

Materia: Desarrollo de Aplicaciones Móviles
Grupo XB Ene-Jun 2023
Profesor: Mario Humberto Tiburcio Zúñiga

Atributo de egreso a evaluar 6.3

Desarrolla soluciones computacionales que impliquen la aplicación de diversas metodologías, herramientas y tecnologías actuales de la computación para la solución de problemas reales.

Tema 4

La unidad se evalúa con tres actividades que implican el desarrollo de tres aplicaciones móviles o apps que resuelven problemas típicos de la vida real integrando diversas tecnologías actuales.

Puesto que todas las actividades constituyen aplicaciones por desarrollar, éstas deberán presentarse funcionando correctamente. De no funcionar una actividad, ésta no generará puntaje en su evaluación. En caso de funcionar, el puntaje asignado dependerá de la puntualidad en la entrega, de acuerdo a la rúbrica mostrada. Las revisiones se realizan personalmente y una vez aprobadas se entregará un reporte incluyendo Nombre de la actividad, Definición del problema u Objetivo, Evidencia de ejecución satisfactoria y conclusiones.

Descripción de actividades:

Actividad 1

Desarrollar una aplicación que realice operaciones sobre una base de datos remota instalada en internet. El manejador de base datos es PostgreSQL y tendrá creada una base de datos con una tabla de datos de máximo 4 campos, uno de los cuales será un id como llave única. La aplicación realizará las operaciones de consulta individual, consulta general e inserción de los registros de la tabla mencionada vía protocolo HTTP. El uso del protocolo HTTP es para permitir a la aplicación solicitar el servicio de rutinas escritas en PHP del lado del servidor, las cuales realizarán las operaciones de inserción y consultas mencionadas. La aplicación debe validar el mecanismo de inserción de manera que visualice mensajes de notificación Toast en caso de éxito o fracaso en la operación.

Actividad 2

Desarrollar una aplicación que implemente un CRUD en una base de datos embebida usando el manejador SQLite. La base de datos debe ser

compatible con la estructura y campos a la indicada en la Actividad 1. La app debe tener la funcionalidad de sincronizar los datos de SQLite con los de la base de datos remota. La interfaz debe ser similar a la que se muestra en la figura siguiente:

APLICACIÓN CON SQLITE

Funciones:

- C – Crear registro (insertar)
- R – Leer registro (consulta individual a partir de un ID dado con salida en los EditText)
- R – Leer registros (consulta general con click largo, desplegando en otra pantalla la salida con ScrollView)
- U – Modificar registro (excepto ID)
- D – Borrar registro
- Camarita: Para capturar ID desde código de barras o QR.
- Círculo con cruz: Borra formulario.

Requisitos:

Se puede usar pantalla con orientación fija, landscape o portrait. El color y forma de los íconos, botones, etc. es al gusto del desarrollador

Validar todas las operaciones con mensajes Toast informativos para el usuario. Por ejemplo cuando se intenta insertar un registro con ID repetido. O cuando se desea consultar un registro inexistente. Capturar campos vacíos. Etc. No mostrar mensajes tipo Exception, que son crípticos para el usuario.

La aplicación deberá tener desactivados por defecto los botones U y D, activándose éstos hasta que se visualicen los datos de una consulta individual. Esto es, que para modificar o borrar un registro, antes se debe consultar y visualizar sus campos en los EditText de la aplicación.

Cada alumno seleccionará su propia tabla en su base de datos, que no sea “animales” ni “dioses”. Es obligatorio usar un ID como llave primaria.



Actividad 3

Desarrollar una aplicación que utilice la librería Retrofit para consumir un servicio REST. EL servicio REST debe implementarse en backend en PHP de manera que devuelva en formato JSON la consulta general de de la tabla de la base de datos remota de la Actividad 1. La app a implementar debe desplegar en el móvil la consulta correspondiente.

