UC012: Place Product In Remark State, Determine Causes and solutions

Product Assembly, Component Assembly, Remark State, PLM



Reference: [Subject]**Document reference to be filled by Quality department if needed**

Edition/review: 1/0

Author(s): David Van Hoecke

Date: 9-3-2017

**Changelog:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Edition** | **Review** | **Date** | **Writing** | **Validation** | **Comments** |
| 1 | 0 | 20170309 | DVH |  | Issuance of the document |
|  |  |  |  |  |  |

SUMMARY

[**Place Product In Remark State**](#_jttnm128jnhh) **5**

[Description](#_aum6igmuguy8) 5

[Level](#_997t879g77q1) 5

[Trigger](#_m62qwfoi6jjw) 5

[Primary Actor](#_81q70orbm99c) 5

[Additional/Supporting Actors](#_8j656flu48ux) 5

[Stakeholders](#_sn07roitfcgi) 6

[Preconditions](#_ix4ncv6r608) 6

[Main Success Scenario](#_z9w9uig8dczg) 6

[Extensions](#_48ozvxkixxgq) 6

[Post Conditions](#_fhli1a42awos) 6

[Frequency](#_gijjxicuphem) 6

[Special Requirements](#_ylko2z2vd0xb) 7

[Issues and Next Steps](#_fv8kck4owr5m) 7

[**Determine remark causes & solutions**](#_jki4qy417adb) **7**

[Description](#_fb28ry31468y) 7

[Level](#_qygdqwrsdgnw) 7

[Trigger](#_bqm84xupw4ou) 7

[Primary Actors](#_6tjjrfw5cl97) 7

[Additional/Supporting Actors](#_gjx33q2dnemx) 8

[Stakeholders](#_zgthkcmj0git) 8

[Preconditions](#_35im265ytvbk) 8

[Main Success Scenario](#_taju66uwzlf4) 8

[Extensions](#_nb1zo36ih62l) 8

[Post Conditions](#_8jwipnjugzm3) 9

[Frequency](#_z64io79jsk1v) 9

[Special Requirements](#_lo6cnw3ge7dz) 9

[Issues and Next Steps](#_9uegdmx3ccyj) 9

UC012: Place Product In Remark State, Determine Causes and solutions

# Place Product In Remark State

## Description

Any user can place a Product Assembly in “Remark”. Additionally, devices or software applications can place a product in remark if they conclude this is necessary. Eg. a test setup may determine a product isn’t functioning properly. A user is advised to also provide a description of the remark and for each cause.

A product in “remark” is blocked and cannot proceed with its assembly / test / packaging / shipping until all remarks have been assigned one or more causes and one solution per cause. This causes the remark state to be suspended. No actions can be undertaken on the product assembly until the remark state is suspended.

Remark states provide a key insight in our production process. Apart from actually preventing a faulty product to end up at a customer it allows us to analyze, quantify and report any problems in our production. Collecting this data and reporting on it teaches us valuable lessons-learned and allows us to optimize production while at the same time increasing quality.

## Level

User goal

## Trigger

This is a non-limitative list:

* + 1. During assembly a situation arises which points out that the current product assembly is not functioning properly or may have something else wrong with it.
    2. An application determines the assembly isn’t functioning to spec and concludes it should be placed in remark
    3. A problem with a batch of components causes a range of product assemblies to be placed in remark
    4. A failed product test may cause a user to decide the product is suspect. Even without knowing conclusively that the product is indeed faulty it still needs to be placed in remark.

## Primary Actor

The PA is any user

## Additional/Supporting Actors

None

## Stakeholders

* Production
* Support
* Management

## Preconditions

The Primary Actor must have use of a workstation with an active internet connection or at least an internal access to the application. The workstation will have a browser window open, pointing to the site.

## Main Success Scenario

* A user can place a product assembly in remark by clicking the “Remark” link / button in the product assembly page.
* This will open a page that allows the user to
  + select a remark type
  + enter a description
  + enter a timestamp for this remark
* The causes and solutions can be linked to a component assembly, eg. Drive Unit DUK00001 in Wing 000100XXXX is faulty due to a supplier error
  + The referenced component is placed in remark as well
  + After the complete remark workflow is finished and the product Remark is Resolved, the referenced component remains in Remark state. In this case the faulty drive unit (=referenced component) would be replaced so the wing remark can be resolved. The faulty drive unit stays in remark state of course.
  + If the product remark can be Resolved by performing maintenance on the component then both remarks flows can be resolved. This would effectively count as two remarks.
* A newly created remark will of course always have its “Resolved” property set to “false”

## Extensions

* **RMA Handling**
  + “RMA” is a type of remark symptom
  + Cause types contain a “Needs shipping” type
  + Solution types contain a “Shipped” type

The following business rule will be added to the Remark classes “**BR\_RemarkResolutionFromRMA**”: For a remark, a user tries to set the “**Resolved” property to “true”** of Remark **of type “RMA”**

