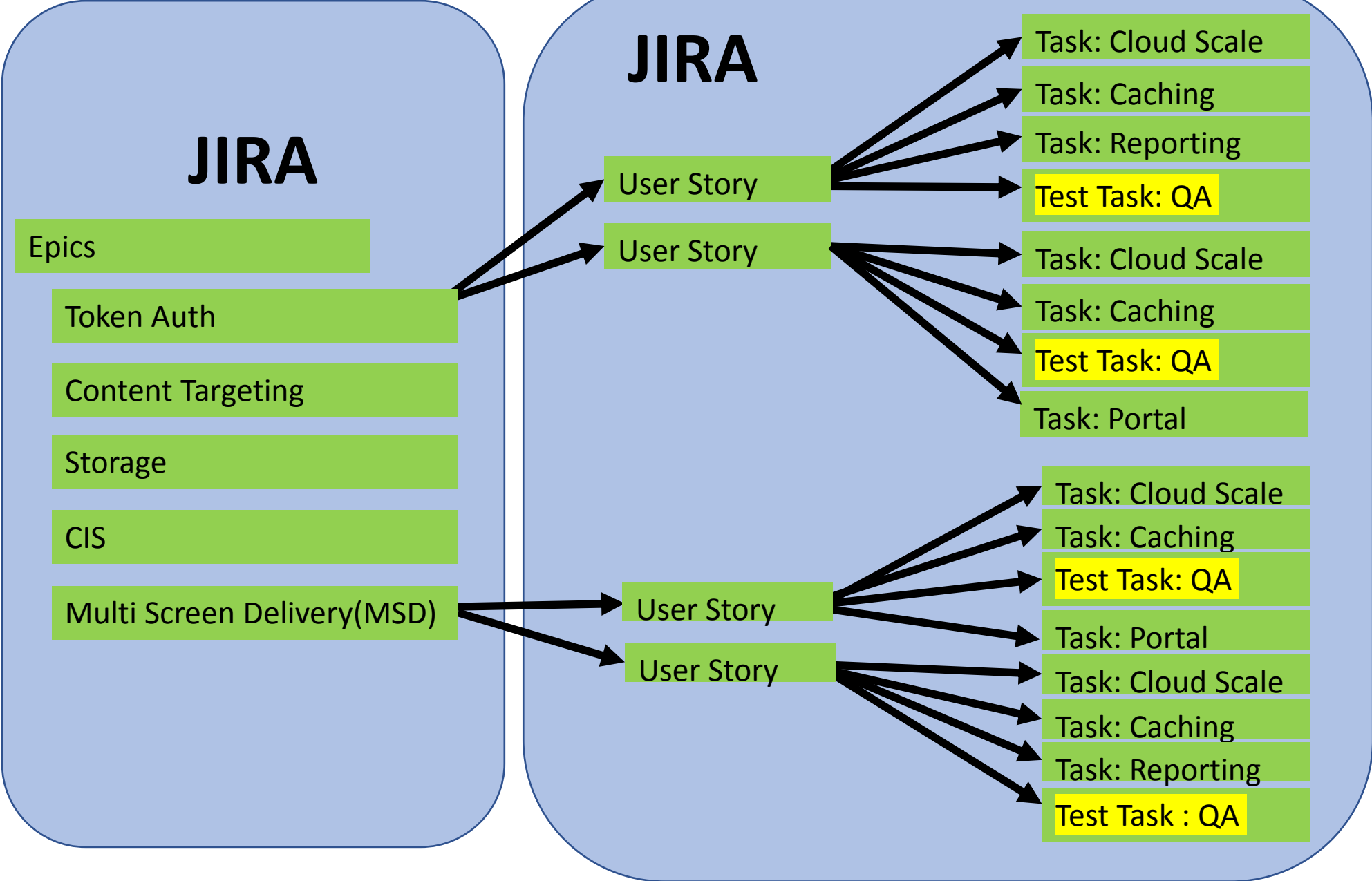


UDN SOFTWARE ENGINEERING PROCESS (SEP)