Rúbrica para evaluación de actividades y tema 4

ACTIVIDAD1	Entrega En fecha 100%	Entrega Un día hábil después 90%	Entrega tres días hábiles después 80%	Entrega cinco días Hábiles después 70%	No Entregó en tiempo 0%	Porcentaje 1 (P1)
ACTIVIDAD2	Entrega En fecha 100%	Entrega Un día hábil después 90%	Entrega tres días hábiles después 80%	Entrega cinco días Hábiles después 70%	No Entregó en tiempo 0%	Porcentaje 2 (P2)
ACTIVIDAD3	Entrega En fecha 100%	Entrega Un día hábil después 90%	Entrega tres días hábiles después 80%	Entrega cinco días Hábiles después 70%	No Entregó en tiempo 0%	Porcentaje 3 (P3)
					Final	(P1+P2+P3)/3

Evidencias de 3 alumnos evaluados:

Caso: Excelente
Alumno : Ricardo Bárcenas Dirzo
Actividad 1

Tarea 11

Inserción, consulta individual y consulta general a base de datos con PHP en backend

Alumno: **Ricardo Bárcenas Dirzo**

N.C. **19091266**

Asignatura: **Desarrollo de aplicaciones móviles**

Unidad: **4** Grupo: **XB**

Docente: **Marío Humberto Tiburcio Zúñiga**

Objetivo

Realizar las operaciones de inserción, consulta individual y consulta general a una base de datos sql, utilizando la sintaxis en PHP.

Software en PHP

Las **ráficas en PHP** permiten realizar operaciones en una base de datos remota mediante peticiones web. Para ello, debemos tener estas ráficas en algún servidor, o en el mismo donde está alojado la base de datos.

Para este proyecto le requirirán las operaciones de inserción

```
1<?php
2
3// Conexión a la base de datos
4
5$host = "localhost";
6$username = "root";
7$password = "";
8
9// Crear conexión a la base de datos
10$conn = mysqli_connect($host, $username, $password);
11
12// Verificar conexión
13if ($conn === false) {
14    die("Error de conexión a la base de datos: " . mysqli_connect_error());
15}
16
17// Crear base de datos
18$db = "base de datos";
19
20// Crear base de datos si no existe
21if (!mysqli_query($conn, "CREATE DATABASE IF NOT EXISTS $db;")) {
22    die("Error al crear la base de datos: " . mysqli_error($conn));
23}
24
25// Seleccionar la base de datos
26mysqli_select_db($conn, $db);
27
28// Crear tabla
29
30// Crear tabla si no existe
31if (!mysqli_query($conn, "CREATE TABLE IF NOT EXISTS `usuarios` (
32  `id` int(11) NOT NULL AUTO_INCREMENT,
33  `nombre` varchar(50) NOT NULL,
34  `apellido` varchar(50) NOT NULL,
35  `email` varchar(100) NOT NULL,
36  `password` varchar(50) NOT NULL,
37  PRIMARY KEY (`id`)
38);")) {
39    die("Error al crear la tabla: " . mysqli_error($conn));
40}
41
42// Insertar datos
43
44// Insertar un usuario
45$nombre = "Juan";
46$apellido = "Perez";
47$email = "juan.perez@example.com";
48$password = "12345678";
49
50// Insertar datos en la tabla
51if (!mysqli_query($conn, "INSERT INTO `usuarios` (`nombre`, `apellido`, `email`, `password`) VALUES ($nombre, $apellido, $email, $password);")) {
52    die("Error al insertar el usuario: " . mysqli_error($conn));
53}
54
55// Consultar datos
56
57// Consultar un usuario
58$id = 1;
59
60// Consultar datos en la tabla
61if (!mysqli_query($conn, "SELECT * FROM `usuarios` WHERE `id` = $id;")) {
62    die("Error al consultar el usuario: " . mysqli_error($conn));
63}
64
65// Consultar todos los usuarios
66if (!mysqli_query($conn, "SELECT * FROM `usuarios`;")) {
67    die("Error al consultar todos los usuarios: " . mysqli_error($conn));
68}
69
70// Mostrar resultados
71
72// Mostrar un usuario
73if ($result = mysqli_query($conn, "SELECT * FROM `usuarios` WHERE `id` = $id;")) {
74    if (mysqli_num_rows($result) > 0) {
75        // Obtener un usuario
76        $usuario = mysqli_fetch_assoc($result);
77        echo "Usuario encontrado: " . $usuario['nombre'] . " " . $usuario['apellido'] . " (ID: " . $usuario['id'] . ")";
78    } else {
79        echo "No se encontró el usuario con el ID: " . $id;
80    }
81}
82
83// Mostrar todos los usuarios
84if ($result = mysqli_query($conn, "SELECT * FROM `usuarios`;")) {
85    if (mysqli_num_rows($result) > 0) {
86        // Obtener todos los usuarios
87        while($usuario = mysqli_fetch_assoc($result)) {
88            echo "Usuario: " . $usuario['nombre'] . " " . $usuario['apellido'] . " (ID: " . $usuario['id'] . ")";
89        }
90    } else {
91        echo "No se encontraron usuarios";
92    }
93}
94
95// Cerrar conexión
96mysqli_close($conn);
97
98// Salir del script
99exit;
```

