



APLICACIONES MOVILES II

*Universidad Politécnica de
Tecámac*

Nombre del profesor:

- Torres Servín Emmanuel

Nombre del alumno:

- Anacleto Ángeles Aritzi Denalyh

Grupo: 1922-IS

INICE DE CONTENIDO

MODELADO DE ACCESO DE DATOS	3
MANIPULACION DE DATOS EN DISPOSITIVOS MOVILES	4
PERSISTENCIA DE DATOS EN DISPOSITIVOS MOVILES	5
MECANISMOS DE TOLERANCIA A FALLOS.....	6

MODELADO DE ACCESO DE DATOS

Es el proceso de analizar y definir todos los diferentes datos que su empresa recopila y produce, y las relaciones entre esos bits de datos. El proceso de modelado de datos crea una representación visual de los datos utilizados en el negocio, y el proceso en sí mismo es un ejercicio para comprender y aclarar los requisitos de los datos.

Conceptual

Un modelo de datos conceptual define la estructura general de su negocio y sus datos. Se utiliza para organizar conceptos comerciales, tal como los definen las partes interesadas comerciales y los arquitectos de datos. Por ejemplo, puede tener datos de clientes, empleados y productos, y cada uno de estos repositorios de datos (llamados entidades) tiene relaciones con otras entidades. Las entidades y las relaciones entre entidades están todas definidas en su modelo conceptual.

Lógico

Un modelo de datos lógicos se fundamenta en el modelo conceptual con atributos específicos de datos en cada entidad e interrelaciones concretas entre aquellos atributos. Este es su modelo técnico de las normas y construcciones de datos definidas por los arquitectos de datos y analistas de negocios, y ayudará a tomar elecciones sobre qué modelo físico necesitan sus datos y necesidades de comercio.

Físico

Un modelo de datos físicos es su utilización específica del modelo de datos lógicos, y lo inventan los administradores de la base de datos y los desarrolladores. Está desarrollado para un instrumento de base de datos específica o tecnología de almacenamiento de datos, y con conectores de

datos para dar los datos en todos los sistemas de comercio a los usuarios según sea primordial. Esta es la “cosa” a la que estuvieron conduciendo lo demás de modelos: la utilización real de su estado de datos.

MANIPULACION DE DATOS EN DISPOSITIVOS MOVILES

La manipulación de datos es el proceso de modificar o perturbar datos para hacerlos más legibles y organizados. Ejemplificando, puede ordenar los datos alfabéticamente para agilizar el proceso de averiguación de información eficaz. Otro ejemplo de manipulación de datos es la administración de sitios web. Los propietarios de sitios web tienen la posibilidad de usar los registros del servidor web para ubicar las páginas web más visitadas, las fuentes de tráfico y muchísimo más. De igual manera, los corredores de bolsa usan la manipulación de datos para adivinar las tendencias del mercado de valores.

La manipulación efectiva de datos consta de cinco pasos clave:

- El primer paso es la extracción de datos de las fuentes de datos.
- Limpiar los datos obtenidos de los sistemas de origen antes de reorganizarlos y reestructurarlos.
- Importe y construya una base de datos que sirva como área de ensayo
- Combine o filtre la información según los requisitos de su negocio.
- Por último, aprovechar los datos manipulados para obtener información valiosa.

PERSISTENCIA DE DATOS EN DISPOSITIVOS MOVILES

En la mayoría de aplicaciones vamos a necesitar guardar datos, para ello vamos a usar persistencia de datos. Son datos que debemos guardar ya sea en el propio dispositivo o en un servidor. Estos datos estarán ahí cuándo el usuario abra de nuevo la aplicación una vez cerrada.

Si la proporción de información que requerimos guardar no es bastante extensa ni tampoco poseemos que hacer consultas o búsquedas tenemos la posibilidad de utilizar APIs de iOS bastante básicas (al menos comparados con los que se utilizan para entrar a bases de datos). Si necesitamos almacenar objetos de clases propias cualesquiera podemos archivarlas, que sería lo que en otros lenguajes se denomina serializar.

Tipos de persistencia

Podemos guardar datos en local o bien en un servidor. Con independencia de respaldo o Backup en un servidor, para guardar datos de forma local podemos utilizar:

- NSUserDefaults
- Carpetas de la SandBox
- MySQL

Tipos de archivo que podemos persistir

Podemos guardar desde un String, hasta una base de datos completa, pasando por diccionarios, arrays, etc. Todo dependerá del modelo de datos de nuestra App.

Para cosas sencillas como puede ser almacenar unos ajustes de usuario, podemos usar la clase NSUserDefaults, y guardar en local la información. Igualmente para cosas sencillas pero un volumen de datos intermedio, podemos usar las carpetas Documents o Preferences. Si nuestra aplicación

requiere de un modelo de datos más sofisticado, siempre podemos usar Core Data.

MECANISMOS DE TOLERANCIA A FALLOS

Tiene su origen en el procesamiento de datos y explica la probabilidad de que un sistema siga en funcionamiento de manera continua inclusive en caso de ingreso errónea o fallo. Referente a la usabilidad, la tolerancia a fallos supone que el cliente puede conseguir su objetivo enmendando entradas o funcionalidades incorrectas con el mínimo esfuerzo. La tolerancia a fallos además pertenece a la regla EN ISO 9241, que especifica criterios para el diseño ergonómico de las interfaces de cliente.

La tolerancia a fallos de las interfaces de cliente se define en el quinto párrafo de la regla EN ISO 9241: "Un cuadro de diálogo es tolerante a errores si el resultado del trabajo previsto puede lograrse con un esfuerzo de corrección mínimo o nulo por parte del cliente pese a los errores de ingreso reconocibles. (Fuente: Ergo-online)" En este entorno, los cuestionarios web, ejemplificando, tienen que comprobarse continuamente para revisar la plausibilidad o totalidad de las entradas. Igualmente, los mensajes de error no tienen que formularse en términos técnicos, sino en una forma principalmente comprensible, por lo cual el cliente logre rastrear el error y evitarlo con poco esfuerzo una vez que vuelva a intentarlo.

La tolerancia a fallos puede ser necesaria tanto para el hardware como para el software. Con el hardware, normalmente se instala una secuencia paralela que puede detectar errores. Un tercer componente lo arregla.

Hay varias áreas de software que pueden ser corregidas por la tolerancia frente a errores:

- Errores de diseño.

- Error en los datos.
- Error en la secuencia cronológica.

En lo que respecta a la usabilidad, lo decisivo es sobre todo la tolerancia a fallos de las interfaces de usuario, como las páginas web. Los siguientes elementos pueden aumentar la tolerancia frente a errores:

- Información sobre posibles entradas erróneas.
- Explicación de los errores para que el usuario pueda corregirlos.
- Visualizar dónde se ha producido el error.
- Corrección automatizada con sugerencias.