



Australian Government

Department of Education, Employment and Workplace Relations

ICAPRG502A Manage a project using software management tools

Release: 1

ICAPRG502A Manage a project using software management tools

Modification History

Release	Comments
Release 1	This Unit first released with <i>ICA11 Information and Communications Technology Training Package version 1.0</i>

Unit Descriptor

This unit describes the performance outcomes, skills and knowledge required to use software-management tools to manage a project from initiation to completion.

Application of the Unit

This unit applies to individuals who either work in a team or individually in order to develop a system to specification and use software management tools in order to manage the delivery of the system within time and budget parameters.

Licensing/Regulatory Information

No licensing, legislative, regulatory or certification requirements apply to this unit at the time of endorsement but users should confirm requirements with the relevant federal, state or territory authority.

Pre-Requisites

Not applicable.

Employability Skills Information

This unit contains employability skills.

Elements and Performance Criteria Pre-Content

Element	Performance Criteria
<i>Elements describe the essential outcomes of a unit of competency.</i>	<i>Performance criteria describe the performance needed to demonstrate achievement of the element. Where bold italicised text is used, further information is detailed in the required skills and knowledge section and the range statement. Assessment of performance is to be consistent with the evidence guide.</i>

Elements and Performance Criteria

1. Identify software management tools	1.1 Determine <i>software development methodology</i> to be used for project development 1.2 Determine <i>project-management software</i> to be used to manage the development of the project 1.3 Determine <i>source-control system</i> to manage source code and handle conflicts 1.4 Determine <i>collaboration software</i> to be used in the project development
2. Implement software management tools	2.1 Create project plan according to software requirement specifications 2.2 Define source-control procedures 2.3 Create collaboration environment
3. Monitor use of software management tools	3.1 Monitor and adjust project plan accordingly to maintain progress according to the project plan 3.2 Ensure code is correctly entered into source control system 3.3 Monitor collaboration environment and resolve issues where required

Required Skills and Knowledge

This section describes the skills and knowledge required for this unit.

Required skills

- communication skills to interact with developers and stakeholders during the development cycle
- literacy skills to read and interpret software requirement specifications
- numeracy skills of basic maths for project planning
- planning and organisational skills to:
 - create project plan
 - implement software-management tools
- research skills to determine appropriate software-management tools
- technical skills to use software-management tools.

Required knowledge

- basic knowledge of:
 - project management
 - software development life cycle (SDLC)
 - software requirement specifications
 - version control.

Evidence Guide

The evidence guide provides advice on assessment and must be read in conjunction with the performance criteria, required skills and knowledge, range statement and the Assessment Guidelines for the Training Package.

Overview of assessment	
Critical aspects for assessment and evidence required to demonstrate competency in this unit	Evidence of the ability to: • use software-management tools to manage the development of a system to specification • prepare a project plan using project-management software • use source-control software to manage code • use collaboration tools.
Context of and specific resources for assessment	Assessment must ensure access to: • internet • project-management software • source-control software • specific tools and licences, depending on particular platform • appropriate learning and assessment support when required • modified equipment for people with special needs.
Method of assessment	A range of assessment methods should be used to assess practical skills and knowledge. The following examples are appropriate for this unit: • review of candidate's project plan • verbal or written questioning on knowledge of: • source-control methods • collaboration methods.
Guidance information for assessment	Holistic assessment with other units relevant to the industry sector, workplace and job role is recommended, where appropriate. Assessment processes and techniques must be culturally appropriate, and suitable to the communication skill level, language, literacy and numeracy capacity of the candidate and the work being performed. Indigenous people and other people from a non-English speaking background may need additional support. In cases where practical assessment is used it should be combined with targeted questioning to assess required knowledge.

Range Statement

The range statement relates to the unit of competency as a whole. It allows for different work environments and situations that may affect performance. Bold italicised wording, if used in the performance criteria, is detailed below. Essential operating conditions that may be present with training and assessment (depending on the work situation, needs of the candidate, accessibility of the item, and local industry and regional contexts) may also be included.

<i>Software development methodology</i> may include:	• agile software development • joint applications design • prototyping • rapid application development • waterfall.
<i>Project-management software</i> may include:	• dotProject • FastTrack Schedule • Microsoft Project • OpenProj • Oracle Project Portfolio Management • Teamwork.
<i>Source-control system</i> may include:	• CVS • Microsoft Visual SourceSafe.
<i>Collaboration software</i> may include:	• blogs • forums • Microsoft SharePoint • wikis • Zimbra.

Unit Sector(s)

Programming and software development