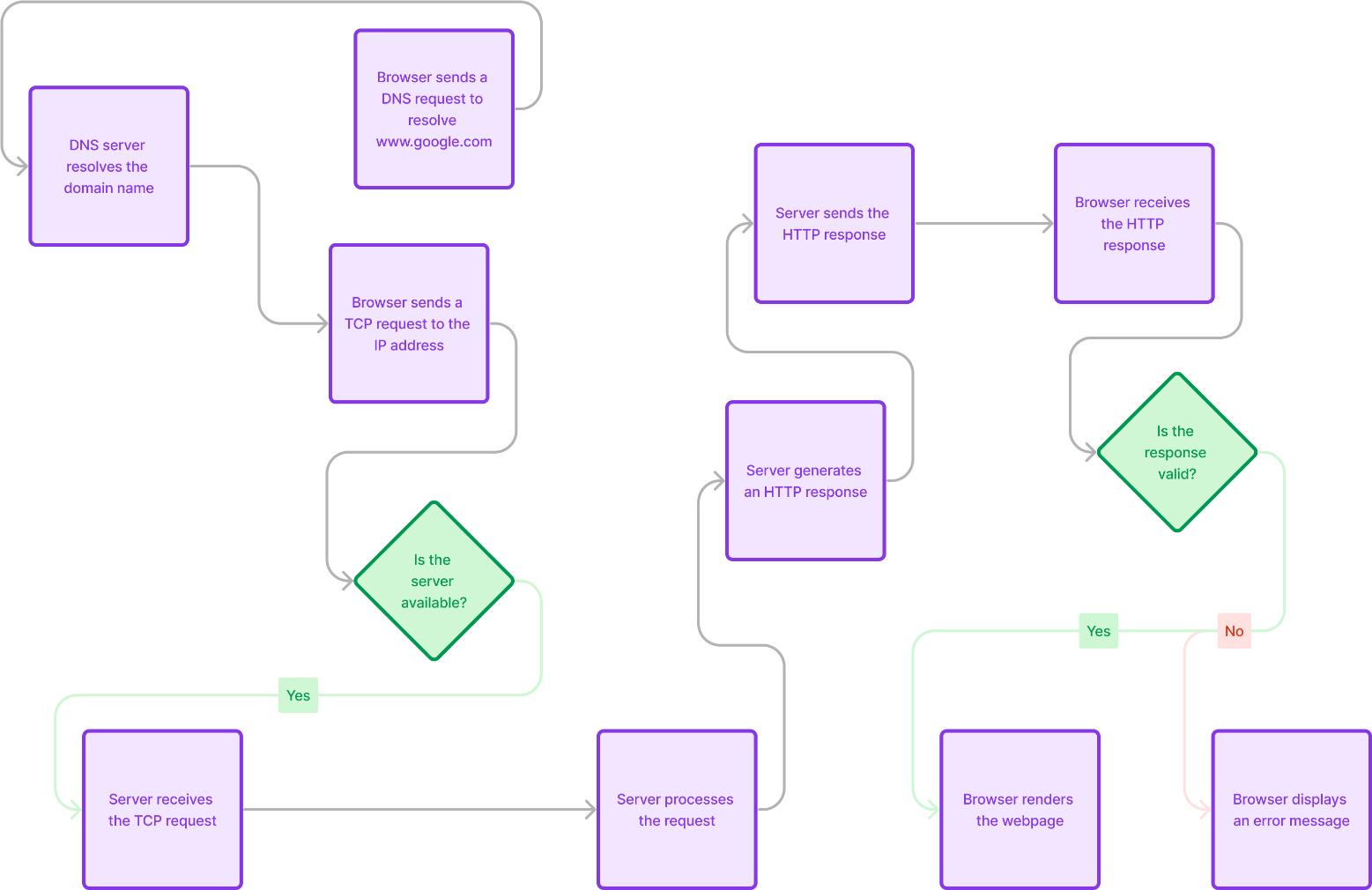
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Student Name | | Carlos Camacho | Student Number | 471450801 | |
| Unit Code/s & Name/s | | ICTPRG535 Build advanced user interfaces  ICTPRG556 Implement and use a model view controller framework | | | |
| Cluster Name  *If applicable* | | Web Interface Cluster | | | |
| Assessment Type | | Case Study  Assignment  Project  Other *(specify)* | | | |
| Assessment Name | | UI Design and Develop Research Questions | Assessment Task No. | | 1 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 | / / | |
| **PRIVACY DISCLAIMER:** TAFE Queensland is collecting your personal information for assessment purposes. The information will only be accessed by authorised employees of TAFE Queensland. Some of this information may be given to the Australian Skills Quality Authority (ASQA) or its successor and/or TAFE Queensland for audit and/or reporting purposes. Your information will not be given to any other person or agency unless you have given us written permission or we are required by law. | | | | | |

