TEC | Tecnológico de Costa Rica

BASES DE DATOS | CE – 3101

TEC Bank

Documentación

Estudiantes:

Saúl Gómez Ramírez

Esteban Morales Ureña

Steven Castro Medina

Profesor:

Marco Rivera Menese

SII-2020

I. Descripción de los métodos implementados

Aplicación móvil:

Para el desarrollo de la aplicación móvil TEC Bank se utilizó la herramienta Android Studio, la cual se especializa en la fabricación de aplicaciones para el sistema operativo Android. Los métodos implementados en el desarrollo fueron:

onCreate(): Encargado de iniciar actividades en Android, para poder ver pantallas, funciones, vistas, textos entre otros.

addUsers(): Este método se encarga de enviar el request POST al REST API, mediante una serialización de ítems que contiene el JSON, el cual parsea y procede a enviar gracias a la librería Retrofit

getAccount(): Este método se encarga de enviar el request GET al REST API, donde obtiene las cuentas agregadas según el cliente asociado

verCuenta():Este método permite mostrar la cuenta al usuario en la pantalla.

REST API:

El REST API fue desarrollado utilizando .NET CLI con el SDK de .NET versión 5.0, puede correrse directamente en la terminal pero para efectos de comodidad con extensiones y dependencias se eligió Visual Studio Code. El REST API contiene las siguientes funcionalidades, tanto para la lista de clientes como la de cuentas:

GetCliente(id): GET request, obtiene un cliente escrito dentro del api en ese momento, según la id especificada

GetClientes(): GET request, obtiene la lista de todos los clientes registrados en la aplicación

CreateCliente(): POST request, crea un nuevo cliente con la información enviada a través del request

UpdateCliente(): PUT request, actualiza la información de un cliente solicitado por ID

DeleteCliente(): DELETE request, borra la información de un cliente según la id especificada

Los métodos se escriben igual para las cuentas, simplemente se reemplaza “Cliente” por “Cuenta”

II. Descripción de las estructuras de datos desarrolladas

Aplicación móvil:

La aplicación móvil TEC Bank utiliza diversas estructuras de datos, tanto para administrar las vistas como para realizar conexiones con el REST API. Esta estructura es:

- Interfaces: La aplicación aplicó una estructura de datos basándose en una Interfaz para correr los protocolos GET y POST, y poder llamarlos desde cualquier clase y utilizar los métodos correspondientes

REST API:

Para guardar la información internamente, el REST API utiliza el patrón de repositorio, este patrón oculta los detalles acerca de cómo son guardados o recuperados los datos exactamente, es un patrón muy útil para cuando los datos son guardados internamente en la memoria del programa en lugar de en una base de datos o algo como un XML.

III. Descripción detallada de los algoritmos desarrollados

Aplicación móvil:

Los algoritmos desarrollados en este apartado no son complejos debido a que no hay necesidad de implementar soluciones con parámetros muy específicos. El algoritmo más sobresaliente es el encargado de administrar los depósitos a otras cuentas, donde se realizan operaciones matemáticas regidas antes por algunas validaciones.

REST API:

El API no maneja algoritmos complejos debido a la simplicidad de la aplicación, los métodos de comunicación del REST API pasan los datos a un DTO (Data Transfer Object) para ser enviados al exterior (GET request) y recuperan los datos en un DTO para desplegarlos en el repositorio (POST request)

IV. Problemas conocidos (sin solución)

No se puede ingresar a la aplicación móvil sin haber antes registrado, esto debido a un exception generado por un nullPointer. Si bien sí se puede resolver, contemplaría una base de datos la cual no se hace en esta tarea.

No se da la solución total a la aplicación Web, esto por problemas en el desarrollo que se presentaron por desconocimiento del Framework a utilizar, y la mala gestión de tiempo que se empleó en este apartado por parte de su encargado (Esteban Morales U.).

