PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL
FORMATO GUÍA DE APRENDIZAJE
GUÍA 6. DIAGRAMA DE CLASES UML Y DESARROLLO BACKEND

IDENTIFICACIÓN DE LA GUIA DE APRENDIZAJE

- Denominación del Programa de Formación: Técnico en programación de software.
- Código del Programa de Formación: 233104 v1.
- Nombre del Proyecto: Desarrollo de aplicaciones de software para el sector empresarial.
- Fase del Proyecto: Fase 3 - Ejecución del Proyecto (Ejecución).
- Actividad de Proyecto: Actividad 5: desarrollar el software a partir del diseño y la arquitectura propuesta.
- Competencia: 220501097 - implementar la solución de software de acuerdo con los requisitos de operación y modelos de referencia.
- Duración de la Guía: Total 240 horas, trabajo directo 192 horas, 48 horas de trabajo independiente.

2. PRESENTACIÓN

2.1 El diagrama de clases UML y la Programación Orientada a Objetos:

El diagrama de clases UML desempeña un papel fundamental en la programación orientada a objetos al proporcionar una representación visual de la estructura y las relaciones entre las diferentes clases de un sistema, su importancia radica en su capacidad para simplificar y clarificar la complejidad del Software a construir por parte del cliente, los diseñadores y los desarrolladores. El diagrama de clases ofrece una vista panorámica de las clases, atributos, métodos y su interacción, además, sirve como guía invaluable durante todas las fases del ciclo de vida del Software, desde la conceptualización hasta la implementación y el mantenimiento. Para concluir, el uso efectivo de los diagramas de clases UML fomenta un diseño sólido y coherente, promoviendo la eficiencia en el desarrollo de aplicaciones orientadas a objetos.



Fig. 1. La importancia de planear antes de construir. Tomada de: https://www.todoparatucasa.mx/wp-content/uploads/2019/08/Home_Construction__Romolo_Tavani_-_Fotolia-1024x682-1024x585.jpg

2.2 El Patrón MVC (Modelo Vista Controlador) y POO (La programación orientada a objetos)

En la presente guía se va a trabajar la implementación de back-end empleando MVC en PYTHON. Esta guía abordará el desarrollo del sistema de información usando programación en Modelo Vista Controlador (MVC), aplicando la conexión a la base de datos MySQL, login y finalmente crear registros, actualización, borrado y consultas de registros de datos. Se utilizará FLASK, Bootstrap y HTML5 para el diseño web en la cual será desarrollado de manera definitiva su proyecto formativo.

Pero antes de continuar, se debe tener presente que esta guía se basa en la Programación Orientada a Objetos (POO), estilo de programación que se basa en el concepto de clases y objetos.

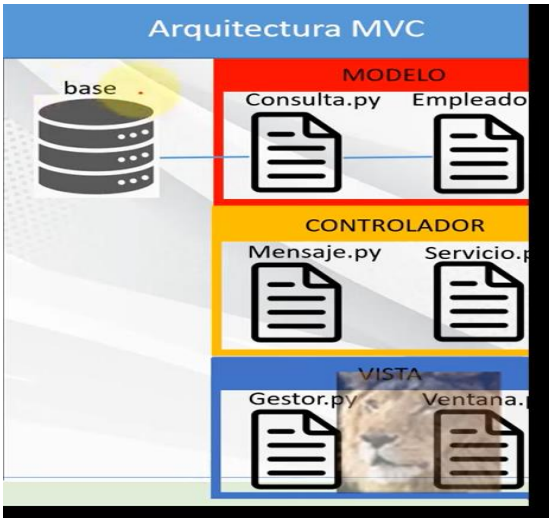
Este tipo de programación se utiliza para estructurar un programa de software en piezas simples y reutilizables que serán los planos base de código y que llamaremos clases, las cuales permitirán crear instancias individuales que llamaremos objetos.

Ejes temáticos:

Programación orientada a objetos, conexión a base de datos, arquitectura MVC, login, control de rutas, control de actores, funciones CRUD en MVC, validaciones de campos vacíos, longitud de campos y tipos de datos.

Está basada en unas características conceptuales como lo son:

- Conceptos básicos de Model-View-Controller
- Conceptos básicos de enrutamiento
- Validación de formularios
- Realización de consultas de base de datos básicas con el "Generador de consultas"



En cuanto al MVC Es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos (Modelo, Vista y Controlador). El Patrón MVC se ve frecuentemente en aplicaciones Web, donde la Vista es la página HTML y el código que provee de datos dinámicos a la página; el Modelo es el Sistema de Gestión de Base de Datos y la Lógica de negocio; el Controlador es el responsable de recibir los eventos de entrada desde la Vista.

Figura 2. http://rodrigogr.com/blog/modelo-vista-controlador/

El Modelo Vista Controlador (MVC) es una forma de estructurar las aplicaciones informáticas, especialmente las páginas web y aplicaciones móviles, que hace que sean más fáciles de entender y desarrollar. Consiste en dividir el software en tres partes principales:

El Modelo: se encarga de acceder a los datos, como una base de datos. Aquí se define la información que maneja el sistema.

La Vista: presenta la información al usuario de forma visual. Por ejemplo, serían las páginas web que ve el usuario.

El Controlador: es el intermediario entre el Modelo y la Vista, gestionando el flujo de información entre ellos y las transformaciones para adaptar los datos a lo que necesita cada uno. Recibe los eventos del usuario en la Vista para realizar las acciones necesarias en el Modelo.

De esta forma, si queremos cambiar cómo se ve la aplicación, sólo tenemos que tocar la Vista. Si cambia la estructura de los datos, trabajamos en el Modelo. El Controlador se encarga de unir ambas partes.

Esta separación en MVC hace más sencillo el trabajo en equipo, testing y mantenimiento de aplicaciones grandes. Cada programador puede enfocarse en una pieza distinta. También es más fácil reutilizar cada componente en nuevos proyectos.