|  |  |
| --- | --- |
| Instructions to Student | **General Instructions:**  You are required to answer 18 questions using your own research skills to provide the answers along with examples that illustrate your answers. You are also required to include a list of references at the end which includes all the sources you accessed. Ensure you use an appropriate reference style. There are websites available which generate them for you. One of these is [MyBib - Free Harvard Referencing Generator](https://www.mybib.com/tools/harvard-referencing-generator)  **Acceptable Answers:**  The answers provided to the questions in this assessment must be of a level of academic quality, using correct sentence structure, punctuation and spelling. While no minimum length is stated for each question, it is required that you provide answers that clearly demonstrate your knowledge, supported by an appropriate example or diagram.  An example unacceptable answer to a question:  Q: Describe the process of submitting assessment to Connect:  A: you click upload and select your file.  **Information / Materials Provided:**  A computer with internet access provided in classroom  Word processing software on classroom computers  **Work, Health and Safety:**  The environment should be assessed for safety prior to class. Special considering should be taken regarding potential ICT related hazards such as tripping hazards, electromagnetic radiation, ergonomics and posture.  TAFE Queensland health and safety policies and procedures should be followed at all times.  **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 computer with internet access, web browser and office suits or similar.  **Time Restrictions:**  Students can start completing this assessment from week 2, however students have until week 7 to complete and submit the assessment.  **Level of Assistance Permitted:**  Teachers and tutors should be available in class, and accessible by email for students working from home. Staff cannot directly provide students answers but can guide them to where to go to complete tasks individually.  If you are unable to submit an assessment you must notify your teacher before the assessment due date and supply a doctor’s certificate and an extension approval from the team manager.  Reasonable adjustments will be made for students as and when appropriate, after consultation with the Accessibility and Counselling team. You must see your teacher prior to assessment regarding this.  RPL (Recognition of Prior Learning) is available for this unit. Speak to your teacher/assessor to check if you qualify for RPL.  **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:   1. Object-Oriented programming 2. Client-Side programming 3. HTTP protocol 4. Debugging and testing tools relative to MVC applications 5. Principles of the MVC design pattern 6. HTTP request + response concepts 7. HTTP methods, routing and query parameters 8. Convention vs configuration in application development 9. HTML templates and dynamic rendering 10. Data models and view models 11. Web Design languages: 12. Hypertext Markup Language 13. Cascading Style Sheets 14. JavaScript 15. User Interface prototyping techniques. |

