PROCESO DIRECCIÓN DE FORMACIÓN PROFESIONAL INTEGRAL

FORMATO GUÍA DE APRENDIZAJE

IDENTIFICACIÓN DE LA GUIA DE APRENDIZAJE

- Denominación del Programa de Formación: ADSI
- Código del Programa de Formación: 2325230
- Fase del Proyecto (si es formación Titulada): Diseño
- Actividad de Proyecto (si es formación Titulada): (AP5) DETERMINAR LA ESTRUCTURA LÓGICA DEL SISTEMA DE INFORMACIÓN.
- Competencia: DISEÑAR EL SISTEMA DE ACUERDO CON LOS REQUERIMIENTOS DEL CLIENTE
- Resultados de Aprendizaje Alcanzar: **22050103304. Construir el prototipo del sistema de información, a partir del análisis de las características funcionales del sistema en relación con facilidad de manejo, funcionalidad y experiencia del usuario, apoyado en software aplicado según protocolos de diseño.**
- Duración de la Guía: 40 horas

2. PRESENTACION

En esta actividad de aprendizaje se realizará el la planeación del diseño de la interfaz gráfica de usuario que permite definir las acciones que activan la comunicación entre el usuario y el sistema. Para esto se hace necesario conceptualizar sobre principios del diseño, tipos de interfaces, el proceso de construcción de los mismos y algunos ejemplos con herramientas que se encuentran disponibles en el mercado.

La apropiación del conocimiento se realiza con el desarrollo de situaciones puntuales donde es requerida la construcción de interfaces de usuario. El laboratorio incluye la presentación de dos manuales para el uso de las herramientas: Mockups, con las cuales se pueden diseñar interfaces gráficas para aplicaciones software orientadas a diferentes plataformas.

Con la aprehensión y práctica de estos conceptos, podrá diseñar la interfaz gráfica para el sistema de información del proyecto formativo, teniendo en cuenta los requerimientos del sistema, casos de uso, los objetos software y el modelo de la base de datos, insumos que ha definido y construido en las actividades anteriores.

La construcción de la Interfaz gráfica de usuario se inicia también con el reconocimiento de herramientas que facilitan su desarrollo y que proveen diferentes funcionalidades del entorno gráfico muy importantes en las aplicaciones de software que se realizan en la actualidad.

Para llevar a la práctica los conceptos presentados se desarrollará actividades, las cuales ayudarán en la comprensión de la estructura básica y las zonas de una interfaz con HTML5, CSS, JavaScript así como a Frameworks Bootstrap, jQuery.
Posteriormente se realizaran los prototipos de algunas de las interfaces gráficas requeridas para el sistema de Información del proyecto formativo, las cuales deberán aplicar los conceptos vistos.

El concepto "usabilidad" demanda al desarrollador, esfuerzos orientados para que toda persona independiente de algunas limitaciones, pueda acceder y beneficiarse cómodamente de los contenidos del programa objeto. Esto es asunto de responsabilidad social. La usabilidad de los objetos construidos desde la perspectiva ética se presenta como material de estudio y se le invita a explorarlo y reflexionar acerca de los elementos que presenta. Complementa su lectura una actividad orientada a identificar errores comunes fácilmente detectables en algunos programas o sitios web para así evitarlos en los diseños propios.

Actividad Reflexion inicial

Responder las siguientes preguntas:

- ¿Qué es una interfaz de usuario?
- ¿Qué se debe tener en cuenta para desarrollar una muy buena interfaz de usuario?
- ¿Cómo se construyen los sitios web?
- ¿Qué es usabilidad?
- ¿Cuál es la importancia de la interfaz de usuario en las aplicaciones web?

Enlaces de Apoyo:

https://www.youtube.com/watch?v=WqWv_mpuvaI

https://www.youtube.com/watch?v=Fxt3G0lDdJI

Duración: 2 Hora

Actividad Conceptos básicos en la web.

Usted como aprendiz deberá realizar la investigación de los siguientes conceptos:

¿Qué es FTP?
¿Qué es un hosting?
¿Qué es un dominio?
¿Qué es un servidor Web?
¿que son mapas de navegación, como se crean y como se usan?
¿ que es un prototipo Web?
¿qué es Diseño Web, Diseño gráfico y Desarrollo Web?, mencionar la diferencia entre los tres últimos (Diseño Web, Diseño gráfico y Desarrollo Web)

Duración: 3 Horas

Actividad grupo Proyecto:

Dado su proyecto de formación y teniendo en cuenta los requisitos de su aplicación web, realizar lo siguiente:

1. Definir la Concepción de la aplicación web:
   - ¿Qué problema soluciona?
   - ¿Cuál es su objetivo?
   - Cuál es su valor agregado/diferenciador?
2. Concepción del usuario de la aplicación (crear minino 3 proto-persona):
   - Cara y nombre
   - Información demográfica/comportamiento
   - Puntos de dolor y necesidades
   - Posibles soluciones
3. Concepción de funcionalidades de la aplicación web:
   - Función-para que sirve
   - Categorizarlas entre principales y secundarias
   -

- Elabore un informe en Word con los puntos pedidos en el taller. Posteriormente se socializará en grupo.
- Para la creación de las proto-personas hacer uso de herramientas online

Links: material de apoyo:

https://edit.org/es/blog/user-persona-ejemplos-plantillas-editables-online#ini

https://www.interware.com.mx/blog/como-creo-una-proto-persona-para-definir-a-mi-usuario-final

1.16 Estructuras de control (`if` , `else` , `switch`)

1.16.1 `if`

La estructura de control `if` permite la ejecución condicional de fragmentos de código PHP.

Sintaxis:

```php
<?php

if (condicion_es_cierta) {
    bloque_de_sentencias
}

?>
```

Ejemplo:

El siguiente ejemplo mostraría `a es mayor que b` si la variable `$a` es mayor que `$b` :

```php
<?php

$a = 10;
$b = 5;

if ($a > $b) {
    echo "a es mayor que b";
}

?>
```

1.16.2 `if - else`

Sintaxis:

```php
<?php

if (condicion_es_cierta) {
    bloque_de_sentencias_1
} else {
    bloque_de_sentencias_2
}

?>
```

Ejemplo:

El siguiente ejemplo puede mostrar `a es mayor que b` si la variable `$a` es mayor que `$b` o `a NO es mayor que b` en caso contrario.

```
<?php

$a = 1;
$b = 5;

if ($a > $b) {
    echo "a es mayor que b";
} else {
    echo "a NO es mayor que b";
}

?>
```

1.16.3 `elseif / else if`

Sintaxis `elseif` :

```
<?php

if (condición_1) {
    bloque_de_sentencias_1
} elseif (condición_2) {
    bloque_de_sentencias_2
} else {
    bloque_de_sentencias_3
}

?>
```

Sintaxis `else if` :

```
<?php

if (condición_1) {
    bloque_de_sentencias_1
} else if (condición_2) {
    bloque_de_sentencias_2
} else {
    bloque_de_sentencias_3
}

?>
```

Ejemplo:

```
<?php
```

```php
$a = 1;
$b = 1;

if ($a > $b) {
    echo "a es mayor que b";
} elseif ($a == $b) {
    echo "a es igual que b";
} else {
    echo "a es menor que b";
}

?>
```

1.16.4 `switch`

Sintaxis:

```php
<?php

switch ($variable) {
    case valor1:
        bloque_de_sentencias_1
        break;

    case valor2:
        bloque_de_sentencias_2
        break;

    default:
        bloque_de_sentencias_3
}

?>
```

Esta estructura de control es equivalente a:

```php
<?php

if ($variable == valor1) {
    bloque_de_sentencias_1
} elseif ($variable == valor2) {
    bloque_de_sentencias_2
} else {
    bloque_de_sentencias_3
}

?>
```

Ejemplo:

```php
<?php
```

```php
$numero = 2;

switch ($numero) {
    case 1:
        echo "La variable es igual a 1";
        break;

    case 2:
        echo "La variable es igual a 2";
        break;

    default:
        echo "La variable es un número distinto a 1 y 2";
}

?>
```

1.16.5 Ejercicios

1. Escribe un script que simule el comportamiento de lanzar una moneda al aire y muestre una imagen con la cara o la cruz de la moneda.