EL lenguaje PYTHON se destaca por Automatización de servicios, desarrollo de software, análisis y ciencias de datos, una excelente rama para Inteligencia Artificial.

3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

3.1. ACTIVIDAD DE REFLEXIÓN INICIAL: Toda persona, animal o cosa puede describirse a partir de sus características un ejemplo básico para una persona podría tenerse en cuenta su Numero de Documento, Nombre, Apellido, Dirección, Teléfono, Color de Piel, Altura. También se podría tener en cuenta sus comportamientos, podemos tomar como ejemplo a un celular el cual puede tener la función de Llamar, Colgar, Tomar Una Foto, Encender Y Apagar. Reúnase en grupo de 3 Aprendices y escojan 3 objetos de la vida diaria y mencione 5 Características y 4 comportamientos por cada uno de ellos. Esta actividad hace parte del entregable EVIDENCIA GUÍA DE APRENDIZAJE 6.

Guardar actividad como: Evidencia_Aprendizaje_Reflexion_3_1.pdf

Ambiente Requerido: Ambiente de formación con equipos de cómputo y acceso a internet.
Materiales y/o Recursos: Libreta de apuntes por aprendiz.
Tiempo para la actividad: 1 hora.

3.1.1 Actividad de Aprendizaje 6_1: Interpretar el Diagrama de Clases partiendo del conocimiento previo.

Antes de continuar, de manera individual, le invitamos a dar lectura al archivo 06_Introduccion_Diagrama_Clases ubicado en MATERIAL DE APOYO APRENDICES / Diagrama de clases / 06_Introduccion_Diagrama_Clases.docx – UML e introducción al diagrama de clases, a partir de la lectura señalada, diseñe un mapa mental o una infografía en la que se pueda establecer de manera breve y concreta la importancia y aspectos principales a tener en cuenta al emplear el paradigma de Programación Orientada a Objetos – POO. Esta actividad hace parte del entregable EVIDENCIA GUÍA DE APRENDIZAJE 6. Una vez terminada la actividad, el instructor dará las indicaciones para que sea socializada con todo el grupo de aprendices.

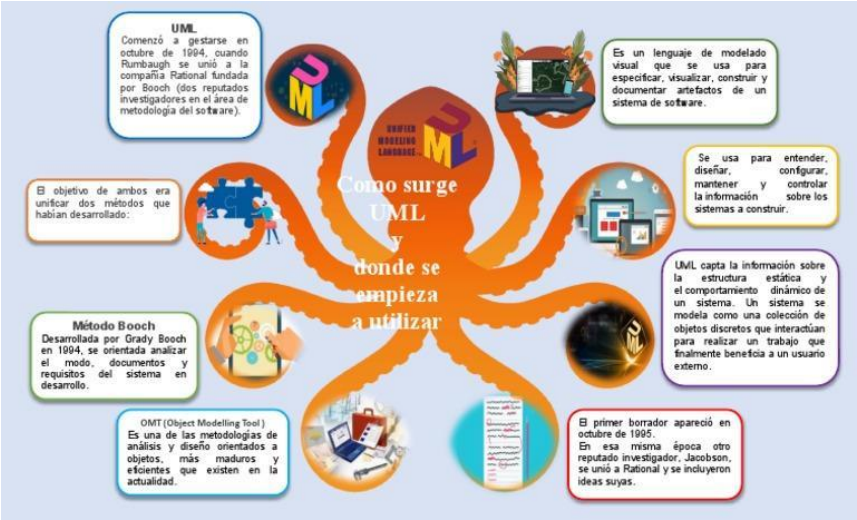


Fig. 3. Mapa mental UML. Tomada de: https://imgv2-2-f.scribdassets.com/img/document/631239330/original/6142bbfa17/1698782348?v=1

Guardar actividad como: Evidencia_Aprendizaje_3_1_1.pdf

Ambiente Requerido: Ambiente de formación con equipos de cómputo y acceso a internet.
Materiales y/o Recursos: Libreta de apuntes de cada aprendiz.
Tiempo para la actividad: 1 hora.

3.1.2 Actividad de Aprendizaje 01_2: Reconocer los componentes del Diagrama de Clases según estándares UML.

Ahora, le invitamos a dar lectura al archivo 06_Elementos_Diagrama_Clases ubicado en MATERIAL DE APOYO APRENDICES / Diagrama de clases / 06_Elementos_Diagrama_Clases.docx – Diagrama de clases y sus elementos, a partir de la lectura señalada, de manera individual, le invitamos a retomar el listado de características y comportamientos de la sección Reflexión Inicial, en el que seleccionó una persona que vive con usted, su mejor amigo o amiga y su mascota, cree las clases necesarias que permitan crear objetos tipo familiar, mejor amigo y mascota, recuerde, debe emplear la estructura adecuada para definir clases. Esta actividad hace parte del entregable EVIDENCIA GUÍA DE APRENDIZAJE 6. Una vez terminada la actividad, el instructor dará las indicaciones para que sea socializada con todo el grupo de aprendices.

NOMBRE_CLASE
[visibilidad] nombre_atributo [: tipo_dato] [= valor_inicial]
[visibilidad] nombre_metodo [(lista_parametros)] [: valor_devuelto]

Guardar actividad como: Evidencia_Aprendizaje_3_1_2.pdf

Ambiente Requerido: Ambiente de formación con equipos de cómputo y acceso a internet.
Materiales y/o Recursos: Libreta de apuntes de cada aprendiz.
Tiempo para la actividad: 1 hora.

3.1.3 Actividad de Aprendizaje 01_3 Adicionar relaciones al Diagrama de Clases.

Para esta actividad, le invitamos a dar lectura al archivo 06_Relaciones_Diagrama_Clases ubicado en MATERIAL DE APOYO APRENDICES / Diagrama de clases / 06_Relaciones_Diagrama_Clases.docx – Relaciones en el diagrama de clases, a partir de la lectura señalada, de manera individual, le invitamos a definir las clases y relaciones pertinentes para los siguientes enunciados, adicione el tipo de relación correcta, su descripción y multiplicidad:

a) Existen 3 tipos de ropa a registrar, ropa deportiva, formal o casual.
b) El cliente realiza reservación.
c) El hotel está compuesto por habitaciones.
d) El colegio está compuesto por estudiantes.
e) Existen 3 tipos de usuario, administrador, docente y estudiante.
f) El docente registra notas.