|  |  |
| --- | --- |
|  | **Number of Attempts:**  You will receive up to two (2) attempts at this assessment task. Should your 1st attempt be unsatisfactory (U), your teacher will provide feedback and discuss the relevant sections / questions with you and will arrange a due date for the submission of your 2nd attempt. If your 2nd submission is unsatisfactory (U), or you fail to submit a 2nd attempt, you will receive an overall unsatisfactory result for this assessment task. Only one re-assessment attempt may be granted for each assessment task.  ***For more information, refer to the Student Rules.*** |
| Submission details | Insert your details on page 1 and sign the Student Declaration. Include this template with your submission.  **Due Date:** Week 15  You are required to answer the questions in this document in the space provided under each question and save the document as:  ClusterName\_StudentNameorNumber\_AssessmentNumber.  For example:  WebInterface\_123456789\_AT1  Or  WebInterface\_SteveSmith\_AT1  For re-submissions, an “R” must be added to the file name. For example:  WebInterface\_SteveSmith\_AT1\_R  Assessment to be submitted to the appropriate assessment folder in **Connect.**  TAFE Queensland Learning Management System: Connect url: [*https://connect.tafeqld.edu.au/d2l/login*](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*](https://passwordreset.tafeqld.edu.au/default.aspx) |
| Instructions for the Assessor | To be judged competent in this assessment item the student is required to demonstrate competence in all indicators shown in the marking guide.  Depending on the delivery mode and/or timetable constraints, the Study Guide and the Study Schedule must be customised to suit the mode.  **Specifications of Assessment:**  There are 18 main questions that must all be answered correctly 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.  **Equipment pr Material Requirements:**   * A computer with internet access * Word processing software   **Details of Location:**  TAFE will provide a simulated work environment in the classroom; however, it is possible to complete these tasks on a home virtual network using a computer with internet access.  **Level of Assistance Permitted:**  Teachers and tutors should be available in class, and accessible by email for students working from home. Staff cannot directly provide students answers but can guide them to where to go to complete tasks individually.  **Interactions:**  Teamwork skills are essential in the IT industry; therefore, students should work in teams to consult and collaborate on the practical activities. However, each student is required to complete the tasks individually.  **Contingencies:**  Reasonable adjustments will be made for students as and when appropriate, after consultation with the Accessibility and Counselling team. – You must see your teacher prior to assessment regarding this.  **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. |
| Note to Student | An overview of all Assessment Tasks relevant to this unit is located in the Unit Study Guide. |

1. Provide a Diagram that demonstrates the communication that occurs when a browser requests the webpage: [www.google.com](http://www.google.com) include a description of the technologies used to allow the HTTP request and response to occur.



Browser: The user's web browser, responsible for initiating the request and rendering the webpage.

Operating System (OS): The user's operating system, responsible for resolving the domain name to an IP address and managing the TCP connection.

DNS Server: The Domain Name System server, responsible for translating domain names into IP addresses.

Web Server: The Google web server, responsible for receiving and processing HTTP requests, and sending back HTTP responses.

1. Match the following HTTP Response Codes to the correct description

***Status Code:*** *405, 500, 200, 400, 201, 415*

|  |  |
| --- | --- |
| HTTP Status Code | Description of Status Code |
| 500 | Internal Server Error |
| 400 | Bad Request |
| 405 | Method Not Allowed |
| 200 | Okay |
| 201 | Created |
| 415 | Unsupported Media Type |

1. Describe three different components of a HTTP Request

Request line (method, path, HTTP version):

The first line of an HTTP request is known as the request line. It is divided into three sections:

The method indicates the action that should be performed on the resource. GET, POST, PUT, and DELETE are examples of common methods.

The path identifies the resource that will be accessed. A URL (Uniform Resource Locator) is commonly used to represent it.

The HTTP version shows the version of the HTTP protocol that was utilised in the request. HTTP/1.1 and HTTP/2 are the most commonly used versions.

Headers (Host, User-Agent, Accept, etc):

Headers are pieces of data that convey information about the request or the client making the request. Typical headers include:

Host: The server's hostname to which the request is being sent.

user-agent.: The programme that initiated the request, such as a web browser or a mobile app.

Accept: The content types that the client can accept in response to the request.

The type of data supplied in the request body is specified by the Content-Type header.

The length of the data in the request body (in bytes).

Body (data being sent to server):

An HTTP request's body is optional and comprises data that is transmitted to the server. The HTTP method used determines the type of data in the body. A POST request's body, for example, might contain form data or JSON data.

1. Consider the following webrequest:

<https://www.webaddress.com?firstname=william&lastname=shakespeare>

Identify any provided parameters, including the keys and values and describe this style of providing parameters, including how they may be interpreted by the server.

The provided parameters in the URL are firstname with value william, and lastname with value shakespeare. This style of passing parameters in the query string is common for GET requests. The server can parse the query string to get the key-value pairs.

