



Office of the Government Chief Information Officer
The Government of the Hong Kong Special Administrative Region

**PRACTICE GUIDE
FOR
SCOPING & PLANNING OF LARGE-SCALE
IT SYSTEM DEVELOPMENT PROJECTS
APPENDIX A
TOOLS, TEMPLATES AND CHECKLIST**

[G63a]

Version: 1.0

December 2016

© The Government of the Hong Kong Special Administrative Region
The contents of this document remain the property of the Office of the Government Chief Information Officer, and may not be reproduced in whole or in part without the expressed permission of the Office of the Government Chief Information Officer

Amendment History				
Change Number	Revision Description	Section Affected	Rev. Number	Date

Table of Contents

1	RACI MODEL.....	1
2	COMPLEXITY ASSESSMENT CHECKLIST.....	3
3	SCOPE MANAGEMENT LIFECYCLE	6
4	SCOPE STATEMENT TEMPLATE	7
5	STAKEHOLDERS QUESTIONS SET	8
6	USAGE OF ENTERPRISE ARCHITECTURE DOMAINS TO ELICIT REQUIREMENTS	10
7	REQUIREMENTS CATALOGUE TEMPLATE.....	12
8	REQUIREMENT ATTRIBUTES LIST TEMPLATE	14
9	MOSCOW PRIORITISATION	17
10	REQUIREMENTS VS. SUB-PROJECTS MAPPING TEMPLATE.....	18
11	SUB-PROJECTS PRIORITISATION ATTRIBUTE LIST TEMPLATE	19
12	PROGRAM EVALUATION AND REVIEW TECHNIQUE	21
13	DEPLOYMENT APPROACHES	22
14	PILOT PROJECTS.....	26
15	IMPLEMENTATION COMPLEXITY MATRIX.....	28
16	OTHER CONSIDERATIONS IN HIGH-LEVEL ESTIMATION.....	33

List of Figures & Tables

Figure 1 - RACI Model	2
Figure 2 - Benefit vs. Complexity Illustration	20
Figure 3 - Deployment Approach Applicability Table	25
Table 1 - Stakeholders Questions Set	8
Table 2 - Enterprise Architecture Domains to Guide Requirements Elicitation	10
Table 3 - Other Considerations	10
Table 4 - Deployment Approaches.....	22
Table 5 - Key Benefits of Pilot Projects	26
Table 6 - Key Considerations of Pilot Projects.....	26
Table 7 - Other Considerations in High-level Estimation.....	33

1 RACI MODEL

(Reference Chapter of the Guide – 1.4 Processes Overview)

- (a) The following illustrates the RACI model by project roles and key processes.

Role	Responsibility
Responsible (“R”)	Has the duty and obligation to complete the activity
Accountable (“A”)	Has the authority as the ultimate decision maker
Consult (“C”)	Has the expertise, experience, and interests who must be given the opportunity to influence decisions prior to finalisation by the “accountable” or “responsible” party
Informed (“I”)	Keeps up-to-date about the activity progress, usually upon completion of the activity

- (b) Note: Responsibilities marked in parenthesis indicate a particular role's potential involvement in the cases where approval is escalated and required by senior levels project organisation members, e.g. a high impact risk that requires review and approval of the PSC.

Processes	Project Owner/Delegate	PSC	PAT	IPM	Project Team Members	Sub-Project Team Members	Sub-Project Alignment Team	Business Analyst (if available)	Users	Other Stakeholders
Initiate Phase										
2.2 Assess Project Complexity										
2.2.1 Conduct Complexity Assessment	A			R				C	C	C
2.3 Identify and Document Stakeholders' Involvement										
2.3.1 Conduct Preliminary Stakeholder Analysis	AC			R					C	C
2.3.2 Define Roles and Responsibilities	AR			R				C	C	C
2.3.3 Ensure Business Stakeholders' Early Involvement	AC			R				C	C	C
2.3.4 Identify and Involve External Stakeholders	AC		C	R				C		C
2.4 Prepare Funding Application										
2.4.1 Define Project Scope	AC			R				C	C	C
2.4.2 Elicit High-Level Requirements	AC			R				R	C	C
2.4.3 Conduct Decomposition and Sub-Project Grouping	AC			R				C		
2.4.4 Prioritise Sub-Projects	AC			R				R		
2.4.5 Develop High-Level Project Schedule	AC		C	R				C	C	C
2.4.6 Sourcing Strategy	AC		C	R				C	C	C
2.4.7 Estimate Project Budget	AC		C	R				R	C	C
Plan Phase										
3.2 Refine Preliminary Estimates and Findings										
3.2.1 Refine Project Scope	(A)C	AC	C	R				R	C	C
3.2.2 Refine Sub-Project Grouping	(A)C	AC	C	R				R		
3.2.3 Refine Project Effort and Resource Estimation	C	C	C	AR				R	C	C
3.2.4 Refine Project Schedule	C	C	C	AR				R	C	C
3.2.5 Refine Project Budget	C	C	C	AR						
3.3 Prepare Project Management Plan										
3.3.1 Define Issue Management and Escalation Process	C	AC	C	R						
3.3.2 Define Communications Management Process	C	AC	C	R				C	C	C

Figure 1 - RACI Model

2 COMPLEXITY ASSESSMENT CHECKLIST

(Reference Chapter of the Guide – 2.2 Assess Project Complexity)

The following checklist helps Project Planner to assess the complexity of the projects.

Please tick the checkboxes in the items that are considered to be complex.

