|  |  |  |  |
| --- | --- | --- | --- |
| **Qualification details** | | | |
| **Training Package Code and Title** | ICT - Information and Communications Technology (Version 8.1) | | |
| **Qualification National Code and Title** | ICT50220 Diploma of information Technology (Release 2) | **State code** | BGJ4 |
| **Assessment Title** | Assessment Task Two – Part B (Individual Project) | | |
| **Unit National Code & Title** | ICTPRG535 Build advanced user interfaces | | |
| ICTPRG547 Apply advanced programming skills in another language | | |
| ICTICT517 Match ICT needs with the strategic direction of the organisation | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Date Due** | Week Twelve | | **Date Received** | |  | |
| **Student Name** | JACK DU BOULAY | | | | | |
| **Student Declaration** | I declare that the evidence submitted is my own work: | | | | | |
| **Assessor Name** |  | | | | | |
| **Assessment Decision** | Satisfactory | | | Not Yet Satisfactory | | |
| **Assessor Signature** |  | | | **Date** | |  |
| **Is student eligible for reassessment (Re-sit)?** | No | Yes | | **Re-assessment Date:** | | Week Nineteen |

|  |  |  |  |
| --- | --- | --- | --- |
| **Feedback to student** | | | |
| *Via Blackboard (LMS) – Please check [Grade] section.* | | | |
| **Feedback from student** | | | |
| *Via Blackboard (LMS) – Please use [Comment] section during submission.* | | | |
| **Student signature** |  | **Date** |  |

|  |  |
| --- | --- |
| **Assessment Instructions** | |
| **TO THE ASSESSOR** |  |
| Type of Assessment | Individual Project |
| Duration of the assessment | 5 class sessions (Weeks 8-12) |
| Location of assessment | Classroom |
| Conditions | Assessor to ensure that the noise levels, natural interactions and time variances are maintained as it would be in the Software Development industry.  Learners are required to complete the required tasks in class and submit the required documentation electronically via Blackboard |
| Elements and Criteria | As detailed in the assessment plan  You are required to make sure that all students meet the elements, performance criteria and oral communication items as outlined in the provided solution |
| **TO THE STUDENT** |  |
| Purpose of Assessment | You are required to show you can:  ICTPRG5335 Build advanced user interfaces   * Plan and design a UI solution according to organisational requirements, * Apply interactions designs and implement validation requirements against the design plan, * Create and display the UI with graphics according to UI requirements.   ICTPRG547 Apply advanced programming skills in another language   * Code advanced data structures using hashing, sorting and searching algorithms, * Apply third party libraries and communication technologies for data exchange, * Test and evaluate the code to resolve logical and syntactical designs flaws, * Create and document the application according to technical specifications.   ICTICT517 Match ICT needs with the strategic direction of the organisation   * Interpret, analysis and report the strategic organisational plan * Propose and document changes for the implementation of a ICT system * Provide action plan and schedule   The student must demonstrate the ability to complete the tasks outlined in this assessment and is expected to use systematic analytical processes and effect time management to meet the goals/deadlines outlined in the DAP. |

|  |  |
| --- | --- |
| Allowable Materials | Blackboard Topics, SDLC, Weekly readings (PDF), Example programs and Independent Outside of Class Activities |
| Required Resources | Web links and example code can be downloaded from the Blackboard portal.  PC with Notepad++, Visual Studio, GitHub, MSOffice.  Internet Access to GitHub and www.citems.com.au/ |
| Reasonable Adjustment | In some circumstances, adjustments to assessments may be made for you. If you require support for literacy and numeracy issues; support for hearing, sight or mobility issues; change to assessment times/venues; use of special or adaptive technology; considerations relating to age, gender and cultural beliefs; format of assessment materials; or presence of a scribe you need to inform your lecturer. |
| Assessment Submission | All questions and programming activities must be attempted. All written answers must be submitted in this assessment document in the appropriate space.  Use of research tools and peers in formulating answers are acceptable – but work submitted must be your own work.  Final project documentation is to be uploaded to the appropriate area in the Blackboard course created for this unit.  If you are marked as NYS (Not Yet Satisfactory) on your first attempt, you will be provided with another opportunity to re-attempt the assessment. |
| Portfolio Description | A project of web coding tasks and written questions which should be completed in class and finished in the students’ own time on a weekly basis as per the Delivery and Assessment schedule.  Part A  Question 1 – Organisational Analysis  Question 2 – Project Specifications  Question 3 – Version Control  Question 4 – Design Approval  **Part B**  **Question 5 – Third-Party Library**  **Question 6 – Server Application**  **Question 7 – Client Application**  **Question 8 – Testing**  **Question 9 – Demonstration, Feedback and Signoff** |

