



PROCESO DE GESTIÓN DE FORMACIÓN PROFESIONAL INTEGRAL

FORMATO GUÍA DE APRENDIZAJE

IDENTIFICACIÓN DE LA GUÍA DE APRENDIZAJE

- Denominación del Programa de Formación: Análisis y desarrollo de software
- Código del Programa de Formación: 228118
- Nombre del Proyecto: Desarrollo de software como recurso para la gestión de información de las organizaciones en el contexto de la innovación, transformación digital y en el marco de políticas y requisitos de la empresa.
- Fase del Proyecto Planificación
- Actividad de Proyecto: Establecer metodología de desarrollo a utilizar
- Competencia: Modelado de artefactos de software
 - Resultados de Aprendizaje Alcanzar: Determinar las características técnicas de la interfaz gráfica del software adoptando estándares.
- Duración de la Guía: 180 horas

2. PRESENTACIÓN



Respetado aprendiz, iniciamos un nuevo trimestre donde continúa la etapa del desarrollo de su proyecto formativo, en la cual profundizará en los conocimientos adquiridos sobre programación orientada a objetos, arquitectura limpia para el desarrollo de software y diseño de interfaces de usuario.

El desarrollo de software y el desarrollo móvil son disciplinas relacionadas que se refieren al proceso de crear aplicaciones y sistemas informáticos. Ambos procesos involucran desarrollar código, probar el código, documentar el código y llevar a cabo otras tareas de mantenimiento.

El desarrollo móvil es cada vez más importante en la actualidad debido a la cantidad de usuarios que usan dispositivos móviles para acceder a la información y a los servicios. Una aplicación móvil bien diseñada y desarrollada es una forma eficaz de llegar a estos usuarios y ofrecerles una experiencia óptima en su quehacer diario. Además, el diseño y la implementación de aplicaciones móviles pueden ayudar a las empresas a aumentar su visibilidad y mejorar su marca.

Las estadísticas de desarrollo móvil del año 2022 muestran que el número de usuarios de aplicaciones móviles se incrementará en un 11,3 por ciento, alcanzando los 3,8 mil millones de usuarios.



(Máximo 10 renglones)

3. FORMULACIÓN DE LAS ACTIVIDADES DE APRENDIZAJE

3.1 Actividades de Reflexión inicial

3.1.1 Actividad 1: Comprender la importancia de la usabilidad en el desarrollo de las aplicaciones móviles.

Descripción de la actividad

Realiza la lectura del artículo “Usabilidad de aplicaciones móviles: todo lo que necesita saber” el cual se encuentra disponible en el siguiente enlace web bit.ly/usabilidadmoviles o escaneando el código QR



Una vez que finalices la lectura, da respuesta a las siguientes preguntas:

- Según tus palabras, ¿qué es la usabilidad de las aplicaciones móviles?
- De las prácticas recomendadas de usabilidad de aplicaciones móviles, ¿Cuál consideras más importante? ¿Por qué?
- ¿Qué problemas consideras que se presenta con una navegación compleja y la compatibilidad de resolución de pantalla en las aplicaciones móviles?
- ¿Qué inconvenientes has tenido al usar aplicaciones móviles?

Participe en la actividad de socialización, compartiendo con el grupo las elaboraciones individuales.

- **Ambiente Requerido:** Computadores, pantalla de TV y sillas.
- **Materiales:** Internet y navegador web.
- **Tiempo sugerido:** 1 hora y 30 minutos.
- **Entregable:** Respuesta de las preguntas.

3.2 Actividades de contextualización e identificación de conocimientos necesarios para el aprendizaje

3.2.1 Actividad 2: Identificar los conocimientos previos de los conceptos fundamentales de bases de programación.



Descripción de la actividad

Responda a las preguntas del cuestionario disponible en https://es.educaplay.com/recursos-educativos/13866201-conocimientos_previos_moviles.html, de manera individual, en el que evidenciará conocimientos previos correspondientes a las bases de la programación.

También puedes acceder escaneando el código QR



Al finalizar comparte el puntaje obtenido con tu instructor.

- **Estrategia didáctica:** La pregunta
- **Técnica didáctica:** Cuestionario

- **Ambiente Requerido:** Computadores, pantalla de TV y sillas.
- **Materiales:** Internet y navegador web.
- **Tiempo sugerido:** 30 minutos.
- **Entregable:** Respuesta de preguntas.

3.2.2 Actividad 3: Comprender las principales diferencias entre aplicaciones nativas e híbridas.

Existen dos tipos principales de aplicaciones móviles: aplicaciones nativas y aplicaciones híbridas. Las aplicaciones nativas están diseñadas para un sistema operativo específico, como iOS o Android. Esto significa que entregan la mejor experiencia de usuario en comparación con aplicaciones híbridas. Estas aplicaciones también se pueden personalizar de acuerdo con el sistema operativo, lo que ofrece una experiencia más enriquecida.

