|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Student Name |  | Student Number | |  |
| Unit Code/s & Name/s | ICTPRG535 Build advanced user interfaces  ICTPRG556 Implement and use a model view controller framework | | | |
| Cluster Name  *If applicable* | Web Interface Cluster | | | |
| Assessment Name | UI Design and Develop Portfolio | Assessment Task No. | | 2 of 2 |
| Assessment Due Date |  | Date submitted | | / / |
| Assessor Name |  | | | |
| **Student Declaration:** I declare that this assessment is my own work. Any ideas and comments made by other people have been acknowledged as references. I understand that if this statement is found to be false, it will be regarded as misconduct and will be subject to disciplinary action as outlined in the TAFE Queensland Student Rules. I understand that by emailing or submitting this assessment electronically, I agree to this Declaration in lieu of a written signature. | | | | |
| Student Signature |  | | Date | / / |

|  |  |
| --- | --- |
| **Instructions to Student** | This assessment requires you to submit a portfolio of tasks. The portfolio assesses skills and knowledge required to develop advanced user interfaces and create and use a model view controller (MVC) framework within a development environment that creates and develops dynamic web architecture by convention  This assessment will work through the planning, design, development and testing stages of developing a front-end Web Application using an MVC framework and advanced user interface techniques based on a provided scenario and documentation + business specifications.  Part 1 - Program and Design Specification  Part 2 - User Interface Design Plan  Part 3 - Application Development  Part 4 - Testing + Validation  Part 5 - Handoff  Throughout these 5 sections, you will be generating documentation, emails and screenshots that must be collected and submitted alongside the source code of the developed application. A template has been provided to assist in collecting these documents:  *WebInterface\_AT2\_Technical\_Report\_<Your Name>.docx*  The generated source code for the developed application will also need to be included in the submission, this should be submitted as a compressed (zipped) file saved as: WebInterface\_AT2\_SourceCode\_<Your Name>.docx  The Files generated throughout the process of completing this portfolio will need to be uploaded to the relevant connect Assignment Folder - Web Interface Cluster Assessment Task 2  **Materials to be Supplied:**  Application design and development scenario  Design / Programming specifications  Technical Report Template  Sample Graphic/s  Multimedia files or links to these files.  **Work, Health and Safety:**  When working with or on computers, ensure that following all WHS guidelines and standards when working with computers. If using physical machines, ensure that you wear closed-in shoes when moving them around and working with them.  **Details of Location:**  TAFE will provide simulated work environment in the classroom all practical activities should be completed in the classroom with teacher/tutor assistance; however, it is possible to complete these tasks on a home virtual network using a freeware/shareware product such as Oracle Virtual Box/VMWare.  **Time Restrictions:**  This is a portfolio assessment designed to take place over 8 weeks. The student is expected to attend in class for 6 hours per week (this includes theory sessions) and should be able to commit up to 3 hours per week in their own time.  **Level of Assistance Permitted:**  Teachers and tutors should be available in class, and accessible by email for students working from home. Staff cannot directly show students answers but guide them to where to go to complete tasks individually. Students with disability will receiver reasonable adjustments.  If you are unable to attend assessment you must notify your teacher before the assessment and supply a doctor’s certificate and approval from the team manager for extension. Reasonable adjustments will be made for students as and when appropriate, after consultation with the Disability and Counselling team. Extra time may be given for Language literacy and numeracy (LLN) or extenuating circumstances - you must see your teacher prior to assessment regarding this.  **Assessment Criteria:**  To achieve a satisfactory result, your assessor will be looking for your ability to demonstrate the following key skills/tasks/knowledge to an acceptable industry standard:  Plan UI design  Create a project using an MVC framework  Develop an application using an MVC framework  Implement interaction techniques  Build customisable and personalised UI  Implement graphics and multimedia  Finalise an MVC project. |
| **Submission details** (if relevant) | Insert your details on page 1 and sign the Student Declaration. Include this form with your submission.  **Due Date:** Week 12  **Naming Convention:**  WebInterface\_AT2\_Technical\_Report\_<Your Name>.docx WebInterface\_AT2\_SourceCode\_<Your Name>.docx  **Evidence Required to be Submitted:**  A completed Technical Report, to provide the required external application documentation and written portfolio evidence  The Source Code of the developed application, this should include the files or scripts required to re-create the database developed with the application, presented in a zip file following the provided naming conventions  This Document, with the date of submission and the candidates name and student number sections completed.  The files above, should be submitted to Connect using the Assessment 1 Assignment Folder, the 3 required files could be provided as individual files in a submission or as a zipped-up collection of the above items.  TAFE Queensland Learning Management System:  Connect url: <https://connect.tafeqld.edu.au/d2l/login>  Username; 9 digit student number  For Password: Reset password go to: <https://passwordreset.tafeqld.edu.au/default.aspx> |
| **Instructions to Assessor** | **Specifications of Assessment:**  There are 5 major activities that must be completed to achieve a satisfactory result for this assessment. Students will be able to resubmit their work a second time if they are not successful on the first attempt.  **Graphic and Multimedia Requirements:**  Various tasks require the student to create graphics and display graphics and multimedia. An initial graphic is included in the files of this assessment, the student could modify this graphic (SVG) inline, or via CSS to meet the graphical creation criteria, Alternatively the student can create their own graphics relevant to the scenario. The multimedia components could include the following:  An embedded YouTube link or video  An embedded video or audio file  Animated images (GIF’s)  Animation of an embedding image via CSS/JS.  **Client Emails and Review:**  Parts 2 and 4 of this assessment require the student to develop emails directed to the client to confirm the direction of planning and development, and to confirm that the developed application meets the clients requirements. Each email should be reviewed by the teacher, and feedback provided on the content presented in the email with the aim of guiding the student towards the criteria defined in the provided scenario, The student must record this provided feedback. If this is being conducted online, the student could be presented with the option of sending the teacher the emails and recording the response emails. |

