

10 things every CSV
URS should consider



Importance

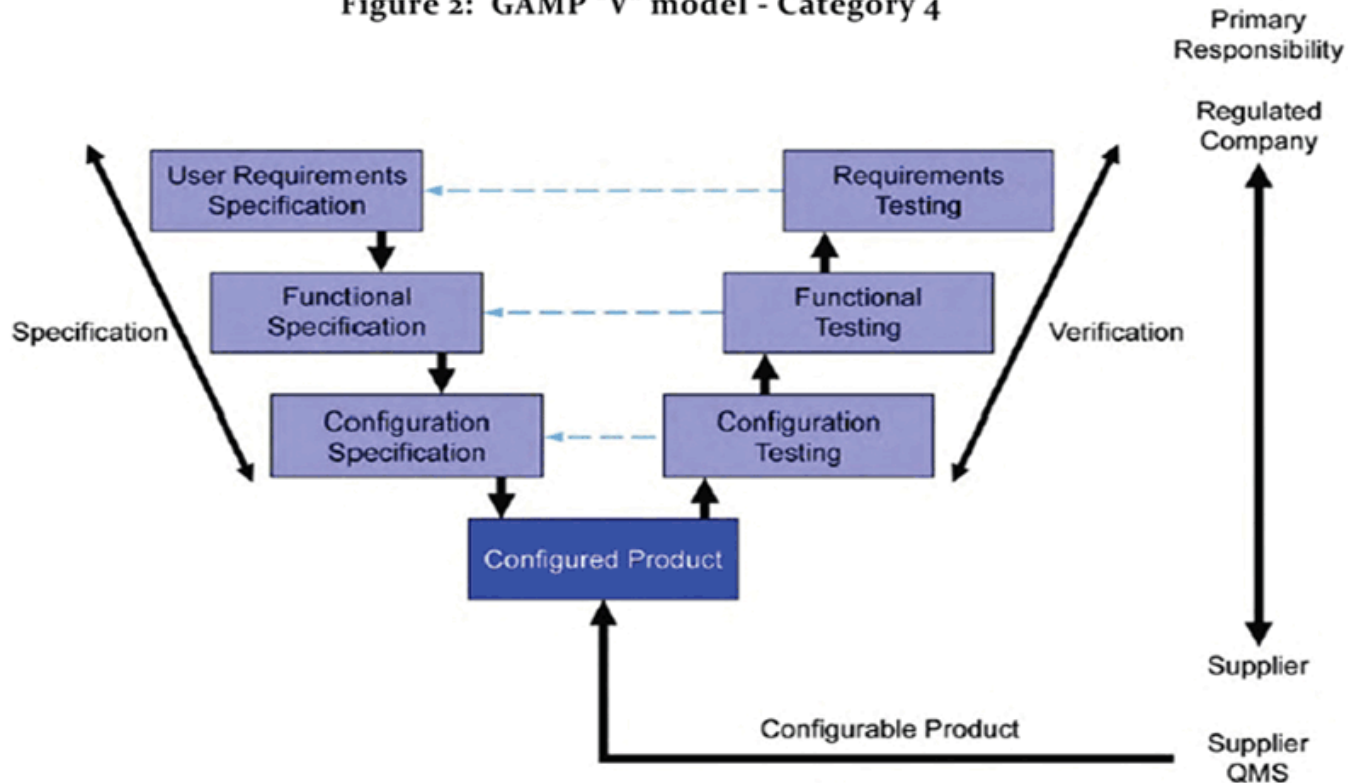
The requirements should define clearly what the system should do and state any constraints.

- Traceability throughout the product lifecycle
- Build data integrity into your design
- Can assist in supplier selection
- The starting point of a project.



URS - the starting point of a project

Figure 2: GAMP "V" model - Category 4



GAMP 5 A Risk-Based Approach to Compliant GXP Computerised Systems. Copyright ISPE 2008. All rights reserved

The 10 things

No.	Area
1	Author
2	Review/Approve
3	S.M.A.R.T
4	Prioritise
5	User
6	Future
7	Availability
8	Size
9	TBD
10	Duplicates



1. Who writes the URS?

The regulated company should always write the URS.

Avoid delegating ownership to an external party. This may result in requirements that don't satisfy the business' needs.

Subject Matter Expert involvement is crucial.



2. Review and approval of URS

Requirements should be reviewed and approved by the stakeholders and the subject matter experts.

Important that all stakeholders understand and agree on the business operational needs.

3. Requirements should be S.M.A.R.T



3a. Requirements should be **SPECIFIC**

Requirements should be documented in a clear concise manner understandable to staff and more importantly vendors/suppliers.

Non-generic and should not be open to interpretation.



3b. Requirements should be **MEASURABLE**

Requirements should be able to be verified as complete.

Should avoid non-measurable words such as “some” or “best” or “optimal”.



3c. Requirements should be **ACHIEVABLE**

Requirements should be realistically achievable in the intended environment.

Understanding of environment limitations.

Realistic to achieve within project constraints.



3d. Requirements should be **REALISTIC**

Requirements should be realistic and relevant to the solution.

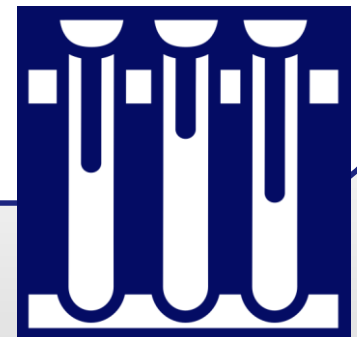
Appropriate and in context with other requirements.



3e. Requirements should be **TESTABLE**

Requirements should be written in such a way that they can be tested.

Good requirements can be traceable through the lifecycle to testing stage.



4. Prioritise your requirements



Mandatory



Beneficial



Nice to have

5. Know your End Users

System should meet the end users skillset.



6. Plan for the future

Specifying future expansion requirements can pay dividends down the line.

How much additional staff will use the system?

Could other departments or sites use the system over time?



7. System Availability

How long must the system be available?

Business hours Monday to Friday; 24/7

Consider downtime for maintenance.



8. Size matters

Avoid specifying requirements that are too long or contain many parts. This may become complex.

Consider splitting into simpler requirements.



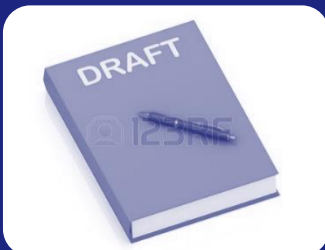
9. To be determined...



Not all requirements can be fully defined up front.



Some requirements will require further development in subsequent phases of the project.



Should be noted in initial version of URS.

10. Beware of duplicates

Increased risk in large specifications or multiple URS documents.

Also avoid contradicting requirements.



URS – Common pitfalls



Ambiguous requirements that can be misunderstood



Not all business stakeholders involved in requirements captured



Functionality specified that won't be used



Scope creep

Thank you for your time.

Grant South

Senior Consultant

grant.south@pharmout.net

www.pharmout.net