Esta actividad hace parte del entregable EVIDENCIA GUÍA DE APRENDIZAJE 6. Una vez terminada la actividad, el instructor dará las indicaciones para que sea socializada con todo el grupo de aprendices.

Guardar actividad como: Evidencia_Aprendizaje_6_1_3.pdf

Ambiente Requerido: Ambiente de formación con equipos de cómputo y acceso a internet.
Materiales y/o Recursos: Libreta de apuntes de cada aprendiz.
Tiempo para la actividad: 1 hora.

3.1.4 Actividad de aprendizaje 01_6 Identificar las clases y restricciones de un Diagrama de Clases.

Para finalizar esta sección de la guía de aprendizaje, de manera individual, le invitamos a ver el video Diagrama_de_clases en formato mp4, que se encuentra en la carpeta MATERIAL DE APOYO APRENDICES / Diagrama de clases / Diagrama_de_clases.mp4, al finalizar el video, desarrolle una nube de palabras donde pueda de manera gráfica y breve incluir los términos e ideas más importantes a tener en cuenta para el adecuado diseño del diagrama de clases UML. Esta actividad hace parte del entregable EVIDENCIA GUÍA DE APRENDIZAJE 6. Una vez terminada la actividad, el instructor dará las indicaciones para que sea socializada con todo el grupo de aprendices.



Fig. 4. Ejemplo Nube de palabras. Tomada de: https://es.123rf.com/photo_25274347_una-nube-de-palabras-de-software-de-dise%C3%B1o-de-art%C3%ADculos-relacionados.html

Guardar actividad como: Evidencia_Aprendizaje_6_1_4.pdf

Ambiente Requerido: Ambiente de formación con equipos de cómputo y acceso a internet.
Materiales y/o Recursos: Libreta de apuntes de cada aprendiz
Tiempo para la actividad: 1 hora.

3.1.5 Actividad de Ejercicios iniciales sobre conceptos de Programación básica en Python

- Aprendiz usted Deberá ingresar al portafolio y consultara el archivo **6_1_5_.Qué es Python.docx** realizara un mapa conceptual con lo mas relevante que encuentre en el texto y lo guardara con el nombre "**Aprendiendo_Python"** a tu portafolio.

3.1.6 Analizar la importancia de la programación orientada a objetos y la importancia de los requisitos para la aplicación.

Descripción de las actividades:

- Iniciaremos con la lectura individual del material de apoyo el archivo "6_2_2_1_Breve_Explicacion_Python**.pdf**" el cual se debe complementar con una breve búsqueda en internet.
- Posteriormente se realiza una socialización en inglés, en la cual cada grupo de proyecto tomará la palabra y dará su punto de vista en un escrito en inglés acerca de las ventajas y desventajas del Lenguaje Python que fueron halladas por su equipo de trabajo.

  Guarde el archivo con el nombre **"Actividad_de_aprendizaje_inicial.pdf"** y subirlo al portafolio del aprendiz en la carpeta asignada por el instructor.

Ambiente requerido: Ambiente de formación

Materiales: Video beam, videos, marcadores, lápices, esferos, hojas, computador.

Tiempo Estimado: 2 horas.

3.2. ACTIVIDADES DE CONTEXTUALIZACIÓN E IDENTIFICACIÓN DE CONOCIMIENTOS NECESARIOS PARA EL APRENDIZAJE.

3.2.1 Actividad de Aprendizaje 06_1 Analizar Diagramas de Clases según el modelamiento UML.

Para los profesionales que se encargarán de codificar los diferentes elementos del Software, el diagrama de clases se convierte en pieza fundamental para establecer el alcance del Software, se convierte en instrumento de navegación para definir la ruta de desarrollo, teniendo en cuenta lo anterior, a continuación, se adiciona un diagrama de clases que describe uno de tantos procesos realizados en una universidad, responda a partir del diagrama:

a) ¿Qué proceso de una universidad representa este diagrama?
b) Adicione los atributos y comportamientos necesarios por clase.
c) Evalúe las relaciones Tiene, Miembro y Asignado, ¿qué tipo de relación fue definida?, ¿considera que es el tipo correcto?
d) Evalúe las multiplicidades por relación definidas, ¿considera que son las correctas?, si alguna de las multiplicidades no es la correcta, argumente.