|  |  |
| --- | --- |
|  | **Equipment or Material Requirements:**  Computer with access to internet  Model view controller framework  Access to relevant integrated development environment  Multimedia tools (for image creation and manipulation)  Access to a system (physical or virtual) to act as the client and server system  Access to organisation code and documentation standards  Access to user requirements  Word processing software  Technical Report Template  Sample Graphic/s  Multimedia files or links to these files.  **Details of Location:**  TAFE will provide simulated work environment in the classroom. the practical activities should be completed in the classroom with teacher/tutor assistance; however, it is possible to complete these tasks on a home virtual network using a freeware/shareware product such as Oracle Virtual Box/VMWare  **Time Restrictions:**  this is a portfolio assessment designed to take place over 8 weeks. The student is expected to attend in class for 6 hours per week (this includes theory sessions), and also should be able to commit up to 3 hours per week in their own time.  **Level of Assistance Permitted:**  Teachers and tutors should be available in class, and accessible by email for students working from home. Staff cannot directly show students answers but guide them to where to go to complete tasks individually. Students with disability will receive reasonable adjustments.  **WHS Requirements:**  All workplace health and safety legislation and policy should apply to the practical activities Interactions - teamwork skills are essential in the IT industry therefore you should work in teams to consult and collaborate on the practical activities. However, each student is to complete the tasks below individually and will be required to show their work (unless indicated).  **Contingencies:**  Reasonable adjustments can be made for students who require variations to assessment conditions. Work Health and Safety: A work health and safety check of the assessment environment is to be conducted prior to the assessment and any hazards addressed appropriately.  **Provided Documents:**  WebInterface\_AT2\_Technical\_Report\_Template.docx  WebInterface\_AT2\_AssessmentFiles.zip  ICTPRG535\_ICTPRG556\_Scenario.docx  **Assessment Conditions:**  Students will need to be provided with access to the following:   1. Physical or virtual systems to act as the development and production environments 2. Multimedia tools for generating CSS animations and creating and editing Image files 3. A solution to apply UI design 4. Physical system with a connected Monitor and Keyboard1 5. Required tools, software and licences required for UI implementation based on programming language and frameworks selected 6. Programming language with an appropriate model view controller framework 7. Integrated development environment (IDE) required to implement MVC frameworks and develop User Interface applications 8. The internet, including connectivity 9. Required hardware, software and applications. |
| **Note to Student** | An overview of all Assessment Tasks relevant to this unit is located in the Unit Study Guide. |

# Assessment 2 - Application and User Interface Development

In this assessment, you are required to participate in the development of a Model View Controller based application, and the planning and implementation of an Advanced User Interface for the application.

You are required to show clear evidence of due process - appropriate planning, gathering requirements, design, development, debugging, testing, and documentation of the MVC application and User Interface, reporting to your project manager on updates required and completion of the project.

This assessment will require the use of both Server-Side and Client-Side programming languages and frameworks to provide the required functionality and interactivity

The assessment is separated into 5 parts:  
**Part 1 - Program and Design Specification**

Analysing the provided specifications and determining application requirements prior to design and development

**Part 2 - User Interface Design Plan**

A detailed plan for the implementation of the Advanced User Interface and all required components

**Part 3 - Application Development**

Creation and development of the MVC Application, including the Implementation of the Advanced User Interface as per the User Interface Design Plan.