# Scenario

You are employed as the Senior Programmer with CITE Managed Services, and you have been assigned the Astronomical Processing Project for an organisation called Malin Space Science Systems (MSSS). This project will require the planning and development of a Windows multi-application system that will connect several clients with a remote server. Ensure your development follows an Agile methodology that is recorded and maintained using your GitHub account. The details and criteria are provided in the following paragraphs.

You should consult with the CITE representative (your Lecturer) if you are unsure about any of the problems or questions in this assessment. Your primary research should focus on the resources on the Blackboard LMS and CITE web site, additional information can be collected from the Internet, ensure all sources are referenced in your submission. You must demonstrate your working applications before uploading to Blackboard, your Lecturer (Assessor) will sign off to ensure all the criteria are satisfied.

## Organisational Objectives

The long-term strategic plan is to replace the existing socket-based system with an Inter-Process Communications (IPC) system using Windows Communications Forms technologies. Senior managers at Marlin would like to create a custom third-party library for the new application which can be licenced for use by other organisations. The business plan is to grow the number of locations to include Eastern Europe, Asia and Canada. Therefore, the Astronomical Processing application will be expanded to include additional languages and UI customizations. The new client-server application will utilise the Inter-Process Communications technologies and require a network upgrade to the communications infrastructure to support the application.

## Data Flow Design

The following diagram outlines the current system for the each of the interlocking processes. The console server runs on Machine A and references the standard third-party DLL file. Machines B, C and D represent the three locations in the UK, France and Germany. Each location uses the same Windows Forms (.NET Framework) application, running an English language GUI which cannot be changed.

Diagram

Description automatically generated

## Application Requirements

Malin Space Science Systems requires a .NET Console Application which runs continuously on the main server and provides calculation services to clients connecting via a private network. The calculation services are provided through a custom third-party DLL with specific mathematical formula. The third-party library must be complied as a DLL and added or referenced in the server implementation. The referenced third-party library must be called AstroMath.DLL and have the following four astronomical functions: Star Velocity, Star Distance, Temperature Conversion (Celsius – Kelvin) and Blackhole Event Horizon.

A single client Windows Application (.NET Framework) will connect to the server program using Named Pipes from the Inter-Process Communication technologies. The client/server will utilise the Windows Communications Foundation for communications between the Form client and Console server. The client interface will provide the user with input text boxes for each astronomical calculation and read only output text boxes for the returned values. The astronomical output must be in the correct scientific format, while all input must be fully error trapped to prevent erroneous results. This includes validating input to ensure values are within the correct range for each calculation.

Malin Space Science Systems has three major European operations and requires a language option so the teams at each of these locations can change the language duringAs runtime. The client application runs the same version of the application at all locations; this single client version will be distributed and must be customisable for language and UI display. The three languages are; English (UK), French and German. The teams work 24/7 and require a UI customisation for backgrounds, textboxes and buttons which can be adjusted for different times during a 24-hour period (ie night mode). Therefore, the user can select or click a menu option on the client Form that will change the language for all controls and labels. The user can also select or click a menu option on the client Form that will change the colour of all the Buttons, Form Background and Text Labels. The user can select or click a menu option on the client Form that will open the Color Dialog and allow the user to select a Form background colour. Finally, the user can select or click a menu option on the client Form that will open the Font Dialog and allow the user to select a Form font and size.

