



Australian Government

ICTPRG506 Design application architecture

Release: 1

ICTPRG506 Design application architecture

Modification History

Release	Comments
Release 1	This version first released with ICT Information and Communications Technology Training Package Version 1.0.

Application

This unit describes the skills and knowledge required to design the structure of software or systems components, and how they interact.

It applies to individuals who may work as software architects, developers, designers, software engineers or programmers responsible for designing, and building, solution architecture.

No licensing, legislative or certification requirements apply to this unit at the time of publication.

Unit Sector

Programming and software development

Elements and Performance Criteria

ELEMENT	PERFORMANCE CRITERIA
<i>Elements describe the essential outcomes.</i>	<i>Performance criteria describe the performance needed to demonstrate achievement of the element.</i>
1. Gather and confirm the architecture requirements	1.1 Gather information regarding the architectural requirements 1.2 Establish and document the architectural requirements
2. Design layered architecture	2.1 Separate the areas of concern into logical layers 2.2 Determine cross-cutting concerns 2.3 Define the system into components 2.4 Identify the responsibilities of each component 2.5 Identify the interconnections between components
3. Plan a strategy to re-use components	3.1 Determine an appropriate strategy for communicating with external systems

ELEMENT	PERFORMANCE CRITERIA
	3.2 Interact with existing legacy components
4. Design for globalisation and localisation	4.1 Determine culture specific information 4.2 Consider database design features 4.3 Select the appropriate user interface 4.4 Develop a software product for worldwide distribution 4.5 Develop a software product for a specific country

Foundation Skills

This section describes language, literacy, numeracy and employment skills incorporated in the performance criteria that are required for competent performance.

Skill	Performance Criteria	Description
Reading	1.1, 2.2, 2.3, 2.4, 2.5, 4.1	Interprets complex technical, and non-technical, information from a range of sources
Writing	1.2	Prepares technical workplace documents that fulfil the expectations of different stakeholders
Communicate with others	4.1, 4.2	Identifies, and explores, the differences between a diverse range of people and cultures in a programming context
Get the work done	1.1, 1.2, 2.1, 2.2, 2.3, 2.4, 2.5, 3.1, 4.2, 4.3, 4.4, 4.5	- Takes responsibility for planning, sequencing, and prioritising tasks in order to achieve the required outcomes - Uses a range of digital tools and sophisticated techniques, to meet the desired outcomes - Identifies technical or conceptual issues, and applies analytical processes to resolve these issues - Uses analytical and lateral thinking to review current practices, and to develop new or improved software or systems

Unit Mapping Information

Code and title current version	Code and title previous version	Comments	Equivalence status
ICTPRG506 Design application architecture	ICAPRG506A Design application architecture	Updated to meet Standards for Training Packages	Equivalent unit

Links

Companion Volume implementation guides are found in VETNet -

<https://vetnet.gov.au/Pages/TrainingDocs.aspx?q=a53af4e4-b400-484e-b778-71c9e9d6aff2>

Assessment Requirements for ICTPRG506 Design application architecture

Modification History

Release	Comments
Release 1	This version first released with ICT Information and Communications Technology Training Package Version 1.0.

Performance Evidence

Evidence of the ability to:

- design layered applications to increase application performance, scalability and re-usability
- create flexible and configurable applications, and adapt these to suit different locales
- adapt a software product to a particular language, and culture, of the target market.

Note: If a specific volume or frequency is not stated, then evidence must be provided at least once.

Knowledge Evidence

To complete the unit requirements safely and effectively, the individual must:

- outline the principles of database design
- outline current software development methodologies
- explain object-oriented programming and its role in application architecture
- describe the software development life cycle.

Assessment Conditions

Gather evidence to demonstrate consistent performance in conditions that are safe and replicate the workplace. Noise levels, production flow, interruptions and time variances must be typical of those experienced in the programming and software development industry, and include access to:

- both functional and non-functional requirements
- the tools to design software architecture
- specific information for the target culture, or language
- the appropriate learning and assessment support, when required.

Assessors must satisfy NVR/AQTF assessor requirements.

Links

Companion Volume implementation guides are found in VETNet -

<https://vetnet.gov.au/Pages/TrainingDocs.aspx?q=a53af4e4-b400-484e-b778-71c9e9d6aff2>