# Assessment event 2 of 3: Project

## Criteria

### Unit code and name

ICTPRG302 | Apply introductory programming techniques

### Qualification/Course code and name

Select your Qualification/Course code and name from the dropdown.

ICT30120 | Certificate III in Information Technology (2)

## Student details

Student name

Student number

Version: 20230504

Date created: 4 May 2023

© TAFE NSW 2023  
RTO Provider Number 90003 | CRICOS Provider Code: 00591E

This assessment can be found in the TAFE NSW [Learning Bank](https://share.tafensw.edu.au/share/logon.do?.page=searching.do?in%3DC1b145167-45e0-41ec-9f64-92af668e3e54%26q%3D%26type%3Dstandard%26sort%3Drank%26dr%3DAFTER%26page%3D1).

The content in this document is copyright © TAFE NSW 2023 and should not be reproduced without the permission of TAFE NSW. Information contained in this document is correct at time of printing: 21 June 2024. For current information please refer to our website or your teacher or assessor as appropriate.

## Assessment instructions

Table 1 Assessment instructions

| Assessment details | Instructions |
| --- | --- |
| **Assessment event overview** | The aim of this assessment event is to assess your knowledge and performance in creating simple applications through introductory programming techniques.  This assessment event assesses the student's knowledge and performance skills requirements of the unit.  This assessment is in 4 parts:   1. The project brief 2. Design the application 3. Develop the application 4. Modify, test and record results   This assessment is supported by:   * [ICTPRG302\_AE\_Pro2of3\_Appx](https://share.tafensw.edu.au/share/items/a1e005b9-95b0-4f87-b455-603e78d5dbf0/0/?attachment.uuid=1f727346-f15f-47c8-a21b-92f0a3e3ad0b), which contains:   + accounts.txt (test data)   + Gelos Algorithm Design Form   + Gelos Program Design Form   + Gelos Software Test Report Form * Interactive video recording * Assessment Checklist * Assessment feedback (not included here)   **Note**: This assessment may contain links to external resources. Access to the long URL is provided via the External resources – Links and URLs section located at the end of this document. |
| **Unit assessment guide** | Refer to the unit assessment guide (UAG) before attempting this assessment event. The UAG contains information, including assessment requirements and how to achieve a satisfactory result. |
| **Submission instructions** | When you complete this assessment:   * read the checklist at the end of the assessment to make sure you have completed everything * keep a copy of all the electronic and hardcopy assessments you submit to TAFE NSW * make sure you have completed the assessment declaration before you submit. |

# Specific task instructions

To complete this part of the assessment, you must provide evidence of participating in an interactive role-play. This will be achieved by viewing an interactive video and then capturing your responses in a recording.

Refer to the [Observation Checklist](bookmark://ObsChklst) to understand what skills you need to demonstrate in this section of the assessment. This checklist outlines the assessment criteria your Teacher/Assessor will mark you on.

Once completed, the recorded evidence will be submitted via the online platform to the Teacher/Assessor for marking.

This digital recording may be an audio file (sound only) or a video file (video and audio). You may use your computer webcam and capture software or your mobile phone. Ensure you have access to the required equipment and resources.

If space or bandwidth is limited, create an audio file rather than a video. Video file uploads are limited to 1 GB.

TIP: The following may be helpful: [video recording instructions (pdf)](https://share.tafensw.edu.au/share/file/744af7d4-a241-45e2-adb0-0e13f2fe4950/1/VideoAssessmentInstructions.pdf). This page includes useful tips, resource links, and a demonstration video.

Refer to the scenario outline and start with task 1.1 to complete this assessment part.

**Before you begin**

**It is important that you read through the entire assessment** before participating in the interview with the supervisor and commencing with any of the tasks.

Download and unzip the file [ICTPRG302\_AE\_Pro2of3\_Appx](https://share.tafensw.edu.au/share/items/a1e005b9-95b0-4f87-b455-603e78d5dbf0/0/?attachment.uuid=1f727346-f15f-47c8-a21b-92f0a3e3ad0b), which contains:

* accounts.txt (test data)
* Gelos Algorithm Design form
* Gelos Software Design form
* Gelos Software Test Report form

These files will be needed to complete the assessment tasks.

## Scenario

You have been employed as an ICT trainee with [Gelos Enterprises](https://share.tafensw.edu.au/share/items/d0b458dc-3922-409d-b1fe-9a2f785f4a38/0/GelosEnterprises.zip/index.html). To further your training, the company has asked Christina Kaiser, the Software Development Team Leader, to train you in all aspects of programming, and you have been assigned to a new programming project.

Gelos requires a simple login program made up of **3 separate components** that are detailed below. You have been asked to write the code for these components, which will be combined into a single program later. The login program currently stores usernames and passwords in a text file.

