



Australian Government

Department of Education, Employment and Workplace Relations

ICAB4232B Maintain open source code programs

Release: 1

ICAB4232B Maintain open source code programs

Modification History

Not Applicable

Unit Descriptor

Unit descriptor	<p>This unit defines the competency required to contribute as a member of an open source software project community and to maintain open source code. The open source development model differs considerably from the traditional commercial model.</p> <p>No licensing, legislative, regulatory or certification requirements apply to this unit at the time of publication.</p>
------------------------	---

Application of the Unit

Application of the unit	
--------------------------------	--

Licensing/Regulatory Information

Refer to Unit Descriptor

Pre-Requisites

Prerequisite units		
	ICAB4222B	Apply introductory programming skills in another language

Employability Skills Information

Employability skills	This unit contains employability skills.
-----------------------------	--

Elements and Performance Criteria Pre-Content

Elements describe the essential outcomes of a unit of competency.	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.
---	--

Elements and Performance Criteria

ELEMENT	PERFORMANCE CRITERIA
1. Introduction to open source paradigm	1.1. Examine the open source paradigm and demonstrate an understanding of the differences from the traditional <i>software</i> development models 1.2. Investigate and demonstrate understanding of the types of <i>on-line resources</i> 1.3. Investigate and demonstrate understanding of the types of project <i>documentation</i> 1.4. Recognise and demonstrate understanding of the role of an <i>on-line community</i> and international collaboration 1.5. Examine and demonstrate understanding of motivational factors for contributors to open source code 1.6. Analyse and demonstrate understanding of open source <i>licensing models</i>
2. Familiarise with target project	2.1. Examine <i>on-line resources</i> associated with the target project 2.2. Download pre-built executable binaries, to install and run project 2.3. Download, read and demonstrate understanding of supporting <i>documentation</i>
3. Prepare for maintenance activities	3.1. Select and register with a relevant <i>on-line community</i> open source group 3.2. Download nightly snapshots of latest source code and supporting <i>documentation</i> 3.3. Build and execute snapshot where appropriate
4. Maintain code	4.1. Access the project bug database and select bugs to be resolved or features to be added 4.2. Make changes to local copy of code to resolve selected bugs 4.3. Test resulting code to ensure it performs appropriately 4.4. Prepare code patch for submission 4.5. Submit code patch to project 4.6. Use appropriate <i>software development tools</i> and environment
5. Maintain documentation	5.1. Access project <i>documentation</i> 5.2. Prepare and contribute new information or updates to existing <i>documentation</i>

ELEMENT	PERFORMANCE CRITERIA
	5.3. Prepare and submit <i>documentation</i> changes to project
6. Participate in community	6.1. Exchange messages with other project members and actively participate in community activities 6.2. Take action to ensure exchanges are <i>socially acceptable</i> 6.3. Submit code and <i>documentation</i> code patches for inclusion 6.4. Access <i>on-line project</i> resources frequently to keep up to date with project and community developments 6.5. Take action to ensure <i>community participation standards</i> are observed and maintained

Required Skills and Knowledge

REQUIRED SKILLS AND KNOWLEDGE

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

Required skills

- Communicating with global peers
- Participating in volatile communities of interest
- Effectively participating in open source projects
- Building existing software projects from source
- Creating and maintaining code
- Testing and debugging code at a high level

Required knowledge

- Open source development methodology
- On-line project communities
- Project-specific knowledge

Evidence Guide

EVIDENCE GUIDE	
<p>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.</p>	
Overview of assessment	
Critical aspects for assessment and evidence required to demonstrate competency in this unit	<p>Evidence of the following is essential:</p> <ul style="list-style-type: none"> • Assessment must confirm that a significant contribution to an open source project has occurred. Contributions will be in the form of: <ul style="list-style-type: none"> • Program source code changes and/or additions • Documentation changes and/or additions • On-line participation in the project direction • Local proof of building of snapshots, code modifications and testing <p>To demonstrate competency in this unit the person will require access to:</p> <ul style="list-style-type: none"> • Software development environment • Access to the internet • Technical requirements • CVS databases • Information repositories • Programming languages • Group facilitation software
Context of and specific resources for assessment	<p>The breadth, depth and complexity of knowledge and skills in this competency would cover a broad range of varied activities or application in a wider variety of contexts most of which are complex and non-routine. Leadership and guidance would be involved when organising activities of self and others as well as contributing to technical solutions of a non-routine or contingency nature.</p> <p>Assessment must ensure:</p> <ul style="list-style-type: none"> • Performance of a broad range of skilled applications including the requirement to evaluate and analyse current practices, develop new criteria and procedures for performing current practices and