* + **Allow:** if a **cause of type “Needs Shipped” AND** corresponding **solution of type “Re-shipped”** with property **Success = “true”** are added
  + **Block:** Any other configuration

*This results in the following behavior:* a product returns to us in RMA, this means it is placed in remark. The remark type is “RMA”. After determining cause and solution to fix whatever reason caused the remark an additional cause / solution pair needs to be defined, namely a cause of type “needs shipping” and a solution of type “re-shipping”. If the “re-shipping” solution’s property **Success** is set to **True** we can close the RMA remark by setting its **Resolved** property to **True**

* **Exception:** an edit attempt is made to stale data, eg. a user has an assembly page open for an assembly that has been placed in remark by another user after the page was opened
  + user is warned that he’s using data that has been changed since he requested it
  + user retries process starting from clicking the edit button
* **Exception:** the application user attempts to change an assembly or attempts to export the assembly to a different model state

## Post Conditions

* Success End Condition
  + The product has the state “Remark” and one or more causes for this remark
  + Optionally each remark entry and cause have their own description as entered by a user
* Minimal Guarantees
  + The system will ensure that a product in remark is completely blocked from a PROD point of view. Only by resolving the remark state is the product assembly released again.
  + The system will guarantee that if a product assembly is in remark that is part of another product assembly as assembly component then that parent product assembly is also placed in remark

## Frequency

This case will occur only occasionally

## Special Requirements

* Performance:   
  Timeout values will be set so the user receives “a resource unavailable” message if this occurs
* Security  
  Only administrators have access to this page
* Usability / Accessibility  
  User must be able to view page in English, French and Spanish.
* Other  
  The UI must be able to be used over a range of platforms: PC, laptop, tablet, smart phone…

## Issues and Next Steps

none

# Determine remark causes & solutions

## **Description**

After a product is placed in remark we will also keep track of any causes and solutions for / to this remark state. A cause can be seen as an **origin or a justification** of a remark while a solution resolves a particular cause.

A solution contains a “Successful” property which signifies if the solution actually is a fix for the issue (since a solution can fail, if the proposed intervention does not work for example)

If a cause / solution combo is not successful (if the Successful property remains false) and the product assembly returns to remark, we reopen **the existing remark** instead of creating a new one

(to juxtapose last rule: If a new cause for remark arises, we create a new remark and do not reopen the previous)

Apart from being able to quantify these causes and solutions we’d also want to be able to persist costs. So we keep track of cost two factors, Labour Cost and Component Cost, which describe the amounts.

The whole remark -> cause -> solution workflow will take place over several time spots and users. A production technician for instance will typically place a product in remark while a quality technician will research causes and solutions.

## Level

User goal

## Trigger

In the product remark pages for a product in remark, the quality technician can choose to enter causes and solutions.

## Primary Actors

User and quality assurance users

## Additional/Supporting Actors

None

## Stakeholders

* Production
* Support
* Management

## Preconditions

The Primary Actor must have use of a workstation with an active internet connection or at least an internal access to the application. The workstation will have a browser window open, pointing to the site.

## Main Success Scenario

A quality technician selects to view the remark for a certain product and wants to enter causes and solutions for this remark issue.

He / she does this by “adding a cause”, a remark cause can added by filling in a form and noting down:

* a description
* a cause date
* a remark cause type
* optionally an assembly component can be used as a reference for this cause

For each cause a solution can be added at any time as well. This solution will describe how to fix / resolve the remark causes while at the same time keep track of costs and factors leading up to those costs. This is done by having the user enter:

* a description
* a solution date
* a remark solution type
* optionally an assembly component can be used as a reference for this solution
* A list of cost factors and their cost
  + cost: numeric value
  + cost factor: a description of that value

## **Extensions**

* **Exception:** A user wishes to create a cause and / or a solution with a reference but chooses the wrong reference. Eg. user tries to create a solution that references a component in remark

## Post Conditions

* Success End Condition
  + The product has the state “Remark”, one or more causes for this remark and potentially (hopefully) some solutions as well
  + Each remark cause and solution entry and cause have their own description as entered by a user
  + causes and solutions can reference existing assemblies
  + costs and cost contributing factors are persisted for solutions
* Minimal Guarantees
  + Creating causes and solutions is the first step of taking an assembly out of remark again. Adding them guarantees that an assembly is in the correct state to be taken out of remark manually

## Frequency

This should occurs for each product in remark. Even for those assemblies that need to be scrapped

## Special Requirements

* Performance:   
  Timeout values will be set so the user receives “a resource unavailable” message if this occurs
* Security  
  Only QA role personnel can create remark cause and solutions
* Usability / Accessibility  
  User must be able to view page in English, French and Spanish.
* Other  
  The UI must be able to be used over a range of platforms: PC, laptop, tablet, smart phone…

## Issues and Next Steps

none