### **GUIA TECNICA PROYECTO:**

API: Servicio Rest - Django Rest Frameowork

App: Cliente en React JS

info@quallitynet.cl

# Introducción

6 de Octubre del 2021

# **Necesidades**

La empresa requería el desarrollo de una herramienta que permitiera la administración de datos de clientes y cómo estos interactúan con sus servicios para finalmente entrar al proceso de pago.

## Arquitectura - Servicio - Cliente

El proyecto fue dividido en 3 etapas;

Arquitectura: Servidores que levantaran nuestros servicios creados

Servicio: Microservicios que ayudarán a transaccionar información desde el cliente al servidor.

Cliente: Es lo que conocemos como interfaz de usuario.

# Sistema de planificación de recursos empresariales

Arquitectura(<u>"Docker"</u>):

Para la integración progresiva de nuestro proyecto utilizaremos logista basada en contenedores, para ello nuestra tecnología de cabecera será <u>"Docker"</u> la cual nos permitirá desplegar las imágenes y volúmenes necesarios sin sufrir eternos periodos de integración apoