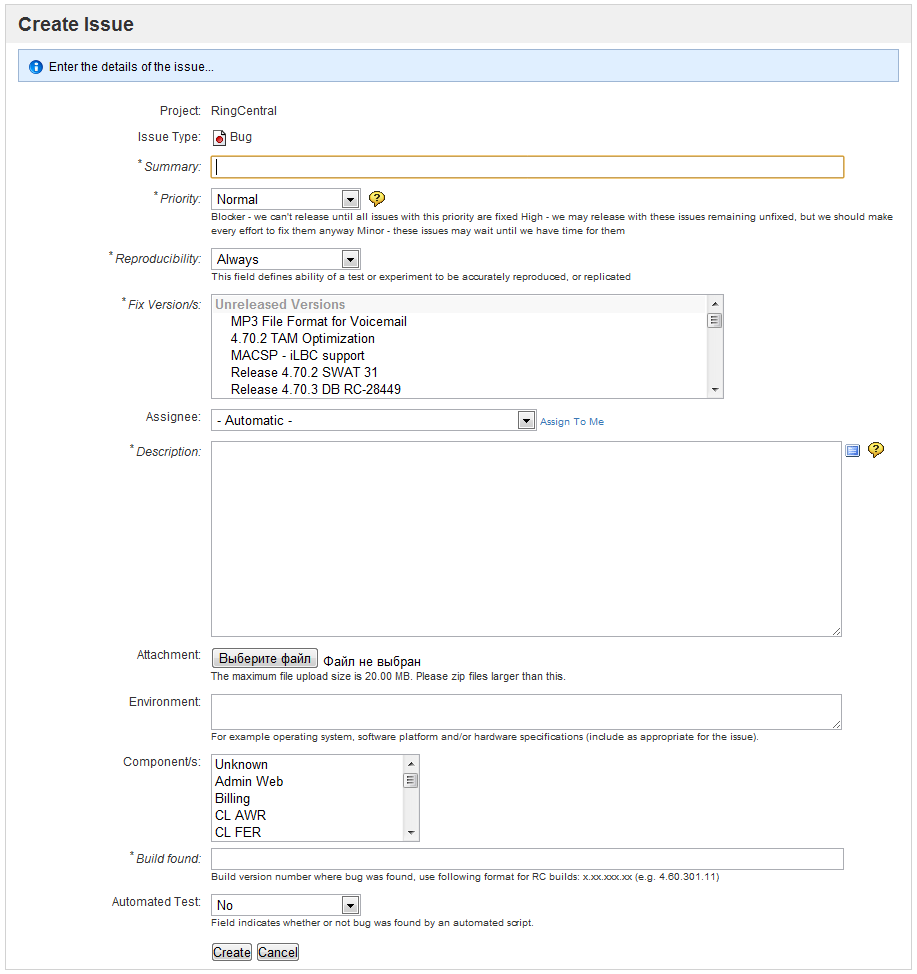
Reopening a bug generally requires it to be updated (e.g. add screenshots or specify steps to reproduce).  
Bug should be assigned to PM on reopen for another round of planning and prioritization.

**Step 6: close**

Closing a bug means it is successfully resolved.



**Bug attributes**



**Required**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Submitted by** |
| Reporter | Reported by, name | Automatically |
| Created | Date and time created, timestamp | Automatically |
| Updated | Date and time updated, timestamp | Automatically |
| Summary | Brief description of a defect, is used in quick ticket management and review | Reporter |
| Description | Detailed defect description, should include conditions, steps to reproduce, expected and actual result | Reporter |
| Priority | Is basically a defect severity  a)      High  b)      Medium  c)      Low | Initially submitted by reporter; can be modified later on the workflow |
| Reproducibility | Reproducibility of a defect  a)      Once  b)      Rarely  c)      Occasionally  d)      Often  e)      Always | Reporter (default is ‘always’) |
| Build found | Build/version, this bug is reported for (there’s can be more than 1) | Reporter |
| Fix version | Is basically a fix priority. Determines when a bug is planned to be fixed. Drop-down list. | Irina Tomilova |
| Assignee | Name of a person, assigned to process the defect | Irina Tomilova |
| Status | Is a workflow status  a)      Open  b)      Resolved  c)       Closed | Is set automatically by Jira according to workflow progression |
| Resolution | Is a resolution to the issue. Does not directly correspond to Status, but since Resolution only appears when transitioning to Resolved workflow Status, most issues with a Resolution are ‘Resolved’ or ‘Closed’. Excluding Reopened ones.  a)      Unresolved  b)      Fixed  c)      Won’t Fix  d)      Duplicate  e)      Incomplete  f)      Cannot reproduce  g)      Not a bug  h)      Reopened | A person, resolving the issue (e.g. developer) |