* **Component 1** – creating a new user account (registration).  
  This component takes a new username and password from the user and appends them to the existing file (accounts.txt). The user can choose their password with a mixture of characters, but the password must be checked to ensure it meets minimum length requirements.
* **Component 2** – checking a username and password (logging in).  
  This component takes an existing username and password (for a previously registered user) and checks if they match a valid entry in the file (accounts.txt).
* **Component 3** – view existing accounts (displaying users).  
  This component displays a numbered list of the existing user accounts (not including their passwords).

You will need to interview your supervisor, Christina Kaiser, to get all the information you need and clarify the project details.

For this assessment, an interactive conversation featuring Christina Kaiser has been pre-recorded. Refer to **Task 1.1** – **interactive video role-play**.

A data file containing test data has been provided to enable program testing. Inspect the file **accounts.txt** to familiarise yourself with this data. The Gelos Software Test Report document (Program Test Data) has already recorded these details.

Details of what you are being assessed on are available in the [Observation Checklist](#ObsChcklst).

### Assessment Overview

There are several submission points throughout this assessment. This list is an overview of what you will need to complete and submit throughout the assessment:

**Part 1**

* Video (or audio) recording

**Part 2**

* Completed Gelos Algorithm Design document

**Part 3**

* A zip file containing your programming code and the Gelos Software Design Document

**Part 4**

* These assessment pages
* The completed Gelos Software Test Report document
* A zip file containing your testing screenshots.

## Part 1: Define requirements

### Task 1.1 Confirm the project brief and clarify requirements

#### Interactive video role-play

Getting ready for the role-play:

You will need to ask **2 questions** and **respond to one question** from the Team Leader. Be clear and specific in your request for each piece of information and use correct terminology – it is not enough to say, 'What about the password?' and assume they will know what you mean.

People speak at around 140 to 170 words per minute. When asked a question by Christina, your response should be about 1 to 1.5 minutes long. **Be prepared before you start recording**.

Use this table to prepare and write down your questions and responses to remind yourself what to ask or say.

* Look at the partially completed software design document to know what details are missing from the project brief.
* See the table below for a list of what these questions should be about and in what order you should ask them. As this is a pre-recorded video, you must ask the questions and respond **in the same order** as they appear in the table below.

Use this table to prepare and write down your questions to remind yourself of what to ask:

Table 2 Questions template

|  |
| --- |
| Topic of discussion with Christina Kaiser |
| 1. The programming language to be used for the project |
|  |
| 1. The length requirements of the passwords |
|  |
| 1. Explain where you could use loops in your program |
|  |

#### Create your recordings

To complete this task, you must access and view the [**interactive video**](https://share.tafensw.edu.au/share/items/a1e005b9-95b0-4f87-b455-603e78d5dbf0/0/?attachment.uuid=bc33e7a5-80b2-496a-ad54-fb4b0ecbf72c).

The interactive video will have 3 pause points where you will ask your questions and respond to Christina Kaiser. You may submit your questions and response **within one recording** or **separate files**. If in one recording, leave a 10-second gap between each part so the assessor can clearly identify your questions and response.

**Recording process:**

* Activate the [**interactive video**](https://share.tafensw.edu.au/share/items/a1e005b9-95b0-4f87-b455-603e78d5dbf0/0/?attachment.uuid=bc33e7a5-80b2-496a-ad54-fb4b0ecbf72c)**.**
* The video will play, Christina will speak, and a message will appear asking you to record your question or answer. Press pause on the video.
* Use your device to start recording and proceed to record your first response.
* When finished recording, press pause on your recording device.
* Return to the video and press play to continue.
* Repeat this process until you have recorded the two questions and the response to Christina's question.

If you are unhappy with your recordings, restart the interactive video and re-record your questions.

If you are happy with your recordings, save the file and submit your work.

## Submission checklist

Submit the following for marking:

Video (or audio) recording (Part 1)

## Observation checklist

The assessment checklist lists the **requirements for each task** in this assessment as outlined in the student's assessment instructions. The assessor will use this checklist to ensure **all** required tasks have been completed and submitted and provide feedback for each task.

Note that S = Satisfactory and U/S = Unsatisfactory.

Table 3 Checklist

| Task number | Did the student do the following? | S | U/S | Assessor comments  Record your comments in enough detail to demonstrate your judgement of the student's performance against the criteria required. |
| --- | --- | --- | --- | --- |
| **Part 1**  Task 1.1 | Submit a recording of the interactive video.  Use effective questioning and listening techniques to confirm requirements and **articulate complex concepts** using common programming terminology. |  |  | Date of observation: |

## Part 2: Design the application

### Task 2.1: Develop an algorithm

1. Develop an algorithm by creating **pseudocode and a flowchart** to define *Component 2 - checking a username and password* (logging in).
2. Include the following:

* Sequence, selection, and iteration constructs.
* Use of logical operators.