Phase	Activity	Owner	Tool	Comments
EPIC / User Story Creation	Create EPICs / User Stories	Product Management	JIRA	
In Triage	All bugs and tasks start here	Development Manager / QA Manager	JIRA	
To Do	Items have an estimate and are assigned to a sprint. They may optionally have an assignee	Development Managers	JIRA	
In Progress	Developers are actively working on tasks. Coding, functional & unit tests	Development team	JIRA	
In Review	Should be used when a completed item is under peer review	Documentation team	JIRA	
Ready for Test	Development is complete & ready for QA validation	Development team	JIRA, Confluence	
In Test	QA is actively testing	QA team	JIRA, Confluence	
Done	Everything is completed & passed QA	QA, Development	JIRA, Confluence	

Release UDN x.y



WORKFLOW (EPICS, STORIES, TASKS – DEV & TEST) - “NEW SIMPLIFIED WORKFLOW”



Must updates (Tasks / Sub Tasks)

- In Triage -> To Do:
 - Fix Version
 - Original Estimate
- In Progress -> In Review
- In Progress -> Ready For Test
 - Log Effort
 - Resolution

Must updates (Stories)

- In Triage -> To Do:
 - Fix Version
 - Release Theme
 - Executive Summary

Must updates (EPICs)

- In Triage -> To Do:
 - Fix Version
 - Release Theme

WORKFLOW (BUG) – “NEW SIMPLIFIED WORKFLOW - BUG”



Must updates

In Triage:

{ Affects Version
Bug Origin Environment

In Triage -> To Do:

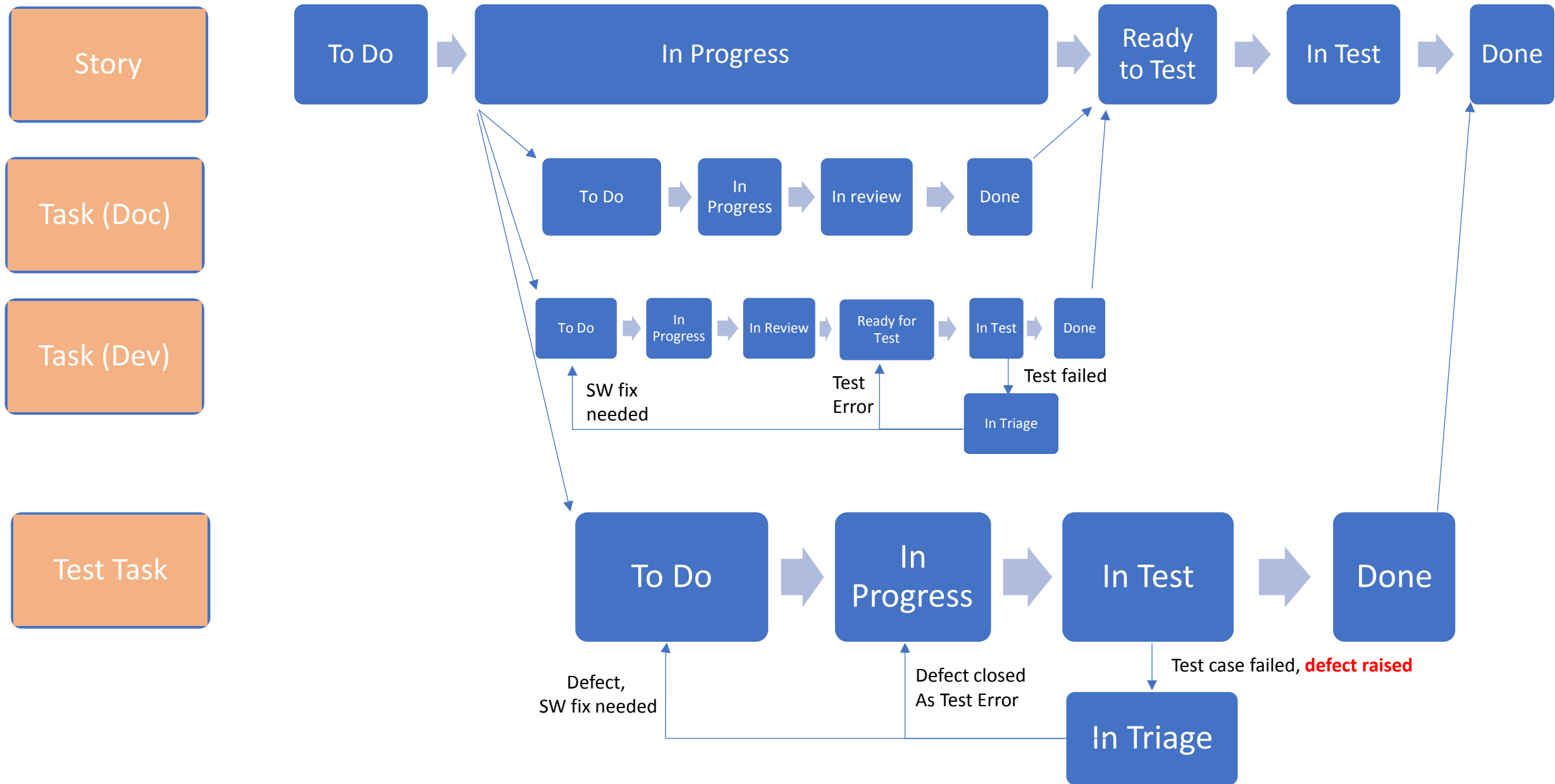
{ Fix Version
Original Estimate

In Progress -> In Review

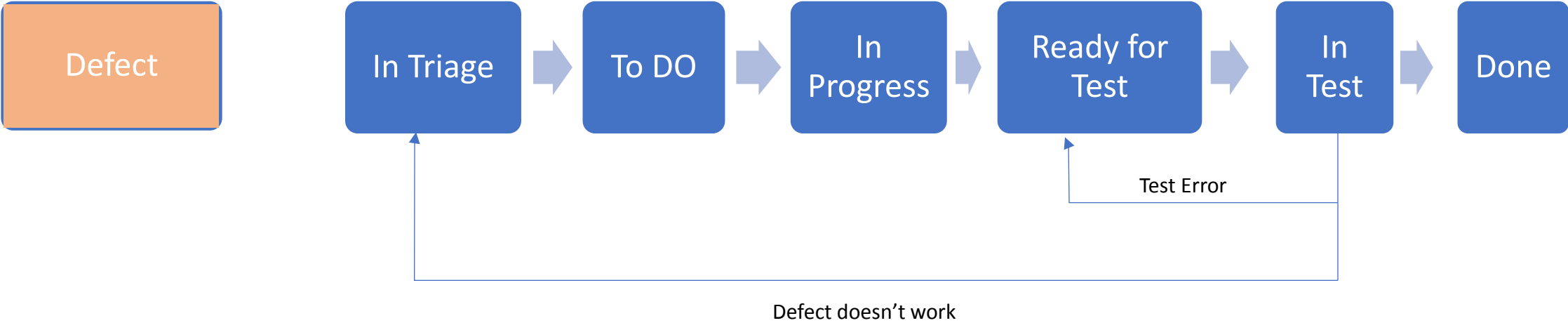
In Progress -> Ready For Test

} Log Effort
Resolution

JIRA WORKFLOW



JIRA WORKFLOW (DEFECT)



BUG ORIGIN ENVIRONMENT

At the time a bug is created the user must select the environment the bug was discovered (group they belong to). The available options are:

- Production (Support, SE, Operations)
- QA
- Staging
- Dev (Anyone not in the other groups)

Create Issue

Project *

 RAMESH TEST PROJECT (...)

Issue Type *

 Bug ?

