# Taller



Aprendiz

Sergio David Rodriguez Robayo

Servicio nacional de aprendizaje (SENA) – Centro de electricidad, electrónica y telecomunicaciones (CEET)

Instructora Isaura María Suarez Novoa

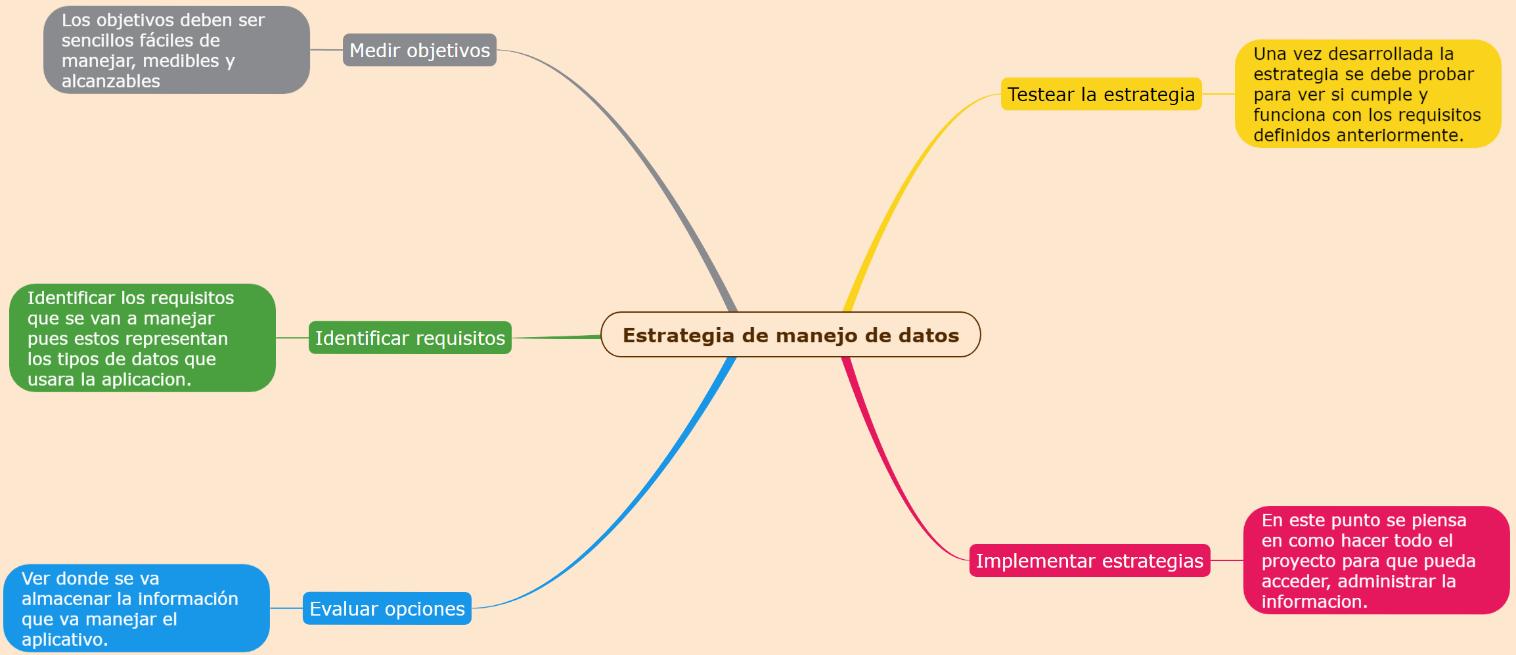
Bogotá D.C

2023

## A través de un mapa mental, explica la estrategia de manejo de datos

Figura 1

*Lo importante de las estrategias de manejo de datos.*



### Escoge una aplicación y explica cada una de las estrategias de obtención de datos

Una aplicación por lo general su manejo de datos por lo general inicia en la obtención de datos del usuario como lo sería el nombre, correo electrónico o fecha de nacimiento etc.

Y así el usuario la podrá utilizar cuando sea necesario cuando cree una cuenta o cuando utiliza la aplicación.

#### Que son los anti-patrones de conectividad. Define y realiza un cuadro con las características más importantes de cada anti-patrón.

Estos anti-patrones hacen que las aplicaciones sean menos eficientes, escalables y segura lo que se considera como ineficaces, pero algunos lo consideran como una buena práctica.