1. Consider the following Class and method definitions:

|  |  |
| --- | --- |
|  | public class Note  {  public int NoteID { get; set; }  public string NoteText { get; set ; }  }  public ActionResult CreateNote(Note note)  {  Console.WriteLine(note.NoteID);  Console.WriteLine(note.NoteText);  } |

a)If a client was attempting to use the method ‘CreateNote’ defined above, how could they pass the data required for the ‘note’ object, and what is the name of the concept that assists in mapping the input parameters to the Model.

To pass the Note data to CreateNote, the client could send it in the request body formatted as JSON or XML. This would allow the data to be deserialized into the Note object based on matching property names, a concept called model binding.

b)Describe appropriate response codes to send as a response from a method that is creating a new object, including codes that could be used for both success and failure.

For creating a new object, common response codes are 201 Created for success, and 400 Bad Request if there is an issue with the data sent.

1. Describe the concept of routing, in the context of handling a HTTP request to a method in an MVC application and returning HTML content.

Routing involves configuring an application to direct incoming requests to the appropriate controller and action. The controller handles the request, performs application logic, and returns the appropriate HTML view.

1. Describe the purpose of using Models to represent the structure of data in relation to representing data in a view, and receiving data from a view when using HTML template engines

Models represent the application data structures. Using models allows view code to work with domain objects rather than raw data. When submitting data from views, model binding maps form values to model properties.

1. Define the paradigm of ‘Convention over configuration’ and provide an example of where this is beneficial in an MVC application

Means favoring conventions and sensible defaults over explicit configuration. An example is model binding mapping form fields to model properties based on matching names.

1. Describe at least 2 Debugging tools that can help in the development of an MVC application, including the primary function and features provided

Browser developer tools for client-side debugging:

Browser developer tools are a set of tools that are incorporated into online browsers and allow developers to inspect, debug, and optimise web pages. These tools are essential for MVC application development since they provide information about both the client-side and server-side aspects of the web application.

**Key Features:**

HTML, CSS, and JavaScript Inspection: Examine and modify the HTML, CSS, and JavaScript code that renders the web page. This allows for identifying and rectifying errors in the code.

JavaScript Debugging: Execute JavaScript code line by line and inspect variable values. This facilitates debugging logical errors in JavaScript code.

JavaScript Performance Profiling: Measure the performance of JavaScript code and identify performance bottlenecks. This enables optimizing code for better performance.

Error Console: View and analyze error messages and warnings generated during page execution. This helps identify and resolve potential issues.

Emulation and Responsive Design Testing: Emulate different devices and screen sizes to test the web page's responsiveness and cross-browser compatibility.

Debugger in IDE for stepping through server-side code:

An integrated development environment (IDE) has a complete set of software development tools, including a debugger. Debuggers enable engineers to step through code line by line, analyse variable values, and pinpoint the source of bugs. Debuggers are useful for debugging server-side code in MVC applications.

**Key Features:**

Breakpoint Setting: Set breakpoints at specific lines of code to halt execution and examine the program state.

Stepwise Execution: Step through code line by line, controlling the flow of execution and observing the program's behavior.

Variable Inspection: View and modify the values of variables at any point in the code execution. This helps identify data inconsistencies or unexpected values.

Conditional Execution: Define conditions that determine when breakpoints are triggered or when execution is halted, allowing for targeted debugging.

Exception Handling Analysis: Analyze exceptions thrown during program execution to identify and resolve potential errors or unexpected behavior.

1. Describe at least 2 Testing tools and methods that can help in the development of an MVC application, including the primary function and features provided

Unit testing frameworks like NUnit for testing components:

Unit testing is a way of testing individual units of code, such as classes or functions, in software. Unit testing frameworks facilitate the writing and execution of unit tests, making it easier to test programmes efficiently. NUnit is a popular unit testing framework for C#.

Simple and Intuitive API:

NUnit boasts a straightforward and user-friendly API, facilitating the seamless creation of unit tests. Its simplicity enhances the overall ease of use, making it accessible even to those new to the testing framework.

Comprehensive Assertion Library:

A key strength of NUnit lies in its inclusive assertion library. This feature empowers developers to comprehensively validate the behavior of their code, ensuring that it meets the specified criteria and performs as intended.