Bug Origin
Environment *

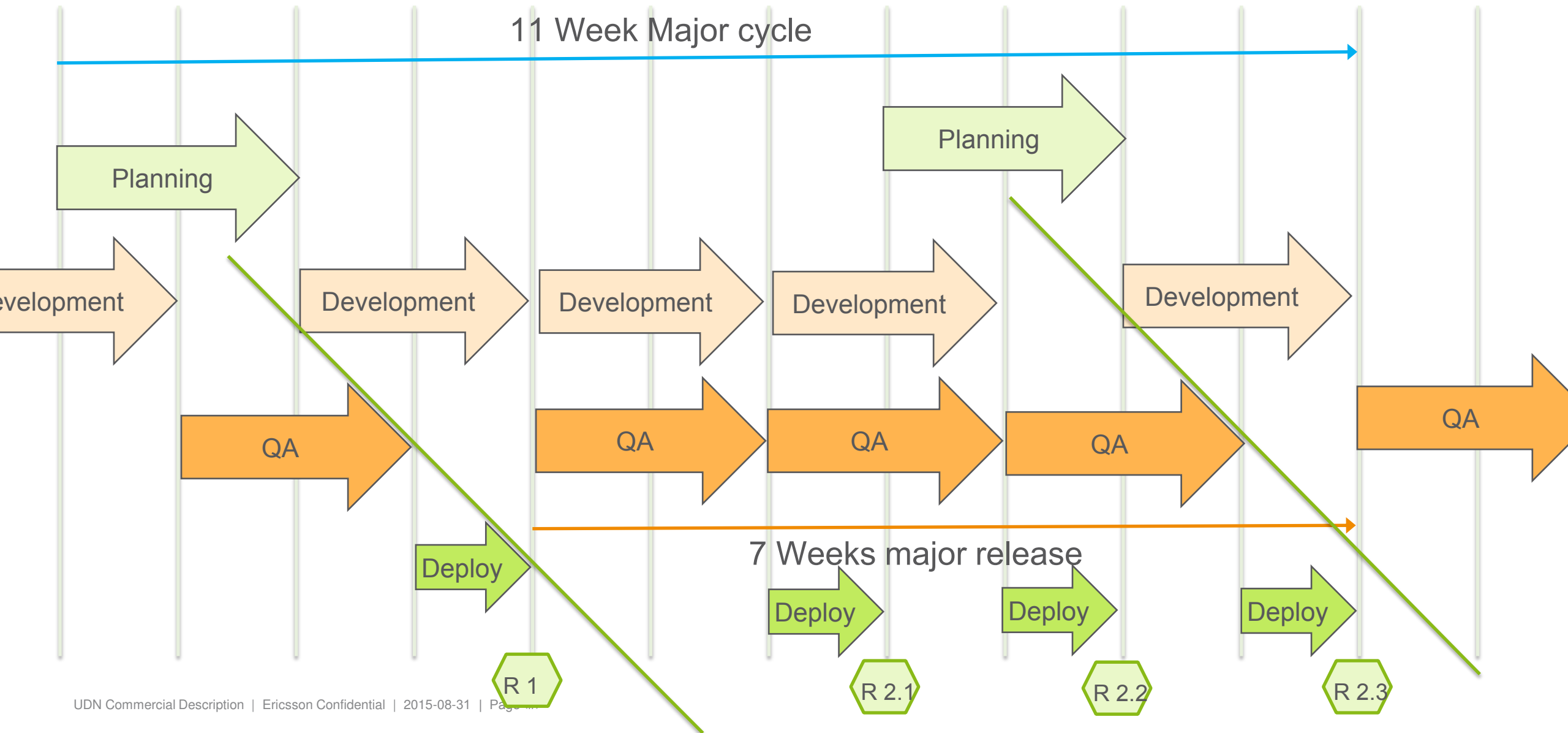
Summary *

Assignee

- ✓ None
- Production
- QA
- Staging
- Dev

Automatic

RELEASE Schedule



UDN Release Deliverables

Deliverables	
Sprint Planning – User Stories	<ol style="list-style-type: none">1. User Stories moved to JIRA and discussion with Development aligns with “TO DO” stage in JIRA
Sprint Planning – Development, Documentation & QA	<ol style="list-style-type: none">1. HLA / HLD stored in Confluence – linked to JIRA Epic &/or User Story2. Dev Tasks created with stage set "TO DO" against all User Stories in JIRA with estimates3. User Stories status set to “Scheduled” in JIRA for items that are confirmed to be in the release4. Test Plan stored in Confluence linked to JIRA5. Test Tasks created status set to “TO DO” against all User Stories in JIRA with estimates6. Start documentation planning process
Sprinting – Development	<ol style="list-style-type: none">1. Develop the code to the user story / HLD’s specifications2. Unit & Function tests created3. All Dev and Documentation Tasks status set to “DONE” against all User Stories in JIRA4. Create Documentation Drafts5. Complete Documentation Reviews & Finalize Documentation
QA	<ol style="list-style-type: none">1. Test Report2. Release Notes3. All Test Tasks status set to “DONE” against all User Stories in JIRA4. All “BUGS” are set to “DONE” against all User Stories in JIRA