☑ *Tick the checkboxes if applicable*

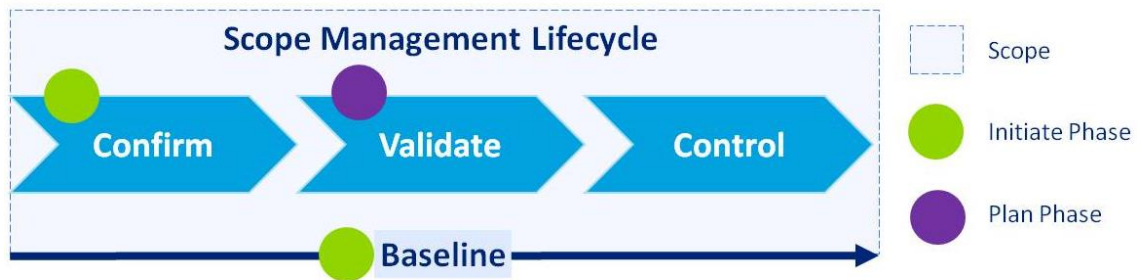
Assessment Area	High-level Considerations	Checkbox
1. Multiple Stakeholders	(a) There are a large number of stakeholders involved, relative to the capacity being handled in previous projects.	<input type="checkbox"/>
	(b) There is a large number of inter-related or cross departmental business processes and extent of system integration.	<input type="checkbox"/>
	(c) There is a high degree of impact and effect generated by the project on the users and stakeholders.	<input type="checkbox"/>
	(d) It is difficult to compromise the extent of operational involvement and project benefits of individual stakeholders.	<input type="checkbox"/>
	(e) No or little experience of the stakeholder(s) in respect of working relationship or senior management support, and in delivering IT projects.	<input type="checkbox"/>
	(f) The availability and maturity of BAs in discharging the BA roles and responsibilities may not be sufficient.	<input type="checkbox"/>
2. Major Business Transformation	(a) There is a large number of business processes and extent of system integration.	<input type="checkbox"/>
	(b) The breadth and depth of changes and the time and effort required for the users and public to adapt to the changes.	<input type="checkbox"/>
3. Technology Risk	(a) There is a lack of market supply, user groups and support expertise available.	<input type="checkbox"/>
	(b) There is incompatibility or non-compliance with the existing IT architecture, central infrastructural facilities and government guidelines.	<input type="checkbox"/>
	(c) There is a lack of proven successful cases.	<input type="checkbox"/>
	(d) There is absence of alternative or substitutive products in case the maintenance and support services are discontinued.	<input type="checkbox"/>

Assessment Area	High-level Considerations	Checkbox
4. Political, Public and Media Impact	(a) There exists political and public sensitivity with regard to the subject area based on previous project experience or political environment.	<input type="checkbox"/>
	(b) There are controversial opinions in the society and a high level of media attention.	<input type="checkbox"/>
	(c) There are policy commitments made by Government officials to the public.	<input type="checkbox"/>
	(d) There is a high level of impact of commitment on people's livelihood.	<input type="checkbox"/>
5. Policy and Legislation Changes	(a) There are high complexities of the new ordinance and its sub-ordinances and the effect to project progress and outcome.	<input type="checkbox"/>
	(b) Internal Government regulation changes such as changes in Store and Procurement Regulations.	<input type="checkbox"/>
	(c) There are controversial views of the policy in the society.	<input type="checkbox"/>
	(d) The scheduled duration on steps of the legislation process, possible delays, and how they will impact the project.	<input type="checkbox"/>
6. Delivery Timeframe	(a) Length of the project timeline, in particular when it will span over multiple years, during which situation may change.	<input type="checkbox"/>
	(b) Inadequate project timeline for the expected scope, based on previous project experience or benchmarking similar projects.	<input type="checkbox"/>
	(c) Inadequate considerations and time allocation for activities on the critical path, including funding application, procurement, user acceptance, and adoption activities.	<input type="checkbox"/>
7. User Population	(a) There are a large number of users impacted by the project.	<input type="checkbox"/>
	(b) There is a broad diversity of user groups impacted by the project.	<input type="checkbox"/>
	(c) The impact to general public, external and internal users, and their expectation of the project.	<input type="checkbox"/>
	(d) The users being in different geographical locations.	<input type="checkbox"/>

Note: This checklist serves the purpose of ensuring that all complexity assessment areas have been considered. Project Planner should pay special attention to the areas which are of high complexity and take appropriate measures to mitigate project risks.

3 SCOPE MANAGEMENT LIFECYCLE

(Reference Chapter of the Guide – 2.4.1 Define Project Scope)



- (a) **Baseline:** The scope defines the boundaries of the project, i.e., what elements are included in the project. While setting the baseline of the project, Project Planner should consider the project goals and objectives, requirements, schedule, deliverables, acceptance criteria, assumptions and constraints. These baseline considerations are further elaborated in *Appendix A – 4 Scope Statement Template* below to assist B/Ds in defining their scope statement.
- (b) **Confirm Scope:** This activity should be completed by the Project Planner during the initiate phase. The objective is to clearly state the scope of the project in writing.
- (c) **Validate Scope:** This activity should be conducted throughout the project lifecycle to ensure the scope remains unchanged when project deliverables are completed.
- (d) **Control Scope:** The objective of scope control is to manage the project activities/decisions, e.g., schedule, objective, functionality, deliverables or budget, which may alter the scope. The Internal PM should bear in mind that scope control should be conducted during the project execute phase.