This assessment is designed to assess the development of a traditional WinForms Application, not WPF or MAUI.

## Question 5 Third-Party Library

Your first programming task is to create the custom third-party library. Create a DLL project file with the following four mathematical formulas and save the file as AstroMath.DLL. Finally, test and debug the code to ensure the formulas work correctly and return the correct values (use a driver program or the built-in Unit Test). Ensure the code has XML comments which reflect the details of the four mathematical formulas.

**Diagram

Description automatically generated with medium confidenceStar Velocity:** Create a method to measure the Star velocity using the Doppler shift, it should have two input parameters of type double (Observed Wavelength and Rest Wavelength) and return a double which represents the velocity. The Doppler shift of a star can be measured by using the change in wavelength of that object. The formula is

Change in Wavelength () = Observed Wavelength – Rest Wavelength (o)

Speed of Light (C) = 299792458 metres per second

V = velocity in metres per second

**A picture containing clock

Description automatically generatedStar Distance:** Create a method to measure the star distance using the parallax angle, it should have a single input parameter of type double (Arcseconds angle) and return a double. The parallax angle is measured at two different points and works on nearby stars. The method must return a double which is a value in parsecs. The formula is

Parallax Angle (P) in arcseconds (1 arcs = 1/3600)

Distance (D) in parsecs (1 parsec = 3.0857 x 1016m)

Diagram

Description automatically generated**Temperature in Kelvin:** The Kelvin temperature scale is the primary temperature used in science and is easily converted from Celsius. Create a method that has a single input parameter of type double (temperature in Celsius) and returns a double which is the temperature in degrees kelvin. The formula is

Temperature is Celsius (C) = a value must be greater than negative 273.

Temperature in Kelvin (K) = a value greater than zero.

**Event Horizon** (Schwarzschild Radius): Create a method that will return the distance from the centre of a blackhole to the event horizon. The method must have a single input parameter of type double (Blackhole Mass) and return a double which is the event horizon in metres. The formula is

Text

Description automatically generatedGravity Constant (G) = 6.674 x 10-11 m3kg-1s-2

Speed of Light (C) = 299792458 metres per second

Mass of the Blockhole (M) = measured in kilograms (the sun is 2 x 1030 kg).

Schwarzschild radius (R) in meters

## Question 6 Server Application

Develop a console server application which references the third-party library using the Windows Communication Foundation. There are three major tasks in this process.

1. Create the ServiceContract file called “IAstroContract.cs” which will require an Interface that references the AstroMath.DLL and four OperationContract (one for each calculation).
2. Create the server file called “AstroServer.cs” which implements the IAstroContract. Add a new instance of the class library method and then create four methods with the suitable input parameters and return types.
3. Create a ServiceHost and NetNamedPipeBinding in the “Program.cs” file to provide connection for the client(s). Ensure to add a Console.ReadLine to pause the program during operation.

Important: Once your code is error free use the build option to compile a release version for testing. Ensure your application is fully functional and has appropriate error trapping/feedback. Add suitable comments to all code files.

Create a separate console client with test data to establish connectivity and check the basic functionality of the server. Include code into final Server Solution Folder.

## Question 7 Client Application

Develop a Windows Forms Application client so users can send raw data to the server and receive processed information. There are six major tasks in this Question.

1. Create the ServiceContract called “IAstroContract.cs” which will need to be identical to the server without a reference to the AstroMath.DLL.
2. Create a form with suitable components for UI,
   1. Series of textboxes for large numeric data,
   2. A listview/datagrid for display of processed information from the server,
   3. Button(s) to initiate an event and send/receive data.
3. Menu and Button option(s) to change the language and layout for the three different countries.
4. Menu and Button option(s) to change the form’s style (colours and visual appearance).
5. Menu and Button option(s) to select a custom background colour from a colour palette (Color Dialogbox)
6. Menu and Button option(s) to select a custom font and size from a font list (Font Dialogbox)

Important: Once your code is error free use the build option to compile a release version for testing. Ensure your application is fully functional and has appropriate error trapping/feedback. Add suitable comments to all code files.

