**Onboarding Practico**

|  |  |
| --- | --- |
| **Información del Recurso** | |
| **Perfil Back o Front** | Back |
| **Nombre:** | Omar Centeno |
| **Email personal:** | omarstalin1990@gmail.com |
| **Email banco:** | ocenteno@pichincha.com |
| **Computador (código)** | CB.401395. |
| **Tribu o Célula** | Empresas/CASH MANAGEMENT NEGOCIO |
| **Información del Documento** | |
| **Tutor:** | Christian Yepez |
| **Fecha Inicio OnBoarding** | 22/08/2022 |
| **Fecha Fin Onboarding** | 29/08/2022 |

# Informe de parte de Colaborador

Luego de haber conversado inicialmente con mi tutor Christian Yépez, quien me ha dado las indicaciones necesarias para poder desarrollar correctamente el ejercicio se ha procedido a implementar el aplicativo.

En primera instancia cloné el arquetipo que está alojado en el repositorio de GitHub, y mediante este proyecto implementé todo lo solicitado.

Se revisó que se cuente con todas las herramientas necesarias para la correcta implementación del aplicativo y también se hizo una revisión de todas las historias de usuario solicitadas.

Mediante el análisis realizado he diseñado la base de datos y se ha creado las tablas en SQL Server, las tablas que he considerado necesarias son: Cliente, Ejecutivo, Marca, Patio, Solicitud y Vehículo.

Siguiendo la estructura de capas que contiene el arquetipo, en la capa arquetipo.API he modificado la cadena de conexión en el archivo appsettings para que apunte a mi base de datos local llamada CreditoVehicularBancoPichincha y también se ha creado en esta misma capa todos los controladores necesarios para cada entidad.

Después en la capa arquetipo.Entity se ha creado las entidades correspondientes a cada tabla con todos los campos requeridos y en la capa arquetipo.Repository se ha añadido a la clase contexto todas las definiciones de estas entidades.

En la capa arquetipo.Domain están todas las interfaces de cada entidad con todas las definiciones de cada método respectivo, los cuales están implementados en la capa arquetipo.Infraestructure.

En resumen, se han creado los métodos para la carga inicial para clientes, marcas y ejecutivos, seguido de los Crud para cliente, vehículo, patio y solicitud y los métodos de asignación de cliente y patio respectivo.

Para comprobar estos métodos se lo ha realizado mediante swagger que ya viene implementado en el mismo proyecto.

El punto más crítico para mí ha sido el desarrollo de las pruebas unitarias, pero con un poco de investigación se ha intentado poner en practica esta teoría, para lo cual se han creado pruebas unitarias para validar los Cruds de cliente, patio, vehiculo, solicitud, ejecutivo y marca, esto se llevó a cabo aplicando la teoría de mocks.

# Observaciones Generales

* Se debe detallar estado actual de colaborador versus el estado inicial luego del Onboarding.
* Se debe detallar los GAPS, o los puntos de mejora.

# Compromiso

* Definir fechas para monitorear los puntos de mejora.