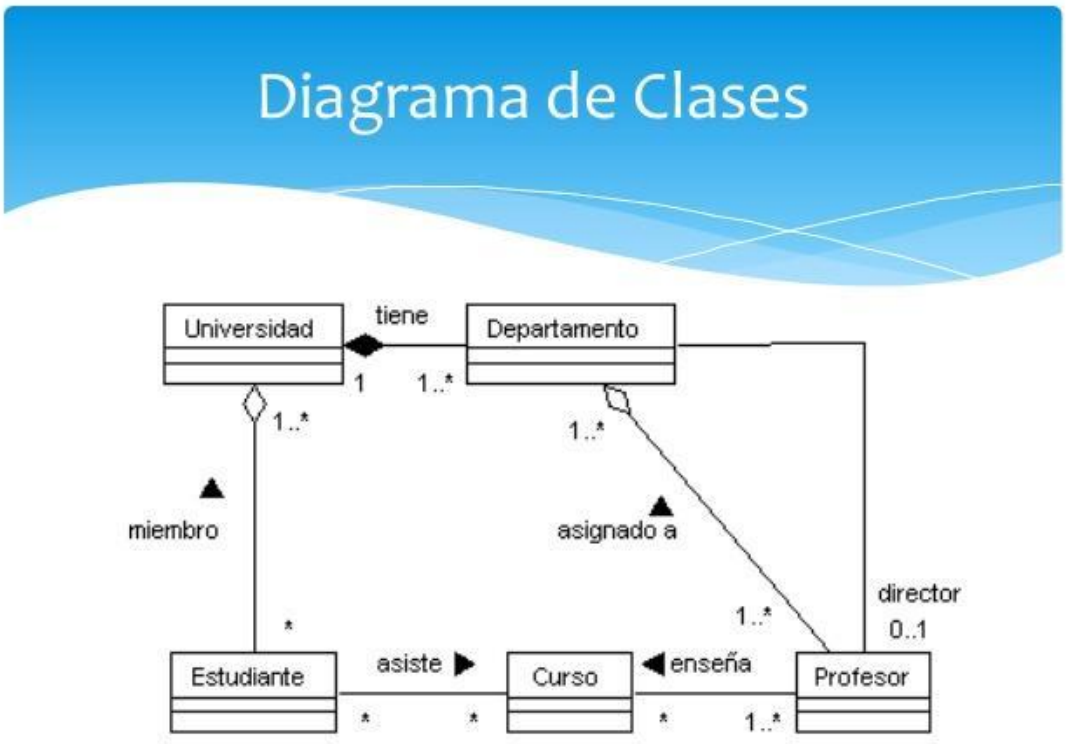


Fig. 5. Diagrama de clases – Gestión de una universidad. Tomado de: Fuente propia.

Para el desarrollo de esta actividad puede emplear la aplicación DIA, StarUML, CANVAS o la aplicación gratuita en línea Lucidchart - www.lucidchart.com. Esta actividad hace parte del entregable EVIDENCIA GUÍA DE APRENDIZAJE 6. Una vez terminada la actividad, el instructor dará las indicaciones para que sea socializada con todo el grupo de aprendices.

Guardar actividad como: Evidencia_Aprendizaje_3_2_1.pdf

Ambiente Requerido: Ambiente de formación con equipos de cómputo y acceso a internet.
Materiales y/o Recursos: Libreta de apuntes de cada aprendiz.
Tiempo para la actividad: 1 hora.

3.2.2 Actividad de Aprendizaje 06_2 Interpretar Diagramas de Clases según el modelamiento UML.

La interpretación adecuada del diagrama de clases es fundamental para los profesionales que se encargarán de codificar los diferentes elementos del Software, este diagrama define de manera gráfica y sencilla de interpretar el alcance del Software, teniendo en cuenta lo anterior, se adiciona un diagrama de clases que describe el proceso de ventas de un local comercial, responda:

a) ¿Qué tipos de usuario tienen acceso al sistema?
b) ¿Qué acciones puede realizar el Administrador del sistema a nivel de Usuarios?
c) Adicione a los atributos y comportamientos de todas las clases, la visibilidad correcta.
d) ¿Los comportamientos de la clase Venta son los correctos o están incompletos? Argumente.
e) Adicione por cada relación definida su descripción y multiplicidad.

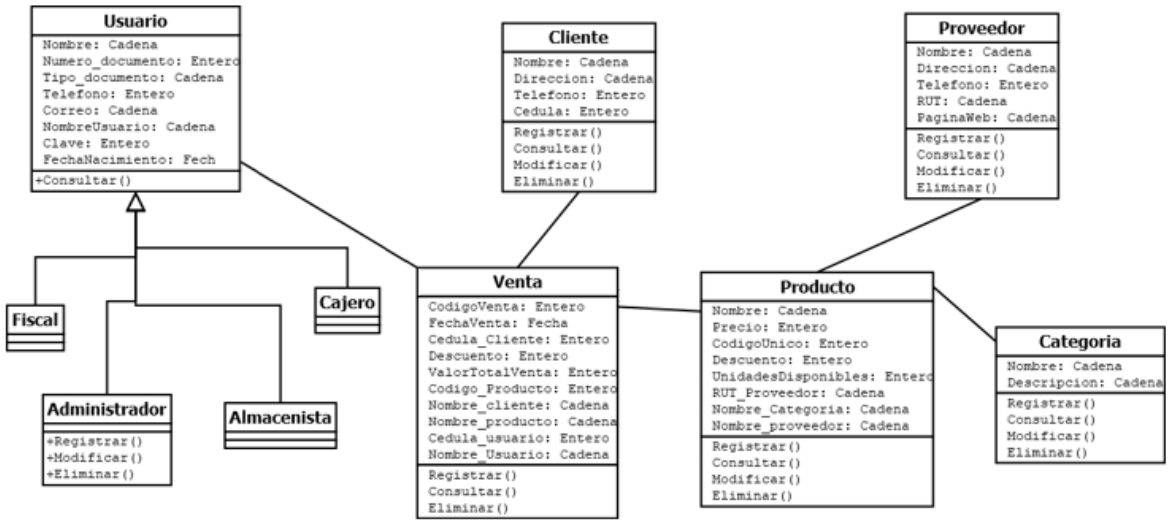


Fig. 6. Diagrama de clases – Gestión de ventas. Tomado de: Fuente propia.

Para el desarrollo de esta actividad puede emplear la aplicación DIA, StarUML o la aplicación gratuita en línea Lucidchart - www.lucidchart.com. Esta actividad hace parte del entregable EVIDENCIA GUÍA DE APRENDIZAJE 6. Una vez terminada la actividad, el instructor dará las indicaciones para que sea socializada con todo el grupo de aprendices.

Guardar actividad como: Evidencia_Aprendizaje_3_2_2.pdf

Ambiente Requerido: Ambiente de formación con equipos de cómputo y acceso a internet.
Materiales y/o Recursos: Libreta de apuntes de cada aprendiz.
Tiempo para la actividad: 1 hora.

3.2.3 Asociar los conceptos relacionados con el patrón de diseño MVC aplicados al desarrollo de aplicaciones web en PYTHON

Descripción de las actividades.

- En actividad se asociarán los principios de la programación orientada a objetos con una representación creativa de los conceptos y el código básico necesario para llevarlo a cabo en el lenguaje de programación. Se deben formar grupos del proyecto formativo.
- Iniciaremos la lectura en grupos del material de apoyo "**6_3_2_3_LECTURA_Entendiendo_PYTHON.pdf**", **6_3_2_3_ANALIZANDO_MVC.docx** y lectura "**6_7_Que_Son_Patrones_de_Diseño.pdf"** que se encuentra en la carpeta Materiales del Aprendiz.
- Posteriormente el grupo socializa los conceptos acerca del MCV aplicados el desarrollo de aplicaciones web.
- Los integrantes del grupo deben presentar en el ambiente de aprendizaje dicha representación haciendo uso de carteleras o diapositivas con los dibujos o gráficas que consideren pertinentes.
- Se debe guardar el archivo en un documento en Word llamado "**Patrón_de_diseño_MVC"** y posteriormente subirlo a territorio en la carpeta asignada por el Instructor

Ambiente requerido: Ambiente de formación

Materiales: Video beam, videos, marcadores, lápices, esferos, hojas, computador

Tiempo Estimado: 4 horas.

3.2.3.1Actividad de Ejercicios iniciales sobre conceptos de Programación Orientada a Objetos.

El aprendiz deberá ingresar al material de apoyo y descargar el archivo llamado 6_3_2_1_MI_PRIMERA_ESTRUCTURA_MVC y realizara el ejercicio propuesto tomara evidencias Guarde el archivo con el nombre "MI_PRIMERA_ESTRUCTURA_MVC**.pdf"** y subirlo al portafolio del aprendiz en la carpeta asignada por el instructor.

Ambiente requerido: Ambiente de formación

Materiales: Video beam, videos, marcadores, lápices, esferos, hojas, computador.

Tiempo Estimado: 2 horas.

3.2.3.2 Actividad de Ejercicios iniciales sobre conceptos de Programación Orientada a Objetos.

- Realizar la lectura individual del material de apoyo sobre "**6_3_Lectura Clases, Objetos y Encapsulamiento, 6_4_Lectura Herencia y 6_5_Lectura Polimorfismo".**
- Se realiza una socialización, en la cual cada grupo de proyecto tomará la palabra y dará su punto de vista acerca de clases, objetos, herencia y polimorfismo que fueron halladas por su equipo de trabajo.
- Guarde el archivo con el nombre "**Actividad_de_objetos y polimorfismo en python.pdf"** y subirlo al portafolio del aprendiz en la carpeta asignada por el instructor.

Tiempo Estimado: 4 horas.

Ambiente requerido: Ambiente de formación

Materiales: Video beam, videos, marcadores, lápices, esferos, hojas, computador.

3.2.3.3 Actividad de Ejercicios iniciales sobre conceptos de Programación Orientada a Objetos.

- El aprendiz deberá ingresar al material de apoyo y debe descargar el archivo llamado:

  6_3_2_3_3 _EJERCICIO BASICO PARA CREAR LOGIN EN PYTHON y realizara el ejercicio propuesto tomara evidencias Guarde el archivo con el nombre "**EJERCICIO_BASICO_PARA_CREAR_LOGIN_EN_PYTHON.pdf"** y subirlo al portafolio del aprendiz en la carpeta asignada por el instructor.

Ambiente requerido: Ambiente de formación

Materiales: Video beam, videos, marcadores, lápices, esferos, hojas, computador.

Tiempo Estimado: 2 horas.

3.2.4 Actividad de Aprendizaje

Asociar los conceptos relacionados con el patrón de diseño MVC aplicados al desarrollo de aplicaciones web en PYTHON

Descripción de las actividades.

- Al hacer uso de patrones de diseño en nuestras aplicaciones web Python, se obtienen varias ventajas y con el fin de analizarlas, se propone que el aprendiz realice la lectura "**6_6_LECTURA_Entendiendo_MVC_en_PYTHON.pdf**" Además, de acuerdo a los diagramas representativos y al ejemplo que se encuentra en el archivo "**6_8_Contextualizacion_Modelo_Vista_Controlador.pdf**" identifique la secuencia de eventos de interacción entre el modelo, la vista y el controlador teniendo en cuenta la siguiente situación: En un sistema de gestión de inventario, un usuario realiza una petición de consulta al sistema, que le permita visualizar todos los productos con los que cuenta en bodega.
- Utilice un modelo de diagrama similar al desarrollado en el material de apoyo y, de forma individual, socialícelo con sus compañeros

Ambiente requerido: Ambiente de formación

Materiales: computador, Xampp , visual studio code,

Tiempo Estimado: 10 horas.

3.2.6 Realizar el siguiente ejemplo ubicado en el "taller 06_2_Programacion_ Orientada_ objetos",

- Guarde el archivo con el nombre "**Actividad_de_programacion oritentada a objetos en Python.pdf"** y subirlo al portafolio del aprendiz en la carpeta asignada por el instructor.

Tiempo Estimado: 8 horas.

Ambiente requerido: Ambiente de formación

Materiales: Video beam, videos, marcadores, lápices, esferos, hojas, computador.

3.2.7 Realizar el siguiente ejemplo ubicado en el taller 06_03_Programacion_ Orientada_ objetos2

- Guarde el archivo con el nombre "**actividad_de_ejercicios oritentada a objetos en python.pdf"** y subirlo al portafolio del aprendiz en la carpeta asignada por el instructor.

Tiempo Estimado: 10 horas.

Ambiente requerido: Ambiente de formación

Materiales: Video beam, videos, marcadores, lápices, esferos, hojas, computador.

3.2.8 Realizar el siguiente ejemplo ubicado en el "7_1_LECTURA_Entendiendo_MVC_en_PYTHON",

- Guarde el archivo con el nombre "**Actividad_de_ejercicios mvc python.pdf"** y subirlo al portafolio del aprendiz en la carpeta asignada por el instructor.