4 SCOPE STATEMENT TEMPLATE

(Reference Chapter of the Guide – 2.4.1 Define Project Scope)

The attached template can be used for scope statement, covering the baselines below:

Baseline	Description
1. Project Description	<i>< Describe what the project will accomplish ></i>
2. Project Objectives	<i>< Describe what the project will achieve ></i>
3. Project Acceptance Criteria	<i>< Describe what requirements must be met in order for the project to be accepted as complete ></i>
4. Project Requirements	<i>< Identify the high-level, key requirements of the project deliverables (this should not be a detailed list of system requirements) ></i>
5. Project Assumptions	<i>< Describe the list of assumptions the project team and stakeholders are working under to complete the project ></i>
6. Project Constraints	<i>< List out the limits on resources for time, money, and manpower ></i>
7. Project Boundaries	<i>< List out what are included and excluded in the project ></i>
8. Project Deliverables	<i>< List out the high-level, key output of the project ></i>

5 STAKEHOLDERS QUESTIONS SET

(Reference Chapter of the Guide – 2.4.2 Elicit High-Level Requirements)

The Stakeholders Questions Set can be leveraged to help Project Planner to develop their own question lists to be used during consultations/workshops.

Table 1 - Stakeholders Questions Set

Stakeholder	Key Questions
1. User Representatives	<ul style="list-style-type: none"> (a) Are there any aspects of the project that you feel will be particularly valuable to you? (b) Are there any requirements that you feel should be fulfilled before others due to constraints or priorities? (c) Do you have any concerns regarding the project? (d) Who else do you feel would be beneficial to talk to about the project?
2. Management of Users	<ul style="list-style-type: none"> (a) Are you able to provide resources to the project? (b) What percentage of time will these users be available to assist with the project? (c) Are there any constraints or other priorities that may compete for users' time? (d) What features of the project do you feel will be most beneficial to you and your staff? (e) What areas of the project do you feel have the most risk?
3. Project Owner	<ul style="list-style-type: none"> (a) What are the features from this project that are essential for project success? (b) Are there any deadlines that must be met, due to constraints, laws or other mandatory requirements? (c) Have you aligned the project timeline with any other goals or projects that may be occurring simultaneously within the B/D?
4. IT Management	<ul style="list-style-type: none"> (a) Are there any technical dependencies that need to be considered for this project? (b) Are there IT initiatives underway to which the project needs to conform? (c) Do you have any documentation or mapping of current systems and interfaces that are impacted by the project?
5. IT Contractors (if necessary)	<ul style="list-style-type: none"> (a) Same questions as asked to IT Management

Stakeholder	Key Questions
6. Other B/Ds	<ul style="list-style-type: none">(a) What impacts do you foresee on your B/D from this project?(b) What are the key benefits from this project for your B/D?(c) What do you think are the major risks for your B/D from this project?(d) What (if any) business critical areas in your B/D do this project impact?(e) Are there any constraints from your B/D surrounding this project? (e.g., timeline, resources, etc.)
7. External Stakeholders	<ul style="list-style-type: none">(a) What are your expectations of this project?(b) From your perspective, what are the major benefits of the project?(c) What do you consider to be the major risks of this project?(d) Are there any sensitive or politically controversial areas of the project?

6 USAGE OF ENTERPRISE ARCHITECTURE DOMAINS TO ELICIT REQUIREMENTS

(Reference Chapter of the Guide – 2.4.2 Elicit High-Level Requirements)

The following table identifies some key topic areas that may be used to expand on the high-level requirements, i.e., during consultation sessions or workshops, based on various perspectives in alignment with Enterprise Architecture domains, if there exist and as appropriate.

Table 2 - Enterprise Architecture Domains to Guide Requirements Elicitation

Domains	Sample Topic Areas
Business	<ul style="list-style-type: none"> (a) What other business functions and processes may be impacted by these requirements? (b) What other business functions and processes are similar/related?
Application	<ul style="list-style-type: none"> (a) How to support new requirements at application level. E.g., can the current application be customised, or new application is required? (b) What applications will be affected? (c) How to support new requirements at system level? E.g., any silo systems, legacy systems, systems affected by requirement, etc. (d) What kind of integration is required and how many?
Data	<ul style="list-style-type: none"> (a) Is a new or updated data model required? (b) What data needs to be converted/migrated? (c) Is there data privacy concerns involved?
Technology	<ul style="list-style-type: none"> (a) Is the infrastructure sufficient to support upcoming requirements?
Security	<ul style="list-style-type: none"> (a) Any IT security policy needs to be addressed? (b) Any security concern over the project?

Table 3 - Other Considerations

Domains	Sample Topic Areas
Business	<ul style="list-style-type: none"> (a) Vision – Understand the purpose of the project, whether the requirement is in line with the purpose of the project. (b) Strategy – Elicit the business requirements, i.e., what needs to be achieved.

Organisational	(a) What user groups, B/Ds, external parties will be impacted? (b) Will different parties have different requirements? (c) Any conflicting interests/requirements? (d) Departmental or geographical considerations?
-----------------------	--

7 REQUIREMENTS CATALOGUE TEMPLATE

(Reference Chapter of the Guide – 2.4.2 Elicit High-Level Requirements)

The following template can be used to capture requirements:

Requirements Catalogue Template	
Functional Requirement	
Item	Description
Requirement ID	< Specify a unique ID for each requirement entry >
Requirement Title	< Specify a title for the requirement >
Priority	< Input priority, refer to Requirements Attribute List Template, Appendix – 8, for more details >
Functional Requirement Description	< Describe the functional requirement in more details >
Frequency of Use	< How frequent is the function used on average; to be included only if that is known at this early stage >
Acceptance Criteria	< Describe how, or to what level of quality the feature should be provided to satisfy the users' needs >
Related Business Process	< Applicable to User Requirement Document only >

Non-Functional Requirement	
Item	Description
Requirement ID	< Specify a unique ID for each requirement entry >
Category	< Specify non-functional requirement category >
Requirement Title	< Specify a title for the requirement >
Priority	< Input priority, refer to Requirements Attribute List Template, Appendix – 8, for more details >
Non-Functional Requirement Description	< Describe the non-functional requirement in more details >

The Project Planner should note the following:

- i) **Requirement ID** – a unique ID should also be assigned to each requirement;
- ii) **Requirement Title** – title of the requirement, for quick differentiation;
- iii) **Functional Requirement Description** – describing the required features of the requirement, i.e., ‘what’ the requirements should do. Requirements should be captured through interviews and workshops with users. The functional requirement can include the reasons and objectives of the requirement and benefits expected when the new service is provided, where appropriate;
- iv) **Non-Functional Requirement Description** – describing the non-functional requirement of the system, i.e. ‘how’ the system should work. ;

- v) **Frequency of Use** – the frequency of the function used on average; to be included only if that is known at this early stage.
- vi) **Priority** – describing the priority attributes of the requirement. The priority attributes of the requirement may be derived from *Appendix – 8 Requirement Attributes List Template*.
- vii) **Acceptance Criteria** – describing how, or to what level of quality the feature should be provided to satisfy the users’ needs.
- viii) **Related Business Process** – this field is not required for the requirement catalogue, however, this is a placeholder of the requirement description details for the user requirements document (URD).

(Project Planner can reuse the information captured in the requirements catalogue and populate the list of functional requirements and list of non-functional requirements under the URD. The requirements catalogue is in the same format as the URD requirement description details, except the related business process field. For further information on the user requirements document, please refer to the “[Best Practices for BA²”](#).)

8 REQUIREMENT ATTRIBUTES LIST TEMPLATE

(Reference Chapter of the Guide – 2.4.2 Elicit High-Level Requirements)

Requirement Attributes List											
Requirement ID	Requirement Title	Requirement Description	Business Function and Process	Mission Criticality	Urgency	Effort	Impacted Users	Dependency	Constraint	Benefit	Other Considerations

The details of the requirements should be logged based on the following attributes:

- (a) **Requirement ID** – a unique ID should also be assigned to each requirement;
- (b) **Requirement Title** – title of the requirement, for quick differentiation;
- (c) **Requirement Description** – describing the required features of the requirement, i.e., ‘what’ the requirements should do. Requirements should be captured through interviews and workshops with users. The functional requirement can include the reasons and objectives of the requirement and benefits expected when the new service is provided, where appropriate;
- (d) **Requirement Attributes** – Business function and process, mission criticality, urgency, effort, impacted users, dependency, constraint, benefit and other considerations such as stakeholder agreement, business or technical risk and implementation difficulty. These attributes are elaborated under *Requirement Attributes* below.

Requirement Attributes

Below are the descriptions of the requirement attributes:

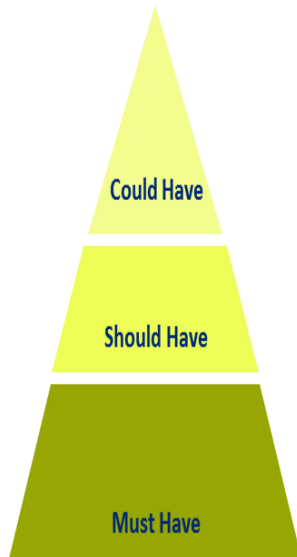
Attributes	Description/Remark
1. Business Function and Process	(a) Grouped by end-to-end business function and process
2. Benefit	(a) The level of benefit being generated given the cost; (b) Benefit should be rated as high, medium, and low; (c) Higher benefit should be implemented first; and (d) Examples of requirements that provide benefit are those that decrease the time to complete a task, reduce the cost of a business process, and improve the customer experience.
3. Mission Criticality	(a) Necessity and criticality of requirements. This is how important this requirement is to the success of the project; (b) MoSCoW Prioritisation Method will be leveraged (<i>Appendix A – 9 MoSCow Prioritisation</i>) (c) ‘Must Have’, ‘Should Have’, ‘Could Have’, ‘Won't Have’; (d) ‘Must Have’ requirements should be implemented first; and (e) Examples of Must Have requirements are those needed to complete essential tasks, such as printing licencing forms. Examples of ‘Should Have’ requirements that will not impact the core operations, such as sending an e-mail reminder.
4. Urgency	(a) If a requirement needs to be implemented earlier in the project; (b) Urgency can be rated as high, medium, and low; (c) Some requirements may have a mandate or hard deadline; and (d) Usually based on management decision, with objective to address pressing issues, which may be policy driven or public facing.
5. Effort	(a) The amount of effort required to implement the requirement

Attributes	Description/Remark
6. Impacted Users	(a) User parties that are impacted by the requirement; and (b) This allows the Project Planner to identify what users are affected by the requirements, and vice versa, during decomposition, deployment approach, and other relevant processes.
7. Dependency	(a) Requirements that have inter-linkages with other requirements; and (b) Some dependencies may not be discovered until a later stage, but it should be captured to the Project Planner's best attempt.
8. Constraint	(a) Limitations that may require different options to address the requirement; (b) Regulatory or policy compliance constraints; (c) Other systems that are also impacted by the requirements; and (d) This allows the Project Planner to identify any technical dependencies and coupling during decomposition.
9. Other Considerations	(a) Items that may induce risk to the project due to this requirement; (b) Political, organisational, public, geography, departments, and other complexities; and (c) BA may also consider stakeholder agreement, business or technical risk, implementation difficulty.

9 MOSCOW PRIORITISATION

(Reference Chapter of the Guide – 2.4.2 Elicit High-Level Requirements)

- (a) The MoSCoW Prioritisation Method is a useful tool when prioritising project requirements. The framework attempts to put a level of importance on each requirement. The framework breaks requirements into 4 groups: ‘Must Have’, ‘Should Have’, ‘Could Have’ and ‘Won't Have’. The definition of each group is shown in the table below:



MoSCow Prioritisation	
Must Have	(a) Must have this requirement to meet core business needs (b) The project cannot deliver on target date without this requirement (c) The project may fail if the requirement is missing
Should Have	(a) Should have this requirement if possible, but project success does not rely on the requirement
Could Have	(a) Could have this requirement if it does not affect anything else in the project
Won't Have	(a) Would like to have this requirement in later phases, but it won't be delivered in this phase of the project

- (b) It is also important to set restrictions around the number of requirements that can be classified as ‘must haves’. In order for the project to meet its schedule and budget, not all requirements can be classified as ‘must haves’. The project manager should examine the requirements and the schedule and limit the amount of must haves accordingly.
- Users and stakeholders have a tendency to assign high importance, or ‘Must Have’, to most requirements, because all requirements are important if there is no assessment on the level of relative importance between requirements. To avoid abusing the use of ‘Must Haves’, the Project Planner should educate the stakeholders about the relative levels and true meaning of each importance level, so the gathered information is meaningful and can truly aid the prioritisation at a later step, when the project schedule is being formulated.

10 REQUIREMENTS VS. SUB-PROJECTS MAPPING TEMPLATE

(Reference Chapter of the Guide – 2.4.4 Prioritise Sub-Projects)

Requirement Attributes List												Sub-Project	
Requirement ID	Requirement Title	Requirement Description	Business Function and Process	Mission Criticality	Urgency	Effort	Impacted Users	Dependency	Constraint	Benefit	Other Considerations	Sub-Project ID	Sub-Project Title

- (a) The information of this mapping is based on information populated on the Requirement Attributes List. The information is included as the columns under the "Requirement and Attributes" section here. Please refer to *Appendix – 8 Requirement Attributes List Template* for further details. The "Sub-Project" section is required to be filled in by the Project Planner with the corresponding sub-project information:
- i) **Sub-Project ID** – The ID of the sub-project which the requirement is grouped under; and
 - ii) **Sub-Project Title** – title of the sub-project, for quick differentiation.

11 SUB-PROJECTS PRIORITISATION ATTRIBUTE LIST TEMPLATE

(Reference Chapter of the Guide – 2.4.4 Prioritise Sub-Projects)

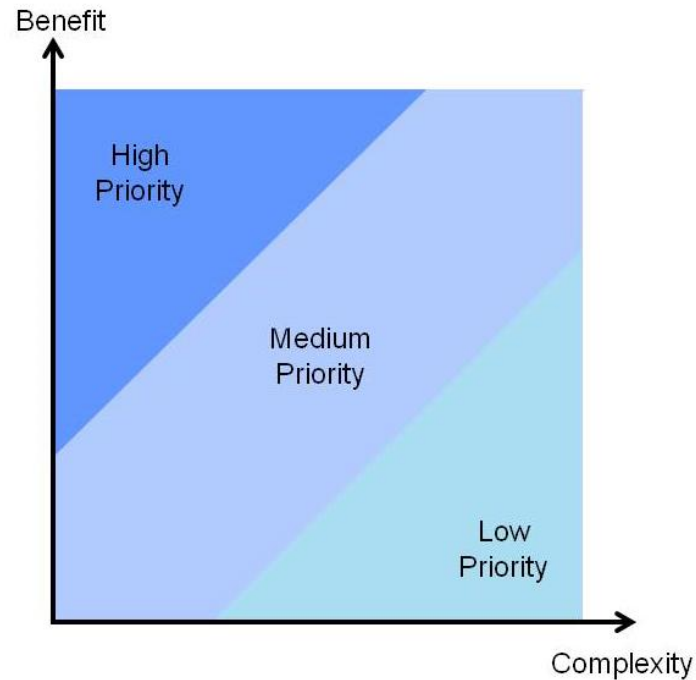
Sub-Projects Prioritisation Attribute List							
Sub-Project ID	Sub-Project Title	Benefit	Complexity	Benefit vs. Complexity	Dependency	Organisational Readiness	Duration

The sub-projects and the attributes of the sub-projects should be logged as follows:

- (a) **Sub-Project ID** – a unique ID should also be assigned to each sub-project;
- (b) **Sub-Project Title** – title of the sub-project, for quick differentiation;
- (c) **Benefit** – Since requirements serving the same business function/process and with higher benefits should be grouped into the same sub-project, the sub-projects that bring higher benefits should have a higher priority to be implemented first. The information captured under Requirements vs. Sub-Project Mapping can be leveraged (*Appendix A - 10 Requirements vs. Sub-Project Mapping Template*).
- (d) **Complexity** – Evaluate sub-project complexity using Complexity Assessment Areas. Please refer to *Chapter – 2.2 Assess Project Complexity*.

- (e) **Benefit vs. Complexity** – High, medium and low priority based on the level of benefit vs. complexity:

Figure 2 - Benefit vs. Complexity Illustration



- (f) **Dependency** – Other sub-projects that are the precedent that this sub-project is relying on.
(g) **Organisational Readiness** – Degree of organisational readiness in low, medium and high.
(h) **Duration** – Estimated time required to complete sub-project for scheduling purpose.

12 PROGRAM EVALUATION AND REVIEW TECHNIQUE (PERT)

(Reference Chapter of the Guide – 2.4.5 Develop High-Level Project Schedule)

The PERT involves the following steps:

- (a) Identify the specific activities and milestones of the project
 - i) The activities are the high-level tasks that are required to complete the project. The milestones are the activities that mark the end of the tasks.
- (b) Determine the proper sequence of the activities
 - i) Go through the list of sub-projects and identify the dependencies among them.
- (c) Construct each activity on a node network diagram
 - i) Based on the sequence of the activities, a network diagram can be drawn to show the dependencies of the activities.
 - ii) The network diagram should consist of arrows and nodes. The nodes represent milestones.
- (d) Estimate the time required for each sub-project
 - i) Optimistic time estimate – the shortest time in which an activity can be completed.
 - ii) Normal time estimate – the completion time with the highest probability.
 - iii) Pessimistic time estimate – the longest time that an activity needs in order to be completed.
- (e) Determine the critical path using Critical Path Methodology.
- (f) Revisit the PERT chart as the project progresses.
 - i) As the project progresses, more details will be revealed where estimates can be replaced with actual figures.
- (g) PERT allows the Internal PM to estimate the expected project completion time and identify the critical path activities that may directly impact on the completion time. However, there may be some limitations of PERT which the Internal PM should be aware of. As PERT relies on professional judgment to estimate the time needed to complete an activity, these estimates are somewhat subjective. To mitigate the risk of under/over-estimation, the Internal PM may leverage past project experience and assess whether the experience is applicable to the sub-projects.
- (h) The Internal PM may leverage project management software to visualise the network diagram.

13 DEPLOYMENT APPROACHES

(Reference Chapter of the Guide – 2.4.5 Develop High-Level Project Schedule)

- (a) The following are common deployment approaches for IT projects. The description and applicability of each approach are listed. Considerations when applying the approach is also listed for the Project Planner to make reference to.

Table 4 - Deployment Approaches

Approach	Description, Applicability and Considerations
<p>1. Phased Deployment</p>	<p>(a) Description:</p> <ul style="list-style-type: none"> i) Same functionality is being deployed to different user groups in phases; and ii) To break deployment into more manageable size. <p>(b) Applicability:</p> <ul style="list-style-type: none"> i) Projects with a large user population, and not demanding a ‘big bang’ roll out. <p>(c) Considerations:</p> <ul style="list-style-type: none"> i) Take longer time than a ‘big bang’ approach; ii) Number of phases should be adequate and not excessive; too many phases gives the project team many overheads; iii) User groups in the same phase should have certain similarity, e.g., functions, scale, geographic location, or organisational readiness; and iv) User groups with higher success rate should take place earlier.
<p>2. Pilot Deployment</p>	<p>(a) Description:</p> <ul style="list-style-type: none"> i) A type of phased approach, with the first phase being the pilot; ii) Allows B/Ds to validate the deployment approach with a selected subset of users; and iii) Identify key issues of approach or change in deployment approach early and address these issues prior to subsequent deployment. <p>(b) Applicability:</p> <ul style="list-style-type: none"> i) Projects without proven deployment approach;

Approach	Description, Applicability and Considerations
	<ul style="list-style-type: none"> ii) Projects with high risk or huge impact during deployment, requiring higher level of assurance; and iii) Projects requiring a proof of success in deployment. <p>(c) Considerations:</p> <ul style="list-style-type: none"> i) Need to allocate time for reviewing the lessons learnt and revising subsequent deployment strategy; ii) May be time consuming to have a separate session if the benefit is not worthwhile; and iii) Does not apply to projects requiring a ‘big bang’ roll out. <p>(d) Pilot user groups selection:</p> <ul style="list-style-type: none"> i) Selection of participants in the pilot should have a good representation of the user population with diversity in user characteristics, if possible; and ii) User groups with smaller scale of end users and less complex business operations may be selected to increase the success rate.
<p>3. Soft Launch</p>	<p>(a) Description:</p> <ul style="list-style-type: none"> i) Where a project deliverable launches an initial version to all the users; ii) Does not require all end users to adopt the new system for a certain period of time; and iii) Announcement of the new system may be optional. <p>(b) Applicability:</p> <ul style="list-style-type: none"> i) Functionalities are new and may not replacing existing ones; ii) New solution may co-exist with existing ones, or replacing them in a later stage; and iii) Usage of the system is optional at the moment. <p>(c) Considerations:</p> <ul style="list-style-type: none"> i) Need to allocate time for reviewing the lessons learnt and revising subsequent deployment strategy; ii) Require double effort in setting up the launch twice; and

Approach	Description, Applicability and Considerations
	<ul style="list-style-type: none"> iii) Time consuming to have a separate session if the benefit is not worthwhile.
<p>4. Big Bang</p>	<ul style="list-style-type: none"> (a) Description: <ul style="list-style-type: none"> i) Where a project deliverable 'goes-live' all at once; and ii) Requires all end users to adopt the new system at the same time. (b) Applicability: <ul style="list-style-type: none"> i) Preferred when the new system is rolled out to a small number of end users who are comfortable with new technology and change ready; and ii) Business operations that do not involve a complex workflow may be good candidates. (c) Considerations: <ul style="list-style-type: none"> i) Single point of failure will affect the entire population of users, rather than a subset of users; ii) Labour intensive to test the entire system prior to roll out to ensure one-time success; and iii) May reduce costs, as it eliminates the additional costs associated with multiple phases and can reduce project time.

- (b) The following diagram is an example of the applicability of different deployment approaches in association with some selected risk areas:

Items	Deployment Approach			
	Phased	Pilot	Soft Launch	Big Bang
Multiple Stakeholders	✓	✓		
Major Business Transformation	✓	✓		
Technology Risk		✓	✓	
Political, Public & Media Impact	✓		✓	✓
Legislation Changes	✓			✓
Delivery Timeframe	✓			
User Population	✓	✓	✓	

Figure 3 - Deployment Approach Applicability Table

- (c) Please note that the above table is for illustration only. As mentioned, the Project Planner should allow flexibilities in deciding the deployment approach for the project and/or sub-projects in view of the actual situation.