Paste a copy of the pseudocode and the flowchart for *Component 2: Checking a username and password* in the spaces provided in the **Gelos Algorithm Design document**.

You will need to **wait for approval** **of your Algorithm Design** before proceeding to the next part of the assessment.

## Submission checklist

Submit the following for marking:

Completed Gelos Algorithm Design document

## Assessment checklist

The assessment checklist lists the **requirements for each task** in this assessment as outlined in the student's assessment instructions. The assessor will use this checklist to ensure **all** required tasks have been completed and submitted and provide feedback for each task.

Note that S = Satisfactory and U/S = Unsatisfactory.

Table 4 Checklist

| Task number | Did the student do the following? | S | U/S | Assessor comments  Record your comments in enough detail to demonstrate your judgement of the student's performance against the criteria required. |
| --- | --- | --- | --- | --- |
| **Part 2** | **Design the algorithm** |  |  |  |
| Task 2.1 | Develop an algorithm and supply pseudocode and a flowchart for approval.  The algorithm includes sequence, selection, iteration constructs and logical operators. |  |  |  |

## Part 3: Develop the application

Once the supervisor has approved your algorithm, you can create the code for the 3 components.

### Task 3.1: Develop the components

Code the 3 components in the programming language you are learning in this unit using the correct syntax, including:

1. A comment at the very top of the program that includes your name, date, and purpose of the program.
2. Use of appropriate function libraries.
3. Comments throughout your code.
4. Use of sequence, selection, and iteration structures.
5. Use of logical operators.
6. Reading and writing to and from text files (accounts.txt).
7. Use of data structures.
8. Use of string manipulation.

Each component of your program should be fully functional. Check for errors and typos and correct them where necessary.

### Task 3.2: Develop a simple menu

Create a **simple menu system** that incorporates the 3 components and will allow the user to:

1. **Register** (Component 1)
2. **View accounts** (Component 3). Users must successfully **log in** (Component 2) before accessing the account listing.
3. **Exit** the program

### Task 3.3: Document your program

Complete the **Gelos Program Design Form**, where required.

### Task 3.4: Present to supervisor, obtain feedback and sign off

Now that you have designed and created the program, you need to present it to your supervisor for feedback and approval.

1. **Create a zip file** of your code and Program Design Form to present to your supervisor for feedback and sign-off. Include your name in the filename of the zip file (for example, **JBSmith\_Code.zip**)

As a part of this review, your supervisor can ask you specific questions about the code you submitted to identify where you have used each programming technique outlined in Task 3.1.

## Submission checklist

Submit the following for marking:

A zip file containing your programming code and the Gelos Software Design Document.

## Assessment checklist

The assessment checklist lists the **requirements for each task** in this assessment as outlined in the student's assessment instructions. The assessor will use this checklist to ensure **all** required tasks have been completed and submitted and provide feedback for each task.

Note that S = Satisfactory and U/S = Unsatisfactory.

Table 4 Checklist

| Task number | Did the student do the following? | S | U/S | Assessor comments  Record your comments in enough detail to demonstrate your judgement of the student's performance against the criteria required. |
| --- | --- | --- | --- | --- |
| **Part 3** | **Develop the application** |  |  |  |
| **Task 3.1** | **Develop the components** |  |  | Date: |
| **Task 3.1.1** | Add comment to top of code with name, date and purpose of program. |  |  |  |
| **Task 3.1.2** | Use appropriate library functions. |  |  |  |
| **Task 3.1.3** | Clarify the meaning of the code using commenting techniques throughout the code. |  |  |  |
| **Task 3.1.4** | Use selection and iteration constructs using logical operators. |  |  |  |
| **Task 3.1.5** | Use logical operators. |  |  |  |
| **Task 3.1.6** | Use expressions to read and write to and from text files (accounts.txt). |  |  |  |
| **Task 3.1.7** | Use data structures. |  |  |  |
| **Task 3.1.8** | Use string manipulation. |  |  |  |
| **Task 3.2** | **Develop a simple menu** |  |  |  |
| **Task 3.2** | Create a simple menu that includes options to:   1. Register 2. View accounts (after login) 3. Exit the program   Menu code incorporates the 3 components created in Task 3.1. |  |  |  |
| **Task 3.3** | Complete the Gelos Program Design Form as required. |  |  |  |
| **Task 3.4** | Present a zip file containing all required code and the completed Program Design Form. Seek feedback and sign off from the supervisor. |  |  |  |

## Part 4: Modify, test and record results

You will use the **Gelos Software Test Report** document in this part of the assessment.

You need to test your program and document the results.