V. Problemas encontrados

Problema	Descripción	Intentos de solución sin éxito	Soluciones encontradas	Recomendaciones	Conclusiones
Conexión API - AppMóvil	Este problema no dejaba que la aplicación móvil hiciera POST en el REST-API	3	Desactivar el Firewall de windows y agregar Cleartext al android manifest	Preparar todos los elementos para que puedan haber conexiones entre dispositivos	Es importante probar diversos métodos exteriores al código para que funcione una conexión
Pasar datos entre activities	El programa de la aplicación móvil no me permitía agregar datos entre activities	1	Se encontró el método putExtra, que permite pasar datos entre activities	Ver documentación relacionada al paso de datos	Es importante buscar información sobre métodos nativos de herramientas tales como Android Studio
Finalización de la aplicación web	Por recurrentes errores en la implementación de los componentes a usar en la app Web, se da un producto incompleto.	4	La implementación de Angular Material con sus componentes necesarios.	Recolección de experiencias externas desarrollando en este framework, que retroalimenten el trabajo antes de una fallida realización	La importancia de conocer e investigar la correcta ruta de desarrollo de un trabajo para concluir oportunamente el producto.

VI. Actividades planeadas, duración, su responsable y fecha de entrega

	Actividades	Duración	Fecha de entrega
Aplicación Web (Esteban)	**Gestión de roles:** El sistema debe permitir ingresar, modificar y eliminar roles, los cuales solamente almacenarán un nombre y descripción.	5h	30/08
	Gestión de cuentas: El sistema debe permitir ingresar, modificar y eliminar Cuentas, de las cuales se debe almacenar: número de cuenta, descripción de la cuenta, moneda (Colones, Dólares, Euros), tipo de cuenta (Ahorros, Corriente) y el cliente al que pertenece	5h	6/09
	Gestión de Tarjetas: El sistema debe permitir ingresar y modificar tarjetas, de las cuales se guarda: el número de tarjeta, tipo de tarjeta (Débito o crédito), fecha de expiración, código de seguridad y saldo disponible/monto de crédito disponible	5h	11/09
	Conexión API: El sistema debe realizar request de los datos asociados a clientes y administradores	5h	13/09
Aplicación móvil (Saúl)	**Login:** El sistema debe proveer la funcionalidad de loguearse a cada uno de los clientes por medio de un usuario y password.	2h	30/08
	Cuentas: Ver movimientos de mis cuentas: el sistema debe proveer un detalle de todos los movimientos que se han realizado a una cuenta (Depósitos-Retiros).Así mismo podrá observar las tarjetas de débito asociadas a la cuenta	5h	11/09
	Cuentas: Realizar transferencia de dinero a otra cuenta. El sistema debe proveer la opción de realizar una transferencia de efectivo a otra cuenta.	3h	11/09
	Conexión API: El sistema debe realizar request de los datos asociados a clientes y administradores	5h	6/09
	Gestión de Clientes: El sistema debe permitir ingresar, modificar y eliminar clientes, Se debe almacenar: nombre completo, cédula, dirección, teléfono, ingreso mensual y tipo de cliente (Físico, Jurídico), usuario y password.	5h	2/09
REST API (Steven)	**Gestion de métodos HTTP:** El sistema debe ser capaz de gestionar diferentes tipos de solicitudes tales como POST o GET	5h	2/09
	Gestión de cuentas: El sistema debe ser capaz de alojar cuentas con sus correspondientes datos en formato JSON	3h	11/09
	Gestión de clientes: El sistema debe ser capaz de alojar clientes con sus correspondientes datos en formato JSON	3h	13/09
	Alojamiento en IIS: El REST API debe estar alojado en el sistema de Administración de Internet Information Services	8h	6/09

VII. Minutas de sesiones de trabajo

#Reunión	*Fecha*	*Hora inicial*	*Hora final*	*Integrantes*
1	27/08/2021	19:00	19:30	Saúl Gómez Ramírez Steven Castro Medina

				Esteban Morales Ureña
Objetivos		*Desarrollo*		*Conclusión*
- Realizar de manera conjunta el plan de trabajo. - Agendar la siguiente reunión.		Se realizó el listado de tareas del punto anterior		Se da por hecho las tareas que tiene cada miembro del grupo y se empieza a trabajar