Notas:

- Documentación de la función `rand` .
- Documentación de la estructura de control `if` .
- Documentación de la estructura de control `else` .
- Documentación del elemento de imagen `<img>` en HTML.

2. Escribe un script PHP que genere un número aleatorio entre 1 y 10, simulando una nota numérica y muestre un mensaje indicando la calificación obtenida teniendo en cuenta los siguientes rangos:

- Insuficiente: [0, 5)
- Suficiente: [5, 6)
- Bien: [6, 7)
- Notable: [7, 9)
- Sobresaliente: [9, 10]

Notas:

- Documentación de la estructura de control `if` .
- Documentación de la estructura de control `else` .

3. Escribe un script PHP que genere un número aleatorio entre 1 y 7, y muestre un mensaje indicando a qué día de la semana corresponde. Por ejemplo, 1 sería lunes, 2 martes, etc.

Notas:

- Documentación de la estructura de control `switch`.

4. Escribe un script PHP que realice la simulación de lanzar un dado y muestre una imagen con un valor aleatorio enre 1 y 6. Resuelva el ejercicio utilizando la estructura de control `if - else`.

Notas:

- Documentación de la función `rand`.
- Documentación de la estructura de control `if`.
- Documentación de la estructura de control `else`.
- Documentación del elemento de imagen `<img>` en HTML.

Imágenes: 1, 2, 3, 4, 5 y 6.

5. Escribe un script PHP que realice la simulación de lanzar un dado y muestre una imagen con un valor aleatorio entre 1 y 6. Resuelva el ejercicio utilizando la estructura de control `switch`.

Notas:

- Documentación de la función `rand`.
- Documentación de la estructura de control `switch`.
- Documentación del elemento de imagen `<img>` en HTML.

Imágenes: 1, 2, 3, 4, 5 y 6.

6. Escribe un script PHP que realice la simulación de lanzar un dado y muestre una imagen con un valor aleatorio enre 1 y 6. Resuelva el ejercicio sin utilizar las estructuras de control `if - else` y `switch`.

7. Escribe un script PHP que realice la simulación de lanzar dos dados y muestre una imagen con los valores obtenidos en cada uno de los dados.

1.17 Bucles (`for`, `while`, `do-while`)

1.17.1 `for`

Sintaxis:

```
for (expr1; expr2; expr3) {
    sentencias;
}
```

Ejemplo:

El siguiente ejemplo muestra los números del 1 al 10.

```
<?php

for ($i = 1; $i <= 10; $i++) {
```

```
    echo $i;
    echo "<br>";
}

?>
```

Ejemplo:

El siguiente ejemplo muestra los números del 10 al 1.

```
<?php

for ($i = 10; $i >= 1; $i--) {
    echo $i;
    echo "<br>";
}

?>
```

1.17.2 `while`

Sintaxis:

```
while (condicion_es_verdadera) {
    sentencias;
}
```

Ejemplo:

El siguiente ejemplo muestra los números del 1 al 10.

```
<?php

$i = 1;
while ($i <= 10) {
    echo $i;
    echo "<br>";
    $i++;
}

?>
```

Ejemplo:

El siguiente ejemplo muestra los números del 10 al 1.

```
<?php

$i = 10;
```

```php
while ($i >= 1) {
    echo $i;
    echo "<br>";
    $i--;
}

?>
```

1.17.3 `do - while`

```php
do {
    sentencias;
} while (condicion_es_verdadera)
```

Ejemplo:

El siguiente ejemplo muestra los números del 1 al 10.

```php
<?php

$i = 1;
do {
    echo $i;
    echo "<br>";
    $i++;
} while ($i <= 10);

?>
```

Ejemplo:

El siguiente ejemplo muestra los números del 10 al 1.

```php
<?php

$i = 10;
do {
    echo $i;
    echo "<br>";
    $i--;
} while ($i >= 1);

?>
```

1.17.4 Ejercicios bucle `for`

1. Escribe un script PHP que muestre los números del 1 al 10 en una tabla de una fila y 10 columnas. Utiliza un bucle `for`

Notas:

- Documentación del bucle `for` .
- Tablas HTML.

2. Escribe un script PHP que muestre los números del 1 al 10 en una tabla de una columna y 10 filas. Utiliza un bucle `for` .