**Optional**

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Submitted by** |
| Link | Link to another issue | Anyone who knows that link exists (e.g. reporter, developer…) |
| Component(s) | Part of the product, a defect is found in (“SMC” by default) | Reporter |
| Attachments | Attached documents, such as log files, data dumps, etc. that can help investigate the defect | Reporter |
| Comments | Any discussion held during defect processing. | Anyone involved in defect processing |
| Original estimate | Estimation to fix the bug | Resolver (Developer) |
| Time spent | Actual time, spent fixing the bug | Resolver (Developer) |
| Why bug appeared? | A drop-down to track why bugs appear | Resolver (Developer) |

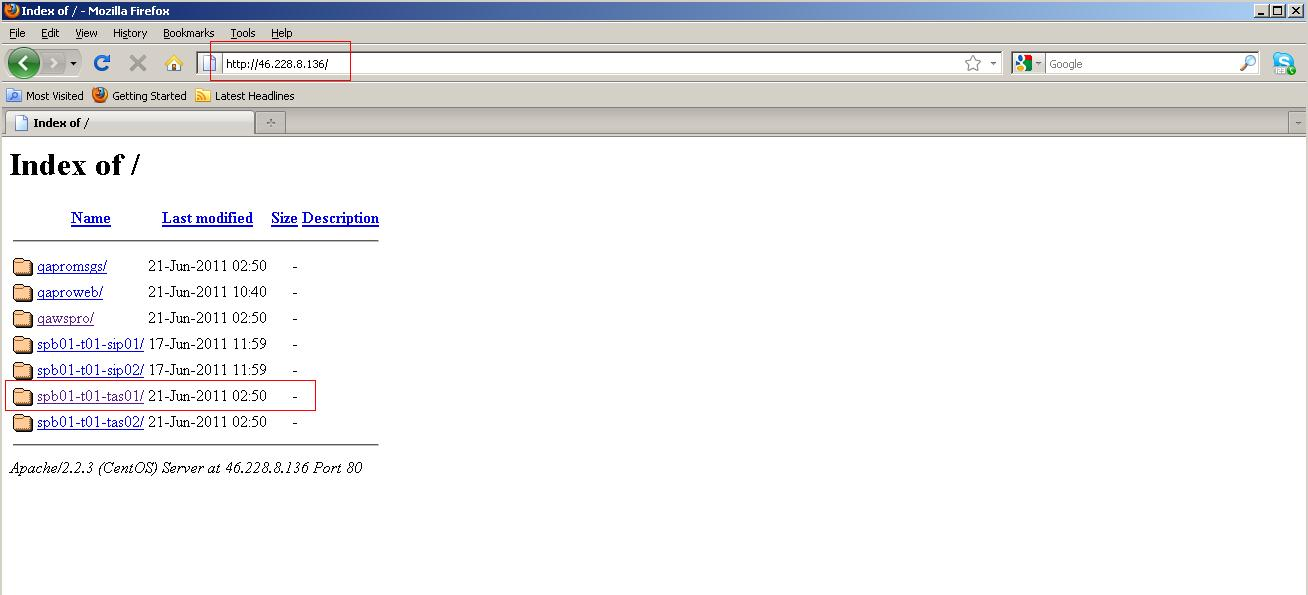
**Product Specifics**

|  |  |  |
| --- | --- | --- |
| **Project** | **Specific environment information** | **Default assignee** |
| TAS | 1. Version number 2. Test environment where bug is reproduced (DB information) 3. Account number 4. TAM, TGS and SMC logs (log files, reproduce the problem, attach file) 5. Call time (in case log is missed or contains several calls) 6. Calling number 7. Called number 8. Record from session log (if call was made not on local machine) 9. SIP traffic from both sides: client and server | Irina Tomilova |

**How to get TAM, TGS, SMC logs and SIP traffic (from QAPRO environment).**

1. Go to <http://46.228.8.136/>.
2. Enter your login/password (see AccessList.xls).
3. Select folder that contains required logs (e.g. “spb01-t01-tas01” for TAS server 01).
4. Open log and attach the part which contains information about your call(s) to your bug.

**How to get record from session log.**



1. Go to Admin Web (<http://service.qa.dins.ru:8088>)
2. Enter your login/password (see AccessList.xls)
3. Find your account in “Subscribers” (e.g. search by Phone Number)
4. Go to Session Log there
5. Select your call and paste it to bug.