Test Fixtures for Code Organization:

NUnit supports the use of test fixtures, providing a structured approach to organizing unit tests. These fixtures not only enhance code readability but also allow for the efficient sharing of common setup and teardown procedures among multiple tests.

Versatile Test Runners:

NUnit offers a range of test runners to execute unit tests. Whether through graphical interfaces or console-based options, developers have the flexibility to choose the most suitable method for running their tests, aligning with their preferred workflow.

Seamless Third-party Integrations:

NUnit goes beyond its core functionalities by seamlessly integrating with various third-party tools. This includes compatibility with continuous integration (CI) servers and code coverage tools. Such integrations contribute to a robust testing ecosystem, extending NUnit's utility beyond standalone use.

Integration testing frameworks like Selenium for testing full stack:

Integration testing is a type of software testing that involves testing the interaction of several components of an application. Frameworks for integration testing give tools for modelling user interactions and ensuring that the application functions as expected. Selenium is a popular integration testing framework.

Cross-browser Compatibility: Selenium supports testing across various web browsers, ensuring consistent behaviour across different environments.

Language Support: Selenium provides bindings for various programming languages, including Java, Python, and C#.

Locating Elements: Selenium provides powerful tools for locating and interacting with web page elements, such as input fields, buttons, and links.

Action Simulation: Selenium allows simulating various user actions, such as clicking, typing, and submitting forms.

Assertion Verification: Selenium enables validating application behaviour against expectations using assertions.

1. Create a diagram that clearly demonstrates the architecture of the MVC Design Pattern, describing the connections between the various parts

A diagram of a diagram

Description automatically generated

Model: The model is responsible for managing the application's data. It stores and retrieves data, and it enforces business rules.

View: The view is responsible for displaying the application's data to the user. It receives data from the model and renders it in a way that is understandable to the user.

Controller: The controller is responsible for handling user input and updating the model. It receives input from the view, processes it, and updates the model accordingly.

The connections between the parts of the MVC patter:

The model sends data to the view.

The view sends user input to the controller.

The controller updates the model.

1. Describe the aspects of the MVC design pattern that assist in creating an application that is Scalable, Maintainable, and Reusable:

MVC enables scalability by separating concerns into modular components. It enables maintainability and reusability by encapsulating logic into controllers and using views and models.

1. Describe Client-Side programming including an example language, providing at least one situation where it may be the preferred choice over Server-Side programming.

Client-side programming is the creation of code that is executed on the user's device rather than on a remote server. It is most commonly used to develop interactive and dynamic web pages that respond to user input and give a consistent user experience. Client-side programming languages such as JavaScript, HTML, and CSS are executed by the user's web browser, allowing them to change the content and functionality of the web page without requiring constant server connectivity.

Example:

Consider a website that enables users to build and manage shopping lists. While server-side programming handles data storage and user authentication, client-side programming is essential for creating a responsive and engaging user experience.

When a user adds an item to the shopping list, the client-side code dynamically updates the list, eliminating the requirement for a server round trip.

Real-time Updates: As users add or remove things from the list, the client-side code would update in real-time to provide a consistent user experience.

Client-side programming could offer drag-and-drop capabilities to rearrange items in the list, improving user interactions.

Client-side programming is commonly used when:

Real-time User Interactions: The programme must respond to user input in real-time, such as updating a shopping list or editing a map.

Reduced Server burden: The application tries to reduce server burden by performing lightweight client-side operations, hence enhancing overall performance.

Offline functionality: The programme must be able to work without an internet connection, allowing users to access and interact with content.

1. Describe the purpose of each of the following languages in the context of designing a User Interface for a web application including an example in each language:

a)Hyper Text Markup Language

HTML provides structure and content for webpages. E.g. <h1>Heading</h1>

b)Cascading Style Sheets

CSS controls visual styling of HTML. E.g. h1 { color: blue; }

Cascading Style Sheets (CSS) are a style sheet language used to style the presentation of HTML elements. CSS allows developers to separate the presentation of content from the content itself, making it easier to maintain and control the appearance of web pages. CSS can be used to style a wide variety of aspects of a web page, including:

Colors: Set the colors of text, backgrounds, and other elements.

