



Australian Government

Department of Education, Employment and Workplace Relations

ICAPRG410A Build a user interface

Release: 1

ICAPRG410A Build a user interface

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 design, build and test a user interface (UI) to specification.

Application of the Unit

This unit is relevant to those who work as user-interface designers and responsible for specifying the layout and style of the desired user interface. They may be developers working in the area of user-interface design and implementation.

The context of the unit applies to different types of user interfaces, such as command-line interfaces (CLI), graphical user interfaces (GUI), web user interfaces (WUI) and natural user interfaces (NUI).

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. Build a prototype user interface	1.1 <i>Gather requirements</i> for the user interface 1.2 Determine the application <i>development language</i> for writing the user interface 1.3 Build a prototype using <i>prototyping tools</i> available in the determined language 1.4 Review UI prototype with client and edit as required
2. Design UI	2.1 Formulate application content flow 2.2 Design <i>UI components</i> 2.3 Define <i>UI actions</i> 2.4 Itemise UI events
3. Build UI	3.1 Determine the appropriate language 3.2 Build UI with the required functionality
4. Test UI	4.1 Test UI for overall functionality according to requirements 4.2 Iterate UI design or build until test results meet requirements
5. Document UI and obtain client sign-off	5.1 Document user requirements 5.2 Document UI 5.3 Obtain client sign-off to completed UI

Required Skills and Knowledge

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

Required skills

- communication skills to provide advice and guidance to others
- literacy skills to read basic technical data
- numeracy skills to make basic calculations for specifying the layout of the UI
- research skills to keep up-to-date with UI technologies and tools
- technical skills to:
 - create applications using basic programming techniques
 - create web pages using hypertext markup language (HTML) and cascading style sheet (CSS)
 - debug applications using basic debugging techniques
 - test applications using basic testing techniques.

Required knowledge

- maths at a basic level
- object-oriented programming
- open-source development tools
- small-size application development.

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: - design a fully functional UI in consultation with the user using the determined language - create a prototype UI - review the prototype with the user - build the UI - test that created UI is complete, consistent and functional.
Context of and specific resources for assessment	Assessment must ensure access to: - internet in order to use online tools available for prototyping and UI design - integrated development environment (IDE) for the determined language - specific tools and licenses are available depending on particular platform, such as Windows, Linux, Unix, Mac and Google Chrome - 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: - evaluation of candidate's: - UI prototype - completed UI - review of candidate's UI documentation - verbal or written questioning to determine candidate's knowledge of: - UI components - prototyping methodology.
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>Gather requirements</i> may include:	• interviews • observation • questionnaires.
<i>Development language</i> may include:	• object-oriented language, such as: • ASP.NET • C++ • Java • JavaScript • JSP • NET (VB, C#) • Python • XHTML, DHTML and CSS.
<i>Prototyping tools</i> may include:	• commercial applications, such as: • Adobe FireWorks CS4 • Adobe Flash Catalyst • Axure • Balsamiq Mockups • Microsoft Expression Blend • MockupScreens • open-source tools, such as: • FluidIA • Napkin Look & Feel • Pencil Sketching • web-based tool, such as: • Hot Gloo • Mockingbird.
<i>UI components</i> may include:	• common and basic controls, such as button, label, textbox, checkbox and combo box • container controls • data controls • dialogs • menus and toolbars controls

	• printing controls.
UI actions may include:	• keyboard actions along with the key modifiers • keyboard actions, such as key press, key release and key action • mouse actions include moves, button presses, button releases and dragging • use of emerging interactive surface technologies and the NUI.

Unit Sector(s)

Programming and software development