EVIDENCE GUIDE	
	<p>provision of some leadership and guidance to others in the application and planning of the skills would be characteristic.</p> <ul style="list-style-type: none"> • Applications may involve responsibility for, and limited organisation of, others.
Method of assessment	<p>The purpose of this unit is to define the standard of performance to be achieved in the workplace. In undertaking training and assessment activities related to this unit, consideration should be given to the implementation of appropriate diversity and accessibility practices in order to accommodate people who may have special needs. Additional guidance on these and related matters is provided in ICA05 Section 1.</p> <ul style="list-style-type: none"> • Competency in this unit should be assessed using summative assessment to ensure consistency of performance in a range of contexts. This unit can be assessed either in the workplace or in a simulated environment. However, simulated activities must closely reflect the workplace to enable full demonstration of competency. • Assessment must confirm competency in all areas of the software development cycle. Code-only solutions are not acceptable. • Assessment will usually include observation of real or simulated work processes and procedures and/or performance in a project context as well as questioning on underpinning knowledge and skills. The questioning of team members, supervisors, subordinates, peers and clients where appropriate may provide valuable input to the assessment process. The interdependence of units for assessment purposes may vary with the particular project or scenario.
Guidance information for assessment	<p>Holistic assessment with other units relevant to the industry sector, workplace and job role is recommended.</p> <p>An individual demonstrating this competency would be</p>

EVIDENCE GUIDE

	<p>able to:</p> <ul style="list-style-type: none"> • Demonstrate understanding of a broad knowledge base incorporating some theoretical concepts • Apply solutions to a defined range of unpredictable problems • Identify and apply skill and knowledge areas to a wide variety of contexts, with depth in some areas • Identify, analyse and evaluate information from a variety of sources • Take responsibility for own outputs in relation to specified quality standards • Take limited responsibility for the quantity and quality of the output of others • Maintain knowledge of industry products and services <p>Additionally, an individual demonstrating this competency would be able to:</p> <ul style="list-style-type: none"> • Contribute to and understand a range of development methodologies and their application to a project or scenario • Demonstrate theoretical knowledge of open source practices • Produce documentation
--	--

Range Statement**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.

Documentation may follow:

- ISO/IEC/AS standards
- audit trails
- naming standards
- version control

RANGE STATEMENT	
	<ul style="list-style-type: none"> • project management templates • report writing principles
<i>On-line project</i> may be:	<ul style="list-style-type: none"> • private website • commercial hosting facility hosting community code • ftp site • CVS site • other type of group repository
<i>On-line community</i>	<ul style="list-style-type: none"> • May include but is not limited to local communities, on-line virtual communities of interest and organisational communities • The main tools for on-line communities are mailing lists, web conferencing and newsgroups.
<i>On-line resources</i> may be:	<ul style="list-style-type: none"> • websites • news groups • CVS trees • gopher sites • ftp sites • community sites • on-line communities
<i>Community participation standards</i> may be:	<ul style="list-style-type: none"> • verbal or written policies • procedures or guidelines • may be informal or formal rules and regulations used by groups to manage their on-line projects.
<i>Software</i> may include but is not limited to:	<ul style="list-style-type: none"> • commercial software applications; organisation-specific software, packaged software, in-house or customised software
<i>Socially acceptable</i> messages should be:	<ul style="list-style-type: none"> • designed and sent with regard to recipient's location • culture • ethnicity • race • social sensitivities
<i>Software development tools</i> may include:	<ul style="list-style-type: none"> • operating systems • target environments • development tools • computer language • version control systems

RANGE STATEMENT	
	<ul style="list-style-type: none"> • development methodology
<p><i>Licensing models</i> may include but are not limited to:</p>	<ul style="list-style-type: none"> • GPL • LGPL • BSD • Mozilla • Apache licences

Unit Sector(s)

Unit sector	Build
--------------------	-------

Co-requisite units

Co-requisite units		

Competency field

Competency field	
-------------------------	--