

EXAMEN DE CONOCIMIENTOS IOS – SWIFT

Nombre del candidato: Adrián Velázquez Contreras

Fecha de aplicación: 3 de agosto de 2023

Instrucciones: El examen se divide en dos partes la primera parte son preguntas abiertas que nos permitirán conocer la experiencia en desarrollo de aplicaciones IOS – Swift, el espacio es limitado así que puedes pedir una hoja adicional o escribir en la parte de atrás. La segunda parte se compone de un ejercicio práctico en donde se te darán instrucciones detalladas para que puedas desarrollar la aplicación móvil en una computadora que se te será proporcionada. **!MUCHA SUERTE!**

Preguntas abiertas

1. Proporciona un ejemplo simple de cada uno de los siguientes estatutos:

(Enumeration, Structure, Class, Extension, Protocol, Inheritance, let, var)

Adjunto un playground con los ejemplos mencionados.

```
// Un ejemplo de una función:  
func saludo(persona: String) -> String{  
    let bienvenida = "Hola \(persona)!"  
    return bienvenida  
}
```

2. Explica en tus propias palabras y con base a tu experiencia, el proceso que utilizas para consumir servicios de tipo REST. En tu respuesta puedes explicar desde librerías de terceros hasta procesos.

Para el consumo de servicio lo hago a través de URLSession de la siguiente manera:

Comienzo creando una clase llamada ServiceManager para gestionar las llamadas a los servicios, defino métodos específicos según el tipo de solicitud cada uno de los cuales acepta tanto la solicitud en sí como un closure destinado a manejar la respuesta, dentro de estos métodos construyo el endpoint completo al cual apuntará la solicitud y configuro un URLRequest incorporando los headers necesarios y el método de solicitud adecuado, si se requiere cifro la solicitud, luego empleo URLSession para llevar a cabo un dataTask utilizando el URLRequest, donde al hacer la petición en un closure responde los datos en formato json, un código de respuesta, y un mensaje de error. A continuación, verifico si la respuesta es válida, de ser necesario decodifico el JSON en una estructura de datos si los datos son correctos y, a través de DispatchQueue, garantizo que la respuesta sea manejada en el hilo principal, desde aquí se manda a presentar los datos en la vista

correspondiente.

3. Explica en que condiciones o en que caso práctico utilizarías un XIB, Storyboard, o elementos creados desde programación y describe el porque de cada uno.
 - Para los storyboards los usaría si y sólo si el proyecto fuese pequeño, ya que podría ver el flujo de la aplicación y pantallas de una mejor manera y así tener mayor comprensión del proyecto, pero si fuese un proyecto muy grande o de gran escala sería muy difícil tener orden con muchos storyboards.
 - Para los XIB los utilizaría en dado caso que necesitara crear un componente genérico que fuese reutilizable en cualquier parte del proyecto cosa que con los storyboards no se puede y sumándole la ventaja de que podría ver el componente de manera visual.
 - Usaría los componentes programáticos en un proyecto de gran escala, ya que se puede mantener mejor orden de los componentes y vistas, también es más fácil trabajar con los constraints y animaciones a comparación de un storyboard o un XIB.
4. ¿Haz utilizado un administrador de dependencias como Carthage o Cocoapods? si tu respuesta es si, lista las más representativas o las que más utilices en proyectos antiguos y/o activos y da una breve explicación de su funcionalidad. Si tu respuesta es no, explica como has solucionado la gestión de dependencias y/o frameworks.

Si, he utilizado cocoapods, e aquí una lista de Frameworks utilizados:

- **IQKeyboardManagerSwift:** Lo utilizamos para expandir el funcionamiento del teclado y restringir la instalación de teclados de terceros.
 - **Loties:** para agregar animaciones visuales a tu aplicación de manera efectiva y agradable.
 - **Firebase:**
 - Firebase Auth: para tener una única sesión por usuario.
 - Cloud Firestore: para almacenar, sincronizar, consultar datos y banderas que habilitan/deshabilitan funciones dentro de la app.
 - Performance: para medir el rendimiento de la aplicación.
 - Crashlytics: para tener un control de fallos en la aplicación.
5. ¿Has usado **React programming** o cualquier tipo de Framework React para Swift? De ser cierta tu respuesta explica detalladamente porque lo has usado y qué beneficios obtuviste en el desarrollo de la aplicación.

No he utilizado React programming o algún Framework React dentro de mis desarrollos.

Ejercicio práctico

¿Todavía quieres seguir?... Perfecto. Ahora te pedimos realizar una aplicación IOS Universal en lenguaje swift 3 o posterior. Esta aplicación consta de un simple formulario. El formulario deberá de contener los campos de entrada: Nombre, Apellido Paterno, Apellido Materno, Correo electrónico y Teléfono celular.

Y la aplicación deberá considerar lo siguiente:

- Validar los campos antes de enviar el formulario
 - Al enviar los datos se deberán de guardar en: Base de datos, Archivos locales, etc. (Tú eliges)
 - UX
 - Diseño
 - Usabilidad
 - Estructura de código
 - Diseño responsivo (iPhone/iPad) usando constraints
- Enfatiza tus esfuerzos en la siguiente frase "**Mantén tu código simple**"