14 PILOT PROJECTS

(Reference Chapter of the Guide – 2.4.5 Develop High-Level Project Schedule)

- (a) Pilot project is a good way to test a new initiative in small scale to prove the feasibility and practicality of a project concept.
- (b) B/Ds can use them to roll-out new software or test new processes. The use of pilot project is even more significant to Projects, due to the level of risk involved.
- (c) **Benefits** - Multiple benefits have been identified for using pilot projects, and the key benefits are listed below:

Table 5 - Key Benefits of Pilot Projects

Key Benefit	Description
1. Prove Concept	(a) When exploring a new topic or attempting research for the first time, a pilot project can help testing the concept.
2. Gain Experience	(a) Experience will be gained, no matter the project succeeded or failed; and (b) Subsequent project can leverage such experience.
3. Build Confidence	(a) A pilot project helps project team to build confidence and acquire professional skills; (b) Initial deployment points should be of small scale and have low risk, to increase success rate; and (c) Success stories can convince sceptics.
4. Reduce Risks	(a) Mitigate risks because of smaller scale; and (b) If the project fails, it fails fast and impact is limited.

- (d) **Considerations** - While pilot projects can bring several benefits and it is quite useful for Projects, it may not necessarily suit all situations, and the following needs to be taken into consideration:

Table 6 - Key Considerations of Pilot Projects

Key Consideration	Description
1. Project Timeline	(a) Is there enough time to allow a pilot project to test out the concepts?
2. Necessity	(a) Is a pilot necessary for the project to conduct? (b) What is the reason to support conducting a pilot?
3. Project Scale	(a) Pilot project scale should be small and manageable (b) Effort should be spent on the project itself, rather than the management of the project

Key Consideration	Description
	(c) A controllable environment can gather clear and valuable results
4. Participants/Deployment Points	<ul style="list-style-type: none"> (a) Selecting the participants or deployment points is important (b) Although small, but the sample size should be representing different groups of the user population (c) Considerations are size of user groups, geographical locations, functions, and characteristics
5. Scalability/Full-Scale Roll-Out	<ul style="list-style-type: none"> (a) Does success with pilot imply success with full-scale roll-out? (b) Would full-scale roll-out induce other requirements? (c) Would full-scale roll-out induce other risks and complexity? (d) Would full-scale roll-out generate the same benefits? (e) Any outliers or special cases?

15 IMPLEMENTATION COMPLEXITY MATRIX

(Reference Chapter of the Guide – 2.4.7 Estimate Project Budget)

- (a) Implementation Complexity Matrix is a bottom-up estimation methodology suitable for high-level project estimation. This approach provides an alternative way for B/Ds to estimate effort and resource. The estimates may be derived from past project experiences, RFI and market research. It focuses on estimating the implementation / customisation effort of a project. The Project Planner may leverage similar previous project estimations, if available, for calculation. The matrix is developed and used in the following steps:
- (b) List the relevant technical components for this project that requires development or customisation. For example, user interface, web portal, report and analytics. General custom build effort is also considered a technical component, to ensure completeness in the estimation.
- (c) List the effort (and cost) estimates for implementation of a certain complexity (high, medium or low) within a technical component. The scope associated with each level of complexity needs to be defined by the Internal PM, or reusing the complexity scale from previous projects.
- (d) These complexity ratings and estimates will populate for the implementation complexity matrix.
- (e) List the required functions. This can be of high-level or lower-level, as long as all the functions are pitched at the same level. WBS or other breakdown structure can be used.
- (f) For each function within an application, the Internal PM can determine the involved technical components and approximate level of complexity involved.
- (g) Use the value defined in the complexity matrix to calculate the estimates for this function.
- (h) Implementation complexity matrix provides a systematic method to approximate the estimates. It is understandable the level of detail is preliminary during earlier phases and the actual effort and resources required are uncertain at the moment. Therefore, an approximation already serves the purpose.

Sample Implementation Complexity Matrix

- (a) The technical aspect of a project can be broken down into components such as an online portal, workflow engine, mobile application, etc. and range of outputs for different levels of complexities, for example, a low complexity portal is less than 20 static pages. Then, for each technical component, the amount of man-months needed for implementation in terms of low, medium and high complexity are estimated based of previous project experience.

(b) The following is a sample Implementation Complexity Matrix, the data of man-months is based upon industry practices :

(B/Ds can refer to the textboxes below for further explanation on how the numbers were derived. The figures are for illustrative purpose only.)

Technical Component	Complexity	Description	Man-Months
Portal	Low	< 20 static pages < 10 dynamic pages None to little workflow and logic None to little portal integration	6.0
	Medium	< 50 static web pages < 20 dynamic web pages Moderate amount workflow and logic Moderate amount of portals	12.0
	High	< 150 static web pages < 60 Dynamic web pages Large amount of workflow and logic Large amount of portents	36.0
Workflow Engine	Low	< 10 process automation < 3 system / human integration points < 5 of business rules	6.0
	Medium	< 20 process automation < 6 system / human integration points < 10 of business rules	12.0
	High	< 40 process automation < 12 system / human integration points < 20 of business rules	36.0
Mobile Device User Interface	Low	< 5 app features Low graphics	3.0

Further explanation:
An estimation of 6 man-months is derived based on similar past project experience in setting up a simple portal.

Technical Component	Complexity	Description	Man-Months
		intensity Low integration complexity	
	Medium	< 10 app features Moderate graphics intensity Moderate integration complexity	6.0
	High	< 25 app features High graphics intensity High integration complexity	18.0
Enterprise Content Management (ECM)	Low	< 3 custom file formats and < 20 document types Simple document management and workflow Simple record management policy	6.0
	Medium	< 6 custom file formats and < 50 document types Moderately complicated document management and workflow Moderately complicated record management policy	12.0
	High	< 18 custom file formats and < 100 document types Highly complicated document management and workflow Highly complicated record management policy	36.0

Further explanation:
An estimation of 6 man-months
is derived based on similar RFI
from vendor.