### Task 4.1: Modify your program and document changes

Once you receive the feedback from the supervisor for your previously submitted program, review your code and make any required changes or modifications.

You have received an additional requirement from the client. They would like to see the total number of users in component 3, using the format: **Total users = X** (where X is the total number of users displayed in the list). Make the changes to your code.

Document the changes made to your code in the **Changes Required from Feedback** section of the Software Test Report.

### Task 4.2: Test and record results

For each test conducted, screenshot the results and update the **Software Test Cases** section of the document with the screenshot filename.

Select **2 login records** from the "Program Test Data" section of the document, and conduct the following tests for each:

1. Correct username and password
2. Correct username with an incorrect password.

When all testing is complete, **create a zip file** containing all your screenshot images (5 in total). Name this file **YourName\_Testing.zip** for inclusion in your final submission.

### Task 4.3: Software evaluation

Now that you have completed your program and testing, you need to evaluate your solution and ensure it meets the specifications requested by the client.

Complete the table with your solution's specifications and evaluations in the **Software Evaluation** section of the **Software Test Report** document.

## Submission checklist

Submit the following for marking:

These assessment pages (Assessment event 2 of 3: Project)

The completed Gelos Software Test Report document

The zip file containing your testing screenshots

The zip file containing your program code

## Assessment checklist

The assessment checklist lists the **requirements for each task** in this assessment as outlined in the student's assessment instructions. The assessor will use this checklist to ensure **all** required tasks have been completed and submitted and provide feedback for each task.

Note that S = Satisfactory and U/S = Unsatisfactory.

Table 4 Checklist

| Task number | Did the student do the following? | S | U/S | Assessor comments  Record your comments in enough detail to demonstrate your judgement of the student's performance against the criteria required. |
| --- | --- | --- | --- | --- |
| **Part 4** | **Modify, test and record results** |  |  | Date: |
| **Task 4.1** | Added "Total users" to the program code.  Completed the Changes Required from Feedback section of the Software Test Report document and code modified as required. |  |  |  |
| **Task 4.2** | Conduct testing on 2 login records for the correct username with correct and incorrect passwords. Provide screenshots in a zip file. |  |  |  |
| **Task 4.3** | Carry out specified evaluations to confirm application meets the initial specifications.   * Component 1 * Component 2 * Component 3 * Menu |  |  |  |

### Additional adhoc question/s asked by the Assessor

The assessor may ask Additional adhoc questions during or after the assessment event. This section provides the assessor opportunity to record these questions and your responses.

1. Assessor question (as required):

[Record your additional questions here]

Student response (as required):

[Record the student response/s]

1. Assessor question/s (as required):

[Record your additional questions here]

Student response/s (as required):

[Record the student response/s]

1. Assessor question/s (as required):

[Record your additional questions here]

Student response/s (as required):

[Record the student response/s]

## External resources – Links and URLs

Long URLs and permalinks are provided for access to content when the assessment is not used digitally, for example, not clickable.

Table 6 Long URLs

| Resource Name | Long URL |
| --- | --- |
| The Learning Bank | https://share.tafensw.edu.au/share/home.do |
| TAFE NSW Assessment Guidelines | https://share.tafensw.edu.au/share/items/d36df03f-9651-4d43-8c9d-a299699e8585/0/?attachment.uuid=30e52f91-8a9f-4df1-bf7f-91168307cfb9 |
| ICTPRG302\_AE\_Pro\_Appx.zip | https://share.tafensw.edu.au/share/items/a1e005b9-95b0-4f87-b455-603e78d5dbf0/0/?attachment.uuid=1f727346-f15f-47c8-a21b-92f0a3e3ad0b |
| Gelos Enterprises | https://share.tafensw.edu.au/share/items/d0b458dc-3922-409d-b1fe-9a2f785f4a38/0/GelosEnterprises.zip/index.html |
| Interactive video – Christina Kaiser | https://share.tafensw.edu.au/share/items/a1e005b9-95b0-4f87-b455-603e78d5dbf0/0/?attachment.uuid=bc33e7a5-80b2-496a-ad54-fb4b0ecbf72c |

This page is not required for online assessment submissions.

### Student assessment declaration

This assessment is my original work and has not been:

* copied from any source without proper referencing
* written for me by any other person except where such collaboration has been approved by a teacher or assessor.

Student signature and date

### Reasonable adjustment

Reasonable adjustment was in place for this assessment event.

If so, please provide details of any reasonable adjustment strategies that were implemented:

[Insert reasonable adjustment strategies]

### Assessment outcome

Satisfactory  Unsatisfactory

Comments

[Insert comments]

Assessor name, signature and date

Student acknowledgement of assessment outcome

[Would you like to make any comments about this assessment?]

Student name, signature and date