Tabla 1

*Comparación de los anti-patrones más importantes*

|  |  |
| --- | --- |
| Anti-patrón | Característica más importante |
| Anti-patrón proyecto acoplado | El proyecto acoplado no dice que esta unido a un proveedor del servicio de datos lo cual hace que se difícil cambiar de proveedor, y que al momento de modificar o cambiar algo tambien sea difícil. |
| Anti-patrón sobre carga de la red | Este patrón carga muchas solicitudes de la red lo que hace que sea difícil ya que consume muchos recursos y batería y este mas expuesto ataque. |
| Anti-patrón conexión persistente | Mantiene una conexión permanente con la red lo que hace que sufra de los mismos problemas mencionado en el anti-patrón de sobre carga de la red |
| Anti-patrón uso de datos no seguros | La aplicación no usa cifrado para proteger los datos lo que lo vuelve vulnerables ataques. |
| Anti-patrón falta de planificación de la estabilidad. | La aplicación no esta diseñada para escalar lo que hace que su rendimiento y disponibilidad provoque problemas a medida que la aplicación crece. |
| Anti-patrón falta de seguridad | La aplicación no implementa medidas de seguridad suficiente lo que hace que sea vulnerable a ataques |

#### Define los siguientes términos.

* Manejo de datos: Se usa en el desarrollo móvil para almacenar, acceder y administrar los datos de manera segura.
* Datos instalados: Son datos que se almacenan en el dispositivo del usuario, pueden ser datos estáticos como imágenes o dinámicos como configuraciones.
* Cache permanente: Es una forma efectiva de almacenar datos ya que no requiere de una conexión a internet.
* Cache temporal: Es una buena forma de almacenar datos que se utilizan con frecuencia, pero no se usan de manera permanente.
* Sin cache: Los datos se almacenan en un servidor remoto y para acceder a ellos se necesita una conexión a internet.

##### Realiza una infografía del ítem 10.2 del libro en estudio.

Figura 2

*Infografía sobre lo más importante del 10.2*

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

##### ¿Explica porque es tan importante el rendimiento en las aplicaciones móviles? Justifica con bases tu respuesta.

El rendimiento es muy importante en el desarrollo móvil ya que es el principal manejo de las aplicaciones ya que si el rendimiento es malo el usuario sentirá que la usabilidad no es la mejor lo que hará que no use el sistema y que no pueda completar las tareas que tiene planeada hacer con la aplicación y tambien se encarga de la seguridad lo que hace que sea vulnerable por eso es bueno que el rendimiento se use de manera moderada al momento de usar el aplicativo.

##### Define los siguientes términos:

* Performance: Es la capacidad de una aplicación para completar una tarea de manera eficiente.
* Memory boat: Es la acumulación de memoria que no utiliza una aplicación y puede ser causado por utilización de memoria innecesaria.
* Error de dormir: Es un error que sucede cuan do la aplicación que no esta en uso sigue consumiendo recursos como rendimiento y batería.
* Error de inmortalidad: Es una aplicación que no se detiene cuando el usuario la cierra.
* Error de bucle/del circuito de energía: Es cuando una aplicación entra en un bucle que consume más energía de lo normal.
* Usos de API codiciosos de energía: Son API’s que consumen una gran cantidad de energía.
* ANR: Los Application Not Responding es un error que ocurre cuando el usuario hace entradas en un periodo tiempo determinado.
* Retrasos en la GUI: es un retraso en la respuesta de la interfaz cuando el usuario entra hace que pase una mala experiencia.
* Renderizado lento: Es el tiempo que tarda una imagen en mostrar resultado en la pantalla.

##### Realiza un cuadro comparativo entre GC (Recolector de basura) y ARC (Conteo Automático de Referencias).

Tabla 2

*Tabla que representa las principales diferencias entre GC y ARC*

|  |  |  |
| --- | --- | --- |
| Ítem | **GC** | **ARC** |
| **funcionamiento** | El GC rastrea las referencias a objetos en tiempo de ejecución y libera los objetos que ya no son referenciados. | El ARC utiliza un contador de referencias para cada objeto. Cuando el contador de referencias de un objeto llega a cero, el objeto se libera. |
| **Eficiencia** | El GC puede ser más eficiente que el ARC en aplicaciones con un número elevado de objetos pequeños. | El ARC puede ser más eficiente que el GC en aplicaciones con un número elevado de objetos grandes. |
| **Soporte** | El GC es compatible con la mayoría de los lenguajes de programación. | El ARC es compatible con los lenguajes de programación que utilizan un sistema de referencias. |

Nota: Tomado de campusMVP.es (2015)

##### ¿Qué es una referencia fuerte y débil? ¿En qué radica su importancia?

Las referencias fuertes son aquellas que no se eliminan hasta que el usuario pida explícitamente que se tiene que borrar.

Mientras que una referencia débil es aquellas son aquellas que se pueden borrar en cualquier momento desde que ya no se necesiten.

Su importancia radica en que el desarrollador podrá controlar el espacio que se maneja en la aplicación haciendo que las relaciones fuertes sean las que más duran y las débiles las que se pueden borrar una vez ya no se necesitan.

##### Realiza un mapa mental del capítulo 12 del libro en estudio.