Technical Component	Complexity	Description	Man-Months
Reporting and Analytics	Low	< 25 reports < 3 data entities < 5 dashboards -Low number of filters -Low number of columns -Non-customisable	6.0
	Medium	< 50 reports < 5 different data entities < 15 dashboards -Medium number of filters -Medium number of columns -Moderately customisable	12.0
	High	< 150 reports < 8 different data entities < 50 dashboards -High number of filters -High number of columns -Highly customisable	36.0
Custom Build Effort	Low	low complexity features	6.0
	Medium	moderate complexity features	12.0
	High	high complexity features	24.0
Security	Low	<3 security features	3.0
	Medium	3-10 security features	6.0
	High	10+ security features	12.0

Elaborations:

- (a) **Technical Component** – Technical component that will be developed/modified in the project.
- (b) **Complexity** – Assigned implementation complexity involved with the technical component. Only 3 levels of Low, Medium, and High to ensure simplicity in evaluation.

- (c) **Description** – What is considered Low, Medium or High complexity for implementing the technical component? These are defined by the Internal PM and can be adjusted depending on the level of detail at the current phase
- (d) **Effort** – The effort for implementing Low, Medium, or High complexity of the technical component. The unit can be in man-month or man-day, depending on the level of detail. The effort should align with the description (the task involved) at the 3 levels.

Sample Calculations

- (a) For illustration purpose, two sample requirements effort estimation are shown:

i) Requirement: Reporting functionality on mobile for management

	Portal	Workflow Engine	Mobile Device User Interface	ECM	Reporting and Analytics	Custom Build Effort	Security
Complexity	N/A	N/A	Medium	N/A	Low	N/A	Low
Effort	0	0	6	0	6	0	3

Total: 15 man-months

ii) Requirement: Advanced reporting functionality on Web Portal for staff

	Portal	Workflow Engine	Mobile Device User Interface	ECM	Reporting and Analytics	Custom Build Effort	Security
Complexity	Medium	N/A	N/A	N/A	Medium	Medium	Low
Effort	12	0	0	0	6	12	3

Total: 33 man-months

16 OTHER CONSIDERATIONS IN HIGH-LEVEL ESTIMATION

(Reference Chapter of the Guide – 2.4.7 Estimate Project Budget)

In order to have a complete estimation in effort and budget, with considerations regarding scale and complexity of the project, the following should be taken into considerations to ensure completeness of the high-level estimation:

Table 7 - Other Considerations in High-level Estimation

Items	Considerations
1. Procurement Activities and Effort	<ul style="list-style-type: none"> (a) Effort concerned to facilitate the subsequent tendering process, to acquire any outsourced services to implement/support the Project; and (b) For complex and high value tenders, sufficient time should be allocated for procurement process.
2. Project Management Effort	<ul style="list-style-type: none"> (a) Extra resource, e.g., a Project Administrator role, is recommended for management of multiple sub-projects and possibly multiple contractors; and (b) Estimated as a percentage of the implementation effort.
3. Other Stakeholder Involvement	<ul style="list-style-type: none"> (a) Internal resources required to support the (internal or out-source) Sub-Project Team by attending consultation sessions, providing business requirements, conducting UAT, and other activities as required by the project; External stakeholders (e.g., other B/Ds, NGOs, and other related organisations) should also be considered for internal resources required to support activities for the project. (b) May consider establishing an independent testing group and separate user representative groups to get diverse requirements; (c) Take considerations regarding overall or separated involvements, such as requirements elicitation, final overall UAT, training, and socialisation; and (d) Leveraging the percentage of business involvement to derive estimation as a percentage of the implementation effort.
4. Hardware Implementation Overhead	<ul style="list-style-type: none"> (a) Extra overhead cost/effort involved in design, coordination, and project management of hardware implementation, in order to align among multiple sub-projects; and (b) Estimated as a percentage of the hardware cost.

Items	Considerations
5. Software Installation Overhead	<ul style="list-style-type: none"> (a) Extra overhead cost/effort involved in design, coordination, and project management of software installation, in order to align among multiple projects; and (b) Estimated as a percentage of the software cost.
6. Data Related Effort	<ul style="list-style-type: none"> (a) Effort required to conduct data cleansing, conversion and migration, where necessary, to ensure data is usable and aligned among the new sub-projects; (b) Assess the amount of data to be converted/migrated from existing systems (including paper-based) to the new ones through Sizing estimation; and (c) Approximation during early phases can be abstracted as a percentage of the implementation effort.
7. System Integration	<ul style="list-style-type: none"> (a) Effort required for developing integration between existing and new systems, as well as any centralised system, such as user directory, Enterprise Content Management/Electronic Information Management, etc.; and (b) Number of system integrations needs to be estimated.
8. Transitional interfaces with existing systems	<ul style="list-style-type: none"> (a) Effort required for developing any transitional interfaces with existing systems; and (b) Number of transitional interfaces with existing systems, in order to make the project transition smoothly to the new systems.
9. Testing (e.g., User Acceptance Testing, System Integration Testing)	<ul style="list-style-type: none"> (a) Take considerations regarding testing requirements due to the business needs of the project, for example, additional final UAT due to Big Bang deployment; and (b) Estimated as a percentage of the implementation effort.
10. Training	<ul style="list-style-type: none"> (a) Cost/effort to educate users for adoption of new systems; (b) Take considerations regarding training requirements, such as level of detail, reusability of training materials, and diversity of training for different user groups; (c) Take considerations of training approaches, such as train the trainers, workshops, and seminars/webinars; and (d) Estimated as a percentage of the implementation effort.

Items	Considerations
11. Nursing	(a) Cost/effort required to maintain the new systems after the roll out to ensure system stability; (b) Considerations regarding if the sub-projects are being rolled-out individually, as a whole, or in combinations; and (c) Estimated as a percentage of the implementation effort.