Fonts: Specify the font family, size, and weight of text.

Margins and padding: Control the spacing around elements.

Borders: Add borders to elements.

Positioning: Control the position of elements on a web page.

Layouts: Create responsive layouts that adapt to different screen sizes.

Example:

body {

font-family: Arial, sans-serif;

font-size: 16px;

margin: 0;

padding: 0;

}

.header {

background-color: #f0f0f0;

padding: 20px;

}

.title {

font-size: 24px;

font-weight: bold;

}

.content {

padding: 20px;

}

c)JavaScript

JavaScript adds interactivity. E.g. button click handler.

JavaScript is a scripting language that allows web page authors to add interactivity. JavaScript can be used to do the following:

Animate components on a web page: Animating elements on a web page can make it more engaging and user-friendly.

Handle user events such as clicks, key presses, and mouse movements: JavaScript can be used to handle user events such as clicks, key presses, and mouse movements. Developers can use this to construct dynamic and responsive web sites.

Send Ajax requests: Ajax (Asynchronous JavaScript and XML) queries enable JavaScript to connect with a server without completely reloading the page. This can be used to update sections of a website without interfering with the user's experience.

Validate user input: Before submitting user input to a server, JavaScript can be used to validate it.

construct interactive games and applications: JavaScript is a strong programming language that can be used to construct complicated games and interactive applications.

Example:

const button = document.getElementById('button');

button.addEventListener('click', () => {

const element = document.getElementById('content');

element.innerHTML = 'Hello, world!';

});

1. Describe the difference between the following User Interface concepts: Wireframe, Mock-up, Prototype. Include in your answer a description of the importance of each concept.

Wireframe: Wireframes are low-fidelity sketches or diagrams that depict the fundamental structure and layout of a web page or application. They are concerned with the general flow of the user experience and the positioning of crucial items rather than the visual details. Typically, wireframes are developed early in the design process in order to quickly receive input and iterate on the design before engaging in more extensive work.

Rapid Iteration

Clear Communication

Focus on Functionality

Mockups: Mockups are high-fidelity renderings of a website or application that give a realistic peek of the finished product. They include visual elements such as colours, fonts, and images, which help designers to fine-tune the visual design and ensure that the product has a consistent and appealing appearance. Mockups are often made after the wireframes and overall design direction have been finalised.

**Visual Appeal**

User Feedback

Marketing and Sales

Prototype: Prototypes are interactive simulations of a web page or application's user experience. They enable people to interact with the product, offering useful feedback on its usability, usefulness, and overall flow. Low-fidelity simulations made with tools like Sketch or InVision to high-fidelity simulations that closely resemble the final product are examples of prototypes.

Usability Testing: Prototypes allow for usability testing, enabling designers to identify and fix usability issues before the product is developed.

1. List and Describe at least 2 Object Oriented Programming languages, including the features of OOP languages that assist in developing User Interface applications.

Object-oriented languages like C# and Java have features like encapsulation, inheritance, and polymorphism that help create reusable UI components.

Java is a widely used, general-purpose OOP language known for its platform independence, robustness, and rich standard library. It's particularly well-suited for developing enterprise applications, including UI applications.

Encapsulation: Encapsulation enables the creation of self-contained modules known as objects, which hide their underlying implementation details while exposing only their public interfaces. In UI development, this increases modularity, reusability, and code maintainability.

Inheritance: Inheritance allows new classes to be created that inherit properties and methods from existing ones. This encourages code reuse and makes it easier to create UI elements with shared functionality.

Polymorphism is the ability of objects of different classes to respond to the same method call in different ways depending on their type. This enables dynamic UI behaviour and makes customisation easier.

C# is another popular OOP language, primarily used for developing Microsoft-based applications. It shares many similarities with Java and is well-suited for developing UI applications, especially for Windows and .NET platforms.

Properties: Properties are useful for data binding and UI manipulation since they give a succinct way to access and edit object attributes.

Interfaces set contracts that classes must follow, fostering loose coupling and making it easier to create UI elements with interchangeable behaviours.

Delegates and Events: Delegates and events allow for asynchronous communication between UI elements, enabling responsive and dynamic UI interactions.