Para profundizar en los tipos de aplicaciones móviles, te invito a leer el artículo “7 diferencias entre aplicaciones híbridas y nativas” disponible en <https://www.renacen.com/blog/diferencias-entre-aplicaciones-hibridas-y-nativas/> o a través del QR.



Luego de realizar la lectura, responda a las siguientes tres (3) preguntas:

1. *¿Qué consideras como mayor ventaja de las aplicaciones nativas?*
2. *Según tu criterio, ¿Cuál es la mayor desventaja del desarrollo nativo?*



3. ¿Qué resaltas sobre la comparativa entre aplicaciones nativas e híbridas?

En un documento de Word escribe las respuestas y socializa con tus compañeros y el instructor. Sube la evidencia en territorium en el espacio “*Aplicaciones nativas vs híbridas*”

- **Estrategia didáctica:** La pregunta
- **Técnica didáctica:** Cuestionario

- **Ambiente Requerido:** Computadores, pantalla de TV y sillas.
- **Materiales:** Internet, navegador web.
- **Tiempo sugerido:** 2 horas.
- **Entregable:** Respuesta de preguntas.

3.3 Actividades de apropiación del conocimiento (Conceptualización y Teorización)

3.3.1 Actividad 4: Codificar programas usando la sintaxis básica del lenguaje Dart.

Para la construcción de aplicaciones móviles usaremos Flutter el cual es un sdk que trabaja con el lenguaje de programación Dart.

Dart es un lenguaje de programación de código abierto creado por Google. Está diseñado para ser fácil de aprender y usar, así como para ofrecer un alto rendimiento. El lenguaje está optimizado para la web, lo que significa que los desarrolladores pueden crear aplicaciones web escalables con Dart. Para conocer la sintaxis básica de este lenguaje, el instructor realiza una presentación de los siguientes temas:

- Variables
- Tipos de datos
- Null Safety
- Colecciones
- Funciones

A continuación, de manera individual debes realizar el taller de programación con Dart, el cual te permitirá afianzar los conocimientos adquiridos de este lenguaje.

Una vez finalizada la actividad se debe subir como evidencia en Territorium en el espacio “*Taller programación Dart*”.



- **Estrategia didáctica:** Taller
- **Técnica didáctica:** Observación

- **Ambiente Requerido:** Computadores, pantalla de TV y sillas.
- **Materiales:** Internet, navegador web y VS Code.
- **Tiempo sugerido:** 20 Horas (5 horas de trabajo autónomo)
- **Entregable:** Código fuente de cada programa.

3.3.2 Actividad 5: Aplicar los conceptos esenciales de programación orientada a objetos con el lenguaje Dart, para la solución de un caso estudio.

Con la finalidad de seguir profundizando en lenguaje Dart, el instructor imparte las temáticas pertinentes al paradigma de programación orientada a objetos.

Se presenta un caso estudio, el cual debe analizar detenidamente para dar respuesta al problema propuesto.

Al finalizar el desarrollo de esta actividad se debe sustentar y subir la evidencia en Territorium en el espacio “Caso estudio programación Dart”.

- **Estrategia didáctica:** Aprendizaje basado en problemas
- **Técnica didáctica:** Observación

- **Ambiente Requerido:** Computadores, pantalla de TV y sillas.
- **Materiales:** Internet, navegador web y VS Code.
- **Tiempo sugerido:** 60 Horas (10 horas de trabajo autónomo)
- **Entregable:** Código fuente del programa desarrollado.

3.4 Actividades de transferencia del conocimiento

3.4.3 Actividad 6: Construir una aplicación móvil utilizando el consumo de una API.

En esta etapa del proceso formativo cuentas con una gran cantidad de conocimientos, entre ellos, las APIs, las cuales permiten a los programadores conectar una aplicación con otra, permitiendo que los datos se compartan entre sistemas.

Para esta actividad se debe consumir una API, la cual será presentada por el instructor.



Al finalizar el desarrollo de esta actividad se debe sustentar y subir la evidencia en Territorium en el espacio determinado por el instructor (*“Aplicación móvil – Consumo de API”*).

- **Estrategia didáctica:** Aprendizaje basado en problemas
- **Técnica didáctica:** Observación

- **Ambiente Requerido:** Computadores, pantalla de TV y sillas.
- **Materiales:** Internet, navegador web y VS Code.
- **Tiempo sugerido:** 40 Horas (8 horas de trabajo autónomo)
- **Entregable:** Carpeta comprimida con todos los archivos del proyecto.