3. Escribe un script PHP que muestre en una tabla los números pares que existen entre 1 y 100. Utiliza un bucle `for` .

4. Escribe un script PHP que muestre la tabla de multiplicar de un número aleatorio. Utiliza un bucle `for`

5. Escribe un script PHP que muestre las tablas de multiplicar del 1 al 10. Utiliza un bucle `for`

1.17.5 Ejercicios bucle `while`

1. Escribe un script PHP que muestre los números del 1 al 10 en una tabla de una fila y 10 columnas. Utiliza un bucle `while`

Notas:

- Documentación de la función `while` .
- Tablas HTML.

1.17.6 Ejercicios bucle `do - while`

1. Escribe un script PHP que muestre los números del 1 al 10 en una tabla de una fila y 10 columnas. Utiliza un bucle `while`

Notas:

- Documentación de la función `do-while` .
- Tablas HTML.

1.19 Ejercicios - Funciones

1. Escribe una función que reciba un número como parámetro de entrada y que imprima su tabla de multiplicar.

Notas:

- Funciones definidas por el usuario en PHP.

2. Escribe una función que reciba dos parámetros de entrada (inicio y fin) y que imprima las tablas de multiplicar entre esos dos números. Utilice la función del ejercicio anterior.

3. Escribe una función llamada `inicializar_array` que reciba tres parémetros llamados `numero_de_elementos`, `min` y `max`, y que devuelva un array de números enteros comprendidos entre los valores `min` y `max`. El número de elementos que contiene el array será el especificado en el parámetro de entrada `numero_de_elementos`

Notas:

- Cómo devolver valores en una función PHP.

4. Escribe una función llamada `calcular_media` que reciba un array como parámetro de entrada y que devuelva la media de todos los valores que contiene.

5. Escribe una función llamada `calcular_maximo` que reciba un array como parámetro de entrada y que devuelva cuál es el máximo valor del array.

6. Escribe una función llamada `calcular_minimo` que reciba un array como parámetro de entrada y que devuelva cuál es el mínimo valor del array.

7. Escribe una función llamada `imprimir_array` que reciba un array como parámetro de entrada y muestre su contenido en una tabla con dos columnas. La primera columna mostrará la posición del array y la segunda el valor que hay en esa posición.

8. Crea un archivo llamado `funciones.php` que contenga todas las funciones creadas en los ejercicios anteriores. Escriba un script PHP que incluya el archivo `funciones.php` y haga uso de cada una de ellas.

Notas:

- Documentación de la sentencia `include` en PHP.
- Documentación de la sentencia `require` en PHP.

1.22 Ejercicios - Objetos

Recursos:

- Documentación oficial sobre clases y objetos en PHP.
- Introducción a la Programación Orientada a Objetos (POO) en PHP por Diego Lázaro.

1. Define una clase llamada `Persona` que cumpla los siguientes requisitos:

- Debe tener las siguientes **propiedades privadas**:
  - `nombre`
  - `apellido1`
  - `apellido2`
  - `edad`
- Debe tener un **constructor** que permita inicializar los valores de las propiedades.
- Define tener **métodos públicos** `get` y `set` para cada una de las propiedades.
- Debe incluir un **método público llamado** `imprimir` que muestre todas las propiedades del objeto.

Una vez definida la clase, relice dos instancias y utilice todos los métodos que ha creado.

Taller introducción a Java y POO

De forma Individual, resolver las siguientes preguntas de manera individual:

1) ¿Qué es java?

2) ¿Qué es JDK?

3) Tipos de licencia en java.

4) Cuáles son los principales IDE en java.

5) ¿Qué es una clase y que elementos la componen?

6) Que es un objeto?

7) Que es una instancia?

8) tipos de datos en java.

9) Buscar un ejemplo de Encapsulamiento en java.

10) Buscar un ejemplo de Herencia en java.

11) Buscar un ejemplo de Polimorfismo en java.

12) Realizar un mapa conceptual sobre La Programación Orientada a Objetos

13) Realizar un mapa mental sobre las principales características de Java.

Tiempo 5 horas

Evidencia: Documento en Word o PDF y subirlo a la tarea creada en Territorium.