Tiempo Estimado: 8 horas.

Ambiente requerido: Ambiente de formación

Materiales: Video beam, videos, marcadores, lápices, esferos, hojas, computador.

3.3. ACTIVIDADES DE APROPIACIÓN DEL CONOCIMIENTO (CONCEPTUALIZACIÓN Y TEORIZACIÓN)

3.3.1 Actividad de Aprendizaje 03_1 Determinar los datos e información necesaria para el diseño del Diagrama de Clases.

Realice un adecuado ejercicio de abstracción, a partir del siguiente enunciado defina clases (Módulos), atributos y comportamientos necesarios para diseñar el diagrama de clases que represente el proceso Cadena de Venta de Electrodomésticos.

CASO DE ESTUDIO: Cadena de Venta de Electrodomésticos.
La cadena de venta de electrodomésticos Gabartino lanzará una nueva línea de créditos personales y sus directivos quieren minimizar el tiempo necesario para el otorgamiento de los mismos. Los potenciales compradores se acercan al mostrador de atención al cliente, donde preguntan qué artículos son aptos para el otorgamiento del crédito. Si le gusta alguno de los artículos que se ofrecen, se acerca a un vendedor y le solicita que se lo muestre. Una vez que ha visto el producto en funcionamiento, el cliente puede decidir si lo compra o no, de no agradarle el artículo, puede acercarse nuevamente al mostrador de atención al cliente y ver si hay otro producto de su interés, también puede decidir irse del local sin comprar nada. En caso de que el artículo sea del agrado del cliente, el vendedor le pide sus datos y emite una orden de solicitud de crédito, la cual es entregada al departamento de ventas. Los empleados de este departamento se encargan de estudiar la condición crediticia del cliente a través del sistema SisCre, simultáneamente se analiza su situación penal a través de un sistema de consulta provisto por la Interpol, si alguno de los dos sistemas arroja elementos en contra del potencial cliente entonces el departamento de venta emite una orden de crédito rechazada. En caso contrario, se emite una orden de crédito aprobada.

Tenga en cuenta, para esta actividad solo es necesario definir el listado de clases, atributos y comportamientos necesarios por clase, el diseño del diagrama en su totalidad se realizará en el siguiente numeral. Esta actividad hace parte del entregable EVIDENCIA GUÍA DE APRENDIZAJE 6. Una vez terminada la actividad, el instructor dará las indicaciones para que sea socializada con todo el grupo de aprendices.

Guardar actividad como: Evidencia_Aprendizaje_3_3_1.pdf

Ambiente Requerido: Ambiente de formación con equipos de cómputo y acceso a internet.
Materiales y/o Recursos: Libreta de apuntes de cada aprendiz.
Tiempo para la actividad: 2 horas.

3.3.1.1 Actividad de Aprendizaje

- Realizar el siguiente ejemplo ubicado en el "**6_3_3_2_TALLER_DESARROLLANDO_ESTRUCTURA_MVC.docx**", utilizando un patrón de diseño o desarrollo de software, se emplea para separar la lógica de la vista, esto facilita la portabilidad de la aplicación y el mantenimiento mucho más organizado y ordenado.
- Guarde el archivo con el nombre "**Actividad_de_ejemplomvc.zip"** y subirlo al portafolio del aprendiz en la carpeta asignada por el instructor.

Tiempo Estimado: 10 horas.

Ambiente requerido: Ambiente de formación

Materiales: Video beam, videos, marcadores, lápices, esferos, hojas, computador.

3.3.2 Actividad de Aprendizaje 06_2 Diseñar Diagramas de Clases para diferentes problemas reales según parámetros establecidos.

A partir del siguiente enunciado y de las clases, atributos y comportamientos obtenidos en el numeral anterior, diseñe el diagrama de clases que represente el proceso Cadena de Venta de Electrodomésticos.

CASO DE ESTUDIO: Cadena de Venta de Electrodomésticos.
La cadena de venta de electrodomésticos Gabartino lanzará una nueva línea de créditos personales y sus directivos quieren minimizar el tiempo necesario para el otorgamiento de los mismos. Los potenciales compradores se acercan al mostrador de atención al cliente, donde preguntan qué artículos son aptos para el otorgamiento del crédito. Si le gusta alguno de los artículos que se ofrecen, se acerca a un vendedor y le solicita que se lo muestre. Una vez que ha visto el producto en funcionamiento, el cliente puede decidir si lo compra o no, de no agradarle el artículo, puede acercarse nuevamente al mostrador de atención al cliente y ver si hay otro producto de su interés, también puede decidir irse del local sin comprar nada. En caso de que el artículo sea del agrado del cliente, el vendedor le pide sus datos y emite una orden de solicitud de crédito, la cual es entregada al departamento de ventas. Los empleados de este departamento se encargan de estudiar la condición crediticia del cliente a través del sistema SisCre, simultáneamente se analiza su situación penal a través de un sistema de consulta provisto por la Interpol, si alguno de los dos sistemas arroja elementos en contra del potencial cliente entonces el departamento de venta emite una orden de crédito rechazada. En caso contrario, se emite una orden de crédito aprobada.

Para el desarrollo de esta actividad puede emplear la aplicación DIA, StarUML o la aplicación gratuita en línea Lucidchart - www.lucidchart.com. Esta actividad hace parte del entregable EVIDENCIA GUÍA DE APRENDIZAJE 6. Una vez terminada la actividad, el instructor dará las indicaciones para que sea socializada con todo el grupo de aprendices.

Guardar actividad como: Evidencia_Aprendizaje_3_3_2.pdf

Ambiente Requerido: Ambiente de formación con equipos de cómputo y acceso a internet.
Materiales y/o Recursos: Libreta de apuntes de cada aprendiz.
Tiempo para la actividad: 2 horas.

3.3.3. Actividades de apropiación del conocimiento (Conceptualización y Teorización). Creando una Aplicación Web Desde Cero Usando Python Flask y MySQL _ Envato Tuts+.