Consulta individual

```
1<?php
2
3// Conexión a la base de datos
4
5$host = "localhost";
6$username = "root";
7$password = "";
8
9// Crear conexión a la base de datos
10$conn = mysqli_connect($host, $username, $password);
11
12// Verificar conexión
13if ($conn === false) {
14    die("Error de conexión a la base de datos: " . mysqli_connect_error());
15}
16
17// Crear base de datos
18$db = "base de datos";
19
20// Crear base de datos si no existe
21if (!mysqli_query($conn, "CREATE DATABASE IF NOT EXISTS $db;")) {
22    die("Error al crear la base de datos: " . mysqli_error($conn));
23}
24
25// Seleccionar la base de datos
26mysqli_select_db($conn, $db);
27
28// Crear tabla
29
30// Crear tabla si no existe
31if (!mysqli_query($conn, "CREATE TABLE IF NOT EXISTS `usuarios` (
32  `id` int(11) NOT NULL AUTO_INCREMENT,
33  `nombre` varchar(50) NOT NULL,
34  `apellido` varchar(50) NOT NULL,
35  `email` varchar(100) NOT NULL,
36  `password` varchar(50) NOT NULL,
37  PRIMARY KEY (`id`)
38);")) {
39    die("Error al crear la tabla: " . mysqli_error($conn));
40}
41
42// Insertar datos
43
44// Insertar un usuario
45$nombre = "Juan";
46$apellido = "Perez";
47$email = "juan.perez@example.com";
48$password = "12345678";
49
50// Insertar datos en la tabla
51if (!mysqli_query($conn, "INSERT INTO `usuarios` (`nombre`, `apellido`, `email`, `password`) VALUES ($nombre, $apellido, $email, $password);")) {
52    die("Error al insertar el usuario: " . mysqli_error($conn));
53}
54
55// Consultar datos
56
57// Consultar un usuario
58$id = 1;
59
60// Consultar datos en la tabla
61if (!mysqli_query($conn, "SELECT * FROM `usuarios` WHERE `id` = $id;")) {
62    die("Error al consultar el usuario: " . mysqli_error($conn));
63}
64
65// Consultar todos los usuarios
66if (!mysqli_query($conn, "SELECT * FROM `usuarios`;")) {
67    die("Error al consultar todos los usuarios: " . mysqli_error($conn));
68}
69
70// Mostrar resultados
71
72// Mostrar un usuario
73if ($result = mysqli_query($conn, "SELECT * FROM `usuarios` WHERE `id` = $id;")) {
74    if (mysqli_num_rows($result) > 0) {
75        // Obtener un usuario
76        $usuario = mysqli_fetch_assoc($result);
77        echo "Usuario encontrado: " . $usuario['nombre'] . " " . $usuario['apellido'] . " (ID: " . $usuario['id'] . ")";
78    } else {
79        echo "No se encontró el usuario con el ID: " . $id;
80    }
81}
82
83// Mostrar todos los usuarios
84if ($result = mysqli_query($conn, "SELECT * FROM `usuarios`;")) {
85    if (mysqli_num_rows($result) > 0) {
86        // Obtener todos los usuarios
87        while($usuario = mysqli_fetch_assoc($result)) {
88            echo "Usuario: " . $usuario['nombre'] . " " . $usuario['apellido'] . " (ID: " . $usuario['id'] . ")";
89        }
90    } else {
91        echo "No se encontraron usuarios";
92    }
93}
94
95// Cerrar conexión
96mysqli_close($conn);
97
98// Salir del script
99exit;
```