#Reunión	*Fecha*	*Hora inicial*	*Hora final*	*Integrantes*
2	3/09/2021	19:00	22:00	Saúl Gómez Ramírez Steven Castro Medina Esteban Morales Ureña
Objetivos		*Desarrollo*		*Conclusión*
- Se pretende conectar aplicación móvil con REST API - Agendar la siguiente reunión.		Desafortunadamente no se pudo conectar el app con el REST API, y se dejó para la siguiente reunión para no atrasar demás tareas		Se sigue con demás tareas mientras se encuentra solución a la conexión

#Reunión	*Fecha*	*Hora inicial*	*Hora final*	*Integrantes*
3	8/09/2021	19:00	19:30	Saúl Gómez Ramírez Steven Castro Medina Esteban Morales Ureña Marco Rivera Meneses
Objetivos		*Desarrollo*		*Conclusión*
- Se pretende conectar aplicación móvil con REST API - Agendar la siguiente reunión.		Gracias a la ayuda del profesor en esta reunión, se entendió mejor que es el IIS y para qué servía		Si bien no se hizo ninguna conexión, esta reunión sirvió para saber que es el IIS y su importancia como servidor

#Reunión	*Fecha*	*Hora inicial*	*Hora final*	*Integrantes*
4	10/09/2021	19:00	22:00	Saúl Gómez Ramírez Steven Castro Medina

Objetivos	*Desarrollo*	*Conclusión*
- Se pretende conectar aplicación móvil con REST API - Finalizar detalles	Finalmente existió conexión entre API y aplicación, se terminaron las tareas correspondientes	Se terminan las tareas correspondientes a la conexión como serían clientes y cuentas

VIII. Bitácora

Integrante	Actividad realizada	Fecha
Steven	Se comienza a desarrollar el API, probándolo con la interfaz de swagger	22/08
Steven	Se termina un modelo básico de API con los métodos básicos de comunicación, se muestra a los compañeros	29/08
Saul	Empiezo a trabajar en la aplicación agregando vistas como inicio de sesión	30/08
Esteban	Realiza investigación preliminar sobre el framework a utilizar	30/8
Saul	Se termina registro e inicio de sesión	3/09
Esteban	Recolecta documentación necesaria.	5/09
Steven	Se logra hostear el REST API en microsoft IIs	7/09
Saul	Se conecta la app con API	10/09
Esteban	Se implementan códigos de testos al que se quiere como producto final.	11/09
Steven	Se genera la funcionalidad para registrar clientes y cuentas	12/09
Saul	Se termina la sección cuentas con su correspondiente conexión	11/09

IX. Conclusiones

Pase a no dar un solución total al problema planteado se sabe que este proyecto es un paso para poder desarrollar BD más adelante, por lo tanto fue de

suma importancia para aprender a utilizar diversas herramientas necesarias para conectar sistemas y exista comunicación entre estas. Gracias a la realización de TEC Bank, se conocieron los primeros cimientos de lo que sería un FrontEnd, un elemento importante para poder implementar BackEnd más adelante con otros proyectos.

X. Recomendaciones del proyecto

Es recomendable buscar soluciones que no involucren el uso de código solamente cuando se esté desarrollando conexiones vía internet tales como las que se usaron en la conexión REST API, porque hay factores externos que pueden bloquear la información receptora, como el Firewall de Windows o especificaciones del navegador predeterminado.

XI. Bibliografía

How to get the response from Retrofit POST request. (2018, 18 octubre). Stack Overflow.https://stackoverflow.com/questions/52880160/how-to-get-the-response-from-retrofit-post-request/52880314

Consuming APIs with Retrofit | CodePath Android Cliffnotes. (s. f.). CODEPATH. Recuperado 16 de septiembre de 2021, de https://guides.codepath.com/android/consuming-apis-with-retrofit

Ramos, J. (s. f.). Android: Aprende a consultar una API y procesar la respuesta (con Retrofit). Programación y más. Recuperado 16 de septiembre de 2021, de https://programacionymas.com/blog/consumir-una-api-usando-retrofit