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: <ul style="list-style-type: none"> Comprender las principales diferencias entre aplicaciones nativas e híbridas 	<ul style="list-style-type: none"> Propone la interfaz gráfica de usuario cumpliendo reglas de usabilidad y accesibilidad para aplicaciones móviles. 	Técnica didáctica: Observación Instrumento de evaluación: Lista de verificación
Evidencias de Desempeño: <ul style="list-style-type: none"> Codificar programas usando la sintaxis básica del lenguaje Dart. Aplicar los conceptos esenciales de programación orientada a objetos con el lenguaje Dart, para la solución de un caso estudio. 	<ul style="list-style-type: none"> Construye el mapa de navegación cumpliendo reglas de usabilidad y accesibilidad. Elabora prototipos según los requisitos del software. 	Técnica didáctica: Observación Instrumento de evaluación: Rúbrica Técnica didáctica: Observación Instrumento de evaluación: Rúbrica
Evidencias de Producto: <ul style="list-style-type: none"> Construir una aplicación móvil utilizando el consumo de una API. 		Técnica didáctica: Observación Instrumento de evaluación: Rúbrica Técnica didáctica: Observación Instrumento de evaluación: Lista de verificación

5. GLOSARIO DE TÉRMINOS



API: Una API, o interfaz de programación de aplicaciones, es un conjunto de reglas que determinan cómo las aplicaciones o los dispositivos pueden conectarse y comunicarse entre sí.

Flutter: Flutter es un framework de aplicación de código abierto de Google para desarrollar aplicaciones nativas para Android, iOS, la Web, Windows, Mac, Linux y otros dispositivos.

Dart: Dart es un lenguaje de programación de código abierto creado por Google. Está diseñado para ser fácil de aprender y usar, así como para ofrecer un alto rendimiento. El lenguaje está optimizado para la web, lo que significa que los desarrolladores pueden crear aplicaciones web escalables con Dart.

POO: es un paradigma de programación que se basa en la abstracción de los programas a objetos reales. Estos objetos se componen de un estado (que es el conjunto de sus características) y un comportamiento, que son las acciones que realiza el objeto.

Aplicaciones móviles: Las aplicaciones móviles son herramientas digitales diseñadas para ser usadas desde un dispositivo móvil, como un iPhone, iPad o teléfono Android.

Usabilidad: La usabilidad es el grado en el que un producto o servicio puede ser usado con facilidad por los usuarios. Esto incluye la facilidad de aprender y recordar cómo usar el producto, la capacidad de realizar tareas con eficiencia y la satisfacción en el uso del producto. El objetivo de la usabilidad es crear productos que sean fáciles de usar, entender y disfrutar.

JSON: Es un formato de texto sencillo para el intercambio de datos. Se trata de un subconjunto de la notación literal de objetos de JavaScript.

Base de datos: Es un almacén de información digital organizada para su fácil acceso, uso, y almacenamiento. Están diseñadas para almacenar grandes cantidades de datos de diferentes tipos y permitir que sean agregados, extraídos, modificados y/o eliminados de una forma relativamente rápida y sencilla.

Experiencia de usuario (UX): La experiencia de usuario (UX) es una disciplina centrada en la comprensión de la interacción de los usuarios con un producto, sistema o servicio. Esta comprensión implica la estructura, los mecanismos, los contenidos y el diseño de la interacción.



6. REFERENTES BIBLIOGRÁFICOS

affde.com. (2021, noviembre 2). Usabilidad de aplicaciones móviles: todo lo que necesita saber.

Affde.com. <https://www.affde.com/es/mobile-app-usability.html>

Alviz, J. (2017, marzo 15). 7 diferencias entre aplicaciones híbridas y nativas. Renacen.

<https://www.renacen.com/blog/diferencias-entre-aplicaciones-hibridas-y-nativas/>

Arnaudelub. (2021, marzo 29). Flutter y Firebase login, todo lo que necesitas saber. Flutter

España. <https://medium.com/flutter-espa%C3%B1a/flutter-y-firebase-login-todo-lo-que-necesitas-saber-b1deada16f0f>

Cárdenas Villavicencio, O. E., Zea Ordóñez, M. P., Valarezo Pardo, M. R., & Ramón Ramón, R. A.

(2021). Comparativa de tendencias de desarrollo de software móvil. 3C TIC Cuadernos de desarrollo aplicados a las TIC, 10(1), 123–147. <https://doi.org/10.17993/3ctic.2021.101.123-147>

David. (2021, enero 7). Desarrollo de Apps con Flutter. Appcinking. <https://cink.es/blog/desarrollo-de-apps-con-flutter/>

Martínez, F. (2011). Aplicaciones para dispositivos móviles [versión electrónica]. Universidad Politécnica de Valencia.

7. CONTROL DEL DOCUMENTO

	Nombre	Cargo	Dependencia	Fecha
Autor (es)	Daniel Benavides Sánchez	Instructor	Teleinformática - Software	20 de enero de 2023

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

Nombre	Cargo	Dependencia	Fecha	Razón del Cambio



Autor (es)	Diego López Zapata	Instructor	Teleinformática - Software	12 Julio de 2023	Cambios en la versión del formato
-------------------	--------------------	------------	-------------------------------	---------------------	---