UDN Release Schedule 2017 H2

Week start date (MM/DD)	7/3	7/10	7/17	7/24	7/31	8/7	8/14	8/21	8/28	9/4	9/11	9/18	9/25	10/2	10/9	10/16	10/23	10/30	11/6	11/13	11/20	11/27	12/4	12/11	12/18	12/25	1/1	1/8
Week END date (MM/DD)	7/7	7/14	7/21	7/28	8/4	8/11	8/18	8/25	9/1	9/8	9/15	9/22	9/29	10/6	10/13	10/20	10/27	11/3	11/10	11/17	11/24	12/1	12/8	12/15	12/22	12/29	1/5	1/12
Sprint Planning - User Stories		2.2-2.4						2.5-2.7												2.8-2.9								
Sprint Planning - Development		2.2-2.4											2.5-2.7							2.8-2.9								
Dev Sprint	2.1		2.2					2.3				2.4			2.5		2.6		2.7			2.8		2.9				
Quality Assurance (QA)		2.1						2.2		2.3		2.4			2.5		2.6		2.7			2.8		2.9				
Deployment to Staging & Production						2.1					2.2		2.3		2.4			2.5			2.6	2.7						2.8
UDN Release Date						8/10					9/14		10/5		10/19				11/9			11/30	12/7					1/11

UDN 2.x Release Schedule 2017

UDN Release Schedule	UDN 2.1		UDN 2.2		UDN 2.3		UDN 2.4	
	Start	End	Start	End	Start	End	Start	End
Sprint Planning User Stories	5/30	6/2	7/10	7/14	7/10	7/14	7/10	7/14
Sprint Planning – Dev	6/5	6/9	7/17	7/21	7/17	7/21	7/17	7/21
Sprinting - Development	6/12	7/14	7/24	8/11	8/14	9/15	9/18	9/29
QA execution	7/17	8/4	8/14	9/7	9/11	9/29	10/2	10/13
Pre-deployment meeting with Operations	8/2	8/2	9/6	9/6	9/27	9/27	10/11	10/11
Deployment to Staging	8/7	8/9	9/11	9/13	10/2	10/4	10/16	10/18
Deployment to Production	8/10	8/10	9/14	9/14	10/5	10/5	10/19	10/19

UDN 2.x Release Schedule 2017

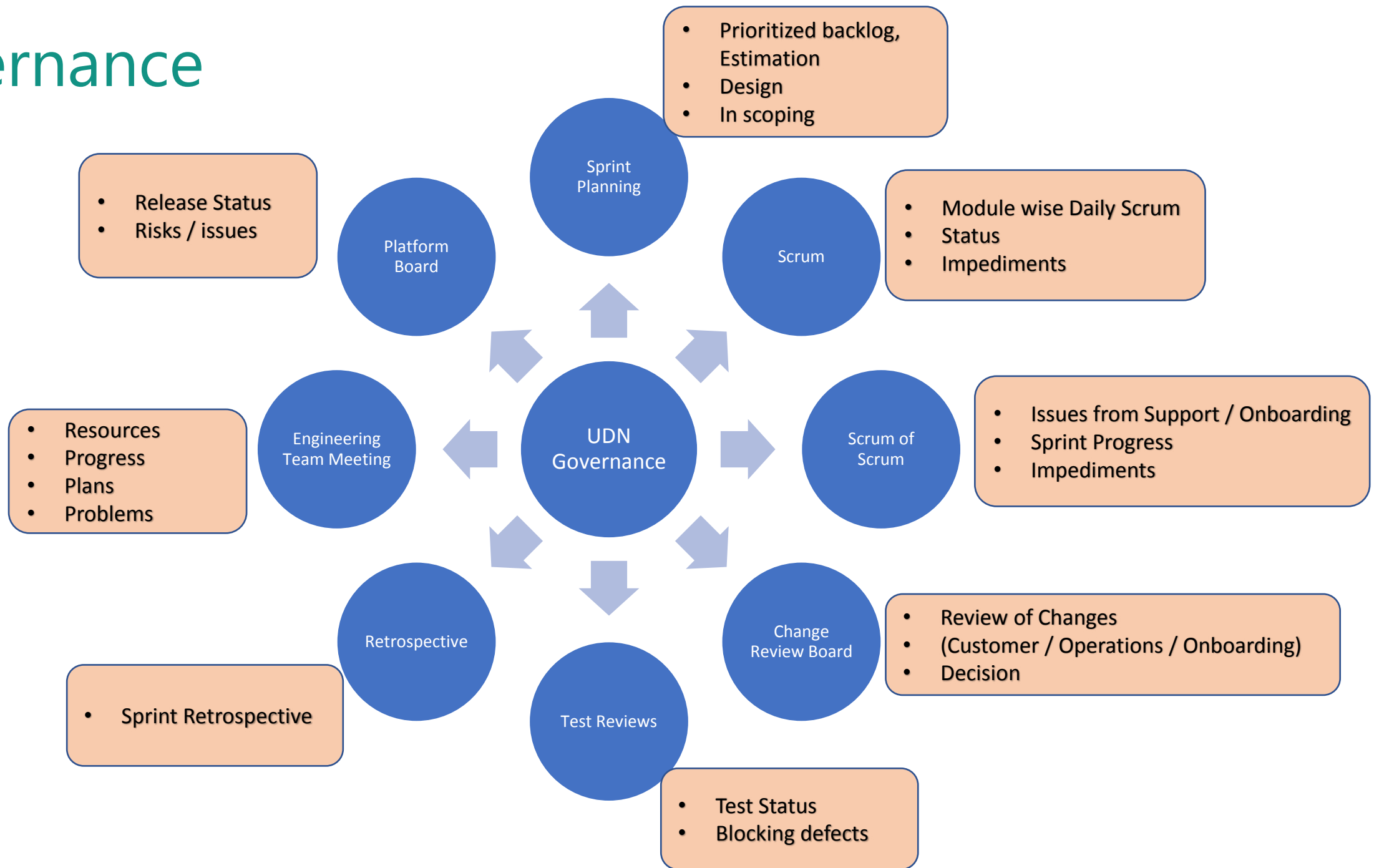
UDN Release Schedule	UDN 2.5		UDN 2.6		UDN 2.7		UDN 2.8	
	Start	End	Start	End	Start	End	Start	End
Sprint Planning User Stories	8/21	8/25	8/21	8/25	8/21	8/25	11/13	11/17
Sprint Planning – Dev	10/2	10/6	10/2	10/6	10/2	10/6	11/20	11/24
Sprinting - Development	10/9	10/20	10/23	11/3	11/6	11/17	11/27	12/8
QA execution	10/23	11/3	11/6	11/17	11/20	12/1	12/11	12/22
Pre-deployment meeting with Operations	11/1	11/1	11/15	11/15	11/29	11/29	12/20	12/20
Deployment to Staging	11/6	11/8	11/27	11/29	12/4	12/6	1/8	1/10
Deployment to Production	11/9	11/9	11/30	11/30	12/7	12/7	1/11	1/11

RELEASE

- Code
 - Stored @ R & D Montreal Hub (/proj/cino)
 - Based on 2.0.0, pending modules : Analytics
- Documentation
 - [Release notes & User Guides](#)
 - [Test Report](#)



Governance



DASHBOARD

- Dev Dashboard @ <https://dashboard.ericssonudn.com>

UDN DEVELOPMENT METRICS

On Time Delivery	
Delay = Actual release date - Planned release date	
Status Indicator	Delay (Calendar days)
Commitment	<4
Robust	4-7
Warning	8-10
Off-track	>10

Fault Slip Through (FTS) to Production	
FTS = (Faults found in production / Faults found in Dev, QA, staging and Production)	
Status Indicator	FTS (%)
Commitment	<6
Robust	6-10
Warning	11-15
Off-track	> 15

Quality	
Status Indicator	Measurement Criteria at Release
Commitment	<u>New functionality</u> <ul style="list-style-type: none">Zero open Blocking defects (JIRA)Zero open Critical defects (JIRA)100% Test completion <u>Legacy functionality</u> <ul style="list-style-type: none">Zero open Regression defects
	<u>New functionality</u> <ul style="list-style-type: none">Zero Blocking defects (JIRA)Zero Critical defects (JIRA) / Open critical defect has acceptable remedy from Support100% Test completion <u>Legacy functionality</u> <ul style="list-style-type: none">Regression defects exist which are “Minor” in nature from customer perspective
Off-track	<u>New functionality</u> <ul style="list-style-type: none">Open Blocking defects that do not have remedyOpen critical defects that do not have remedyTest completion is not 100% <u>Legacy functionality</u> <ul style="list-style-type: none">Regression defects exist which are “Major” in nature from customer perspective



ERICSSON