**Part 4 - Testing + Validation**

Testing of the developed application and user interface against the confirmed client requirements, requesting client feedback and adjusting where necessary

**Part 5 - Handoff**

Documentation, packaging and sign-off of the project

## Part 1 - Program and Design Specification

1. Review the provided project documents:
2. WI\_Project\_Scenario.docx
3. Programming and Documentation Guidelines.pdf
4. Based on the interpretation of the Project Specifications, develop a detailed **Program Specifications Report** that outlines the following points and provides justification for each choice:
5. Technology and Programming Language/s to be used for the project
6. Development Environment/s and Testing Tools
7. Application Frameworks
8. Target platform and operating system
9. Templates chosen for application development
10. Under a heading **Application Logic Requirements** create a list of all the functionality that the application must provide to the client as per the provided Project specifications, including a diagram of the data models required in the application
11. Created and Document **conceptual designs** or **sketches** of:
12. The planned user interface navigation methods
13. At least 4 different pages or forms that the application may present to the user
14. A colour scheme for the application as per the personalisation requirements in the Project Specifications.

## Part 2 - User Interface Design Plan

1. Prepare a mock email to the client including the content developed in Part 1.4 requesting feedback on the designs and schemes to confirm alignment with project specifications
2. Present this mock email to your teacher, requesting and recording any feedback or adjustments that will align the project with the clients’ requirements
3. Based on the Feedback provided by the Teacher, and the content prepared in Part 1, develop a User Interface Plan that contains the following detail:
4. A UI sketch for each planned page or form
5. A Diagram describing the connection between each form and the flow of the application
6. A description of the **Interaction design pattern** techniques that will be used
7. A list (at least 2) of **Client-Side validation** techniques that will be implemented
8. A description of the planned user interface **customisability**
9. **Graphical** and **Multimedia** requirements as per the provided scenario and planned implementation including:
10. A description of the plan included to modify existing or create new graphics to meet the scenario requirements
11. A description of the multimedia to be included in the application.

## Part 3 - Application Development

1. Create a new Project using the Integrated Development Environment, Language, Framework and Template selected in Part 1
2. Develop and implement required data models ensuring:
3. Each data model aligns with the requirements determined in Part 1.3
4. All data models are compatible with the Tools and technology selected.
5. Create MVC Controllers to suit the Application Logic as determined in Part 1.3. Ensure at least one controller presents endpoints that make use of each of the following methods, accessible via client-side scripts:
6. **HTTP GET**
7. **HTTP POST**
8. **HTTP PUT**
9. **HTTP DELETE**
10. Create the required **views** or pages to align with the UI Design plan created in Part 2, ensuring the views or pages meet the following criteria:
11. At least 2 views or pages that receive and render data dynamically using either partial views, or content updated dynamically via client-side scripting, making use of the endpoints created in 3.3
12. At least 1 view, page or component to act as a settings page or form to manage user interface customisation.
13. At least 1 view or page to:
14. Display included graphics
15. Play included multimedia content
16. At least 1 view or page that demonstrates implementation of the personalisation requirements of the user interface in **Part 1.4.c**
17. At least 2 different interaction design patterns used in application.
18. Implemented Client-Side validation for **all user input scenarios** and as per the User Interface Plan developed in part 2.
19. Usage of CSS concepts such as **margin** and **padding** to correctly lay out user interface components
20. Include screenshots demonstrating each of the points in the tasks above (1,2,3,4) in a Technical Report Document (Template Provided)
21. Implement logic required to connect the Models to Controllers and Controllers to Views as per the program specifications developed in Part 1.

## Part 4 - Testing + Validation

1. Create a testing table, including all the functionality the developed application must present to the client
2. Run and test the application ensuring the following:
3. The steps or process followed to test individual functionality is recorded in the testing table
4. The outcome of each test is recorded in the testing table
5. A summary of all testing, including the tools used to test User Interface and application functionality is included
6. Screenshots of using at least 1 Browser and 1 IDE feature to test the application are included
7. Provide Screenshots of Controller and View communication
8. Run the application and navigate from the launch page to a page that renders dynamic data
9. Provide a screenshot of the code execution paused when the navigation begins
10. Provide a screenshot of the code execution paused before the controller returns model data to the View
11. Provide a screenshot of the rendered dynamic View
12. Create an email to the client to provide details and images of the developed user interface and functionality, include in this email:
13. Screenshots of the developed user interface
14. A description of the personalisation and customisation applied to the developed application
15. A request for feedback on the user interface and user experience
16. Review this email with your teacher, recording any feedback provided
17. Implement any changes required to align the application with the provided feedback, including a description of this in the Technical Report.

## Part 5 - Handoff

1. Create an email to the client to confirm the updates and the improvements to the User Interface or User Experience
2. Create Internal documentation throughout the developed application as per the documentation and business specifications
3. Complete a sign-off sheet to confirm the development of the application has finished, including all items that are being presented to the client.
4. Save the Source code and documentation as per the programming and documentation Guidelines provided.