Actividad 2









CRUD con SQLite y sincronización

Tarea 12

Alumno:
 Ricardo Sánchez Díaz
 N.C.
 19071244

Asignatura:
 Desarrollo de aplicaciones móviles
 Inicial Grupo
 4 XB
 Docente:
 Mario Humberto Tiburcio Zúñiga

Fecha de entrega:
 31 de mayo del 2023

Desarrollo de aplicaciones móviles

[illegible]

Actividad 3

UNIVERSIDAD DE BARRIOCENOS DIZO		UNIVERSITY OF BARRIOCEÑOS DIZO	
Tarea 13		UNIVERSITY OF BARRIOCEÑOS DIZO	
Consulta general usando GET con Retrofit		UNIVERSITY OF BARRIOCEÑOS DIZO	
Alumno: Ricardo Bárcenas Dizo		UNIVERSITY OF BARRIOCEÑOS DIZO	
N.C. 19091266		UNIVERSITY OF BARRIOCEÑOS DIZO	
Asignatura: Desarrollo de aplicaciones móviles		UNIVERSITY OF BARRIOCEÑOS DIZO	
Unidad: 4	Grupo: XB	UNIVERSITY OF BARRIOCEÑOS DIZO	
Docente: María Humberto Tiburcio Zúñiga		UNIVERSITY OF BARRIOCEÑOS DIZO	
Fecha de entrega: 31 de agosto del 2023		UNIVERSITY OF BARRIOCEÑOS DIZO	

Clase Productos

En esta clase se declaran variables de acuerdo con los campos de la tabla de la base de datos. Las variables deben tener el mismo nombre que los campos de la tabla. Después se crean sus respectivos métodos get:

```
1 public class Productos {
2     //campo
3     private int id;
4     //campo
5     private int precio;
6     //campo
7     private String nombre;
8     //campo
9     private String tipo;
10
11     //campo
12     public int getId() {return id;}
13
14     //campo
15     public int getPrecio() {return precio;}
16
17     //campo
18     public String getNombre() {return nombre;}
19
20     //campo
21     public String getTipo() {return tipo;}
22 }
```

Evaluación:

ACTIVIDAD1	Entrega En fecha 100%	Entrega Un día hábil después 90%	Entrega tres días hábiles después 80%	Entrega cinco días Hábiles después 70%	No Entregó en tiempo 0%	Porcentaje 1 (P1) 100
ACTIVIDAD2	Entrega En fecha 100%	Entrega Un día hábil después 90%	Entrega tres días hábiles después 80%	Entrega cinco días Hábiles después 70%	No Entregó en tiempo 0%	Porcentaje 2 (P2) 100
ACTIVIDAD3	Entrega En fecha 100%	Entrega Un día hábil después 90%	Entrega tres días hábiles después 80%	Entrega cinco días Hábiles después 70%	No Entregó en tiempo 0%	Porcentaje 3 (P3) 100
					Final	100

Caso medio

Alumna: BETZAIDA AMARILIS ROJAS JIMÉNEZ

Actividad 1



INSTITUTO TECNOLÓGICO DE ZACATEPEC

Tarea 11
Inserción, Consulta individual y Consulta general a Base de Datos con PHP en backend

Materia: Desarrollo de aplicaciones móviles

Docente: Tiburcio Zúñiga Mario Humberto

Ingeniería en sistemas computacionales

Alumna:

Rojas Jiménez Betzaida Amarilis C19090604

Grupo: XB

Zacatepec, Morelos a de 23 mayo del año 2023.

Pruebas de ejecución

EjemploConexionHttp

Consulta General

Consulta individual

Introduce ID

Introduce Producto

Introduce Código

Introduce Precio

AGREGAR

TextView

Actividad 2



INSTITUTO TECNOLÓGICO DE ZACATEPEC

Tarea 12
CRUD con SQLite y sincronización

Materia: Desarrollo de aplicaciones móviles

Docente: Tiburcio Zúñiga Mario Humberto

Ingeniería en sistemas computacionales

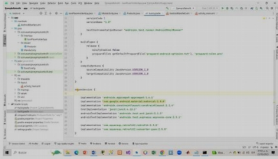
Alumna:

Rojas Jiménez Betzaida Amarilis C19090604

Grupo: XB

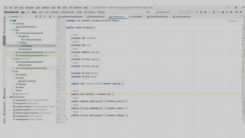
Zacatepec, Morelos a de 23 mayo del año 2023.

Se tendrá que ejecutar la aplicación para que se descarguen las librerías de retrofit a nuestra aplicación y así se puedan usar en el desarrollo de esta.



Después de ello creamos una clase llamada Producto, esta contendrá todos los atributos de nuestro JSON que es id, código, nombre y precio.

Se crearán sus getter y setter igual que su método constructor.



Actividad 3



INSTITUTO TECNOLÓGICO DE ZACATEPEC

Tarea 13
Consulta general usando GET con Retrofit

Materia: Desarrollo de aplicaciones móviles

Docente: Tiburcio Zúñiga Mario Humberto

Ingeniería en sistemas computacionales

Alumna:

Rojas Jiménez Betzaida Amarilis C19090604

Grupo: XB

Zacatepec, Morelos a de 23 mayo del año 2023.

Pruebas de ejecución



Se observan los registros de la base de datos la cual hemos estado utilizando en prácticas pasadas.

Con esto comprobamos el buen funcionamiento de nuestra aplicación utilizando las dependencias de retrofit.

Evaluación:

ACTIVIDAD1	Entrega En fecha 100%	Entrega Un día hábil después 90%	Entrega tres días hábiles después 80%	Entrega cinco días Hábiles después 70%	No Entregó en tiempo 0%	Porcentaje 1 (P1) 90
ACTIVIDAD2	Entrega En fecha 100%	Entrega Un día hábil después 90%	Entrega tres días hábiles después 80%	Entrega cinco días Hábiles después 70%	No Entregó en tiempo 0%	Porcentaje 2 (P2) 70
ACTIVIDAD3	Entrega En fecha 100%	Entrega Un día hábil después 90%	Entrega tres días hábiles después 80%	Entrega cinco días Hábiles después 70%	No Entregó en tiempo 0%	Porcentaje 3 (P3) 90
					Final	80

Caso malo:

Alumno: Emmanuel Patiño Romero

Evaluación: No entregó trabajos

ACTIVIDAD1	Entrega En fecha 100%	Entrega Un día hábil después 90%	Entrega tres días hábiles después 80%	Entrega cinco días Hábiles después 70%	No Entregó en tiempo 0%	Porcentaje 1 (P1) 0
ACTIVIDAD2	Entrega En fecha 100%	Entrega Un día hábil después 90%	Entrega tres días hábiles después 80%	Entrega cinco días Hábiles después 70%	No Entregó en tiempo 0%	Porcentaje 2 (P2) 0
ACTIVIDAD3	Entrega En fecha 100%	Entrega Un día hábil después 90%	Entrega tres días hábiles después 80%	Entrega cinco días Hábiles después 70%	No Entregó en tiempo 0%	Porcentaje 3 (P3) 0
					Final	0