Actividades de Apropiación del conocimiento de acuerdo al documento compartido y a las prácticas de construcción de la arquitectura MVC el aprendiz deberá realizar la CRUD utilizando Python y Mysql.

- Basado en el material de apoyo disponible (Videos, documentos y páginas web) y el ejemplo "**Creando una Aplicación Web Desde Cero Usando Python Flask y MySQL _ Envato Tuts+.pdf**", se debe construir una base de datos del proyecto asignado y realizar las respectivas clases en Python.
- Se tendrán que diseñar los formularios básicos para la CRUD del proyecto asignado (usando Bootstrap).
- Posteriormente se debe crear la clase de conexión a base de datos usando Mysql.
- Esta actividad es de carácter grupal según como esté conformado el proyecto y se debe tener en cuenta que todo el código programado debe estar con (métodos, clases, atributos) y adicionalmente debe estar documentado en su totalidad.

Ambiente requerido: Ambiente de formación

Materiales: computador, Xampp, MicroSotf visual studio code,

Tiempo Estimado: 20 horas.

3.3.4. Actividades de apropiación del conocimiento: Comprender la implementación del patrón Modelo Vista controlador en aplicaciones web Python

- Siga los ejemplos básicos de implementación del patrón MVC, que se encuentran en la carpeta comprimida "**6_9_EJEMPLOS_Modelo_Vista_Controlador.rar**" Y realice el ejercicio propuesto
- Guarde sus evidencias copiando y pegando el código desarrollado y tomando pantallazos de los resultados, en un único archivo de Word con el nombre "**Evidencias_MVC_PYTHON**"

Ambiente requerido: Ambiente de formación

Materiales: computador, Xampp, Microsift visual studio code,

Tiempo Estimado: 30 horas.

3.3.5 Actividades de apropiación del conocimiento: Implementar el patrón Modelo Vista controlador en aplicaciones web Python

Construir un proyecto utilizando el patrón MVC:

- Teniendo en cuenta los ejemplos desarrollados anteriormente, y el ejemplo de implementación de una CRUD con MVC y Python, que se encuentra dentro del material de apoyo en el archivo "**6_11_CRUD_MVC_PYTHON.zip**" implemente las operaciones CRUD propias de su proyecto de formación.
- Agregue sus evidencias (Pantallazos del funcionamiento) a un archivo de Word, con el nombre "**CRUD_PYTHON_MVC_Proyecto final_PYTHON"**

Ambiente requerido: Ambiente de formación

Materiales: computador, Xampp,Microsoft visual studio code,

Tiempo Estimado: 60 horas

3.4. ACTIVIDADES DE TRANSFERENCIA DEL CONOCIMIENTO.

Antes de diseñar el diagrama de clases UML de su propuesta de proyecto formativo, se sugiere repasar el material de apoyo del aprendiz:
UML_Diagrama_de_Clases.pdf
UML-DiagramaClaseObjeto.pdf

3.4.1 Actividad de Aprendizaje 04_1 Definir las clases y elementos fundamentales para el diseño del proyecto.

Realice un adecuado ejercicio de abstracción, a partir del problema detectado en la empresa cliente, defina el conjunto de clases (Módulos), atributos y comportamientos necesarios para diseñar el diagrama de clases que represente el proceso ideal de la empresa cliente.

Guardar actividad como: Evidencia_Aprendizaje_3_4_1.pdf

Ambiente Requerido: Ambiente de formación con equipos de cómputo y acceso a internet.
Materiales y/o Recursos: Libreta de apuntes de cada aprendiz.
Tiempo para la actividad: 2 hora.

3.4.2 Actividad de Aprendizaje 04_2 Diseñar el Diagrama de clases del proyecto.

A partir del conjunto de clases, atributos y comportamientos obtenidos en el numeral anterior, diseñe el diagrama de clases UML que represente el proceso ideal de la empresa cliente.

Para el desarrollo de esta actividad puede emplear la aplicación DIA, StarUML o la aplicación gratuita en línea Lucidchart - www.lucidchart.com. Esta actividad hace parte del entregable EVIDENCIA GUÍA DE APRENDIZAJE 6. Una vez terminada la actividad, el instructor dará las indicaciones para que sea socializada con todo el grupo de aprendices.

Guardar actividad como: Evidencia_Aprendizaje_3_4_2.pdf

Ambiente Requerido: Ambiente de formación con equipos de cómputo y acceso a internet.
Materiales y/o Recursos: Libreta de apuntes de cada aprendiz.
Tiempo para la actividad: 2 horas.

3.4.3 Actividad de Aprendizaje 04_3 Diseñar el Diagrama de clases del proyecto.

A partir de su propuesta de proyecto formativo, codifique:

a) Cree la estructura de carpetas MVC a partir de su propuesta de proyecto formativo, adiciónelo a la carpeta HTDOCS del servidor de pruebas XAMPP.
b) Cree el conjunto de clases (Atributos y comportamientos) empleando el lenguaje de programación Python a partir del diagrama de clases de su propuesta de proyecto formativo.
c) Diseñe el conjunto de interfaces que permitan la entrada de datos por parte del usuario.
d) Defina mecanismos que permitan validar los datos de entrada por interfaz de usuario.
e) Defina la conexión al sistema gestor de base de datos seleccionado.
f) Codifique y establezca la funcionalidad del LOGIN.
g) Codifique y establezca la funcionalidad CRUD de los módulos establecidos.

Ambiente Requerido: Ambiente de formación con equipos de cómputo y acceso a internet.
Materiales y/o Recursos: Libreta de apuntes de cada aprendiz.
Tiempo para la actividad: 60 horas.

4. ACTIVIDADES DE EVALUACIÓN

Tome como referencia la técnica e instrumentos de evaluación citados en la guía de Desarrollo Curricular

Evidencias de Aprendizaje	Criterios de Evaluación	Técnicas e Instrumentos de Evaluación
Evidencias de Conocimiento: Características de la programación orientada a objetos.	Comprende los conceptos básicos de la programación orientada a objetos: el alumno demuestra entender las ventajas y desventajas de este paradigma, así como sus	Formulación de Preguntas **Instrumento** Cuestionario Taller de programación.

Responder sobre que es el patrón de diseño MVC, el esquema de ficheros y la interacción entre cada una de las capas que lo componen **Evidencias de Desempeño:** Implementa los conceptos de la programación orientada a objetos y la conexión a base de datos mediante PDO. Presentar sustentación de ejercicios propuestos en la guía de aprendizaje correspondiente a MVC. **Evidencias de Producto:** Aplica a su proyecto de formación cada uno de los conceptos de forma pertinente y de acuerdo a los estándares de programación. Desarrollar ejercicios propuestos en la guía, en los cuales aplique los conceptos de MVC de acuerdo a las situaciones planteadas	conceptos centrales como abstracción, encapsulamiento, herencia y polimorfismo. Comprende la teórica de la arquitectura MVC: el alumno identifica correctamente los principios y funcionamiento del patrón Modelo-Vista-Controlador en el desarrollo de aplicaciones web. Aplicaca de POO y bases de datos: el alumno aplica adecuadamente los principios de la programación orientada a objetos, así como las técnicas de conexión a bases de datos mediante PDO en el desarrollo de software. Reconoce el valor de los patrones de diseño: el alumno compara aplicaciones con y sin patrones de diseño, reconociendo las ventajas en mantenibilidad, escalabilidad y buenas prácticas que proporcionan. Desarrolla aplicaciones web con MVC: el alumno implementa correctamente la arquitectura Modelo-Vista-Controlador en el desarrollo de aplicaciones web funcionando de forma adecuada.	Incorporación al proyecto formativo Actividad de Pre Saberes Cuestionario sobre MVC Lista de Chequeo Entregable CRUD con MVC aplicada a proyecto de formación

5. GLOSARIO DE TÉRMINOS

Clase: modelo o plano para la creación de objetos que comparten atributos y métodos. Define las características y el comportamiento.

Objeto: instancia individual de una clase, con valores concretos para sus atributos.

Atributo: variable que forma parte de la definición de la clase, caracteriza al objeto.

Método: función o procedimiento asociado a una clase, define su comportamiento.

Abstracción: proceso mental de aislar un elemento de su contexto y considerarlo de forma independiente.

Encapsulamiento: empaquetar atributos y métodos dentro de una clase. Ocultación de detalles de implementación.

Herencia: mecanismo que permite basar una clase en otra ya existente, heredando sus atributos y métodos.

Polimorfismo: propiedad por la que objetos de diferentes clases pueden responder a un mismo mensaje cada uno a su manera.

Modelo: representa los datos, lógica de negocio y reglas de la aplicación.

Vista: interfaz de usuario. Presenta el modelo y captura la entrada del usuario.

Controlador: intermediario entre el modelo y las vistas. Gestiona la lógica y el flujo de datos.

Enrutador: gestiona las peticiones entrantes y las redirecciona al controlador apropiado.

Patrón: solución reutilizable a un problema común en el desarrollo de software.

Instancia: objeto concreto creado a partir de una clase. Tiene estado y comportamiento.

Interfaz: conjunto de métodos y propiedades que caracterizan un tipo de objeto.

Modularidad: principio de diseño, divide un sistema en partes separadas y discretas (módulos)

Encapsulamiento: empaquetar y ocultar partes independientes de un sistema.

Acoplamiento: grado de interdependencia entre módulos de un sistema.

Cohesión: grado en que partes de un módulo están conectadas.

Reutilización: uso de código existente en nuevas aplicaciones.

Herencia múltiple: clase que hereda de más de una clase padre.

6. REFERENTES BILBIOGRÁFICOS

- El paradigma de Programación Orientada a Objetos en ASP y el patrón de arquitectura MVC:
- Documentación oficial PYTHON.. Disponible en:
- https://aws.amazon.com/es/what-is/python/
- Fundamentos básicos PYTHONDisponible en:
- https://softwarecrafters.io/python/tutorial-de-python-introduccion
- Sintaxis básica PDO. 30 de Enero de 2018. Disponible en:

https://diego.com.es/tutorial-de-pdo

- Manual PYTHON. Disponible en:
- https://persoal.citius.usc.es/eva.cernadas/informaticaparacientificos/material/libros/Python%20para%20todos.pdf

Videos

PYTHON DESDE CERO:

https://www.educacionit.com/cursos-de-python?gad_source=2&gclid=EAIaIQobChMIrf6kk8zNiwMVsEtHAR3F4iBpEAAYASAAEgKWGfD_BwE&utm_campaign=co_conv_search_pyd_python_ca_12024___&utm_medium=search&utm_source=google

CLASES Y OBJETOS EN PYTHON

https://www.youtube.com/watch?v=nKPbfIU442g

CLASES Y OBJETOS EN PYTHON

https://www.youtube.com/watch?v=9x7RK6mb1uA

7. CONTROL DEL DOCUMENTO

	Nombre	Cargo	Dependencia	Fecha
Autor (es)	Jannez Milson Urrego Quiroz. William Ramon Flórez	Instructor	Coordinación	13/12/2021

8. CONTROL DE CAMBIOS (diligenciar únicamente si realiza ajustes a la guía)

	Nombre	Cargo	Dependencia	Fecha	Razón del Cambio
Autor (es)	Magaly lozano Beltrán	Instructor	Articulación	01/07/2022	Actualización Guía
	Edwin Cañon	Instructor	Articulación	24/11/2022	Actualización actividad 3.3
	Jannez Urrego	Instructor	Articulación	24/11/2022	Actualización actividad 3.3
	Gabriel Pinilla	Instructor	Articulación	13/12/2023	Actualización Guía
	Fernando Villegas V.	Instructor	Articulación	13/12/2023	Actualización Guía