This assessment is designed to assess the development of a traditional WinForms Application, not WPF or MAUI.

Recommendation: Use the same .Net version for all development work, ensure each process/application is created and built in a separate Solution and VS workspace. Build and test the release version of your code.

## Question 8 Testing

Ensure your code is error free and functions correctly, then test the applications using several different sets of data. During these tests check the returned information is correct and formatted to the appropriate scientific units. Your Test Report must include appropriate evidence that your client/server functions as expected (references to screen captures). Finally, test the Client interface to ensure all the user customizations and globalization setting work. Complete the following Test Report template to answer this question.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Report | | | | | |
| Developer Name | JACK DU BOULAY | | **Date** | 17/09/24 | |
| Astronomical Data Test | | | | | |
| Input Data | Description | Actual Result | | | Pass / Fail |
| 500.1 500 | 1. Star Velocity |  | | | Pass |
| 1000.1 1000 | 2. Star Velocity |  | | | Pass |
| 0.537 | 3. Star Distance |  | | | Pass |
| 0.567 | 4. Star Distance |  | | | Pass |
|  | 5. Temp in Kelvin |  | | | Pass |
| 0 | 6. Temp in Kelvin |  | | | Pass |
| 6.05985810220889885728804e23 | 7. Event Horizon |  | | | Pass |
| 1.209812e33 | 8. Event Horizon |  | | | Pass |
| User Experience Tests | | | | | |
| UI Component | Description | | | | Pass / Fail |
| Background Button | Colour Picker works as intended, background colour changes upon pressing okay. | | | | Pass |
| Star Velocity Button | Observed wavelength rested wavelength need to be filled with valid doubles prior to pressing the button. Once pressed delivers result into the listview that displays the results and the method name | | | | Pass |
| Star Distance Button | Arcsecond angle needs to be filled out with a valid double prior to pressing the button. Once pressed delivers result into the listview that displays the results and the method name | | | | Pass |
| Celsius to Kelvin Button | Celsius needs to be filled out with a valid double prior to pressing the button.  Once pressed delivers result into the listview that displays the results and the method name | | | | Pass |
| Event Horizon button | needs to be filled out with a valid double prior to pressing the button.  Once pressed delivers result into the listview that displays the results and the method name | | | | Pass |
| Clear Data  Button | Once pressed, clears all data in the list view. | | | | Pass |
| Language button | A Once pressed, click on the following language to convert the UI to the desired language. | | | | Pass |
| Connection button | Tests the connection so that it pings the server with a very basic function. If it passes it will say it connected successfully. | | | | Pass |
| Style colour picker button | Colour Picker works as intended, background colour of all the buttons in the UI changes upon pressing okay. | | | | Pass |

## Question 9 Demonstration, Feedback and Signoff

Ensure your code is fully commented with your Name, ID, and Date placed above the main code body of each file. Check all the above documentation has been completed and is ready for inspection. Email your Lecturer (Assessor) and arrange a date and time to demonstrate your working applications, use the following Checklist to ensure you have completed all the assessment criteria.

|  |  |  |  |
| --- | --- | --- | --- |
| Checklist | | completed | |
| **Questions** | | YES NO | |
| Q5 | Third Party Library: all four methods have been created |  |  |
| Q6 | Server Application: a console implementation with appropriate support files |  |  |
| Q7 | Client Application: user can enter raw data and click event to process |  |  |
| Q8 | Testing: All the fields in the Testing Report have been filled in. |  |  |
| Q9 | Demonstration: The IPC functions as required, and all components work correctly. |  |  |

**Note:** All documentation must use the supplied templates/forms.

**Submit the zipped solution folder with relevant documents to Blackboard.**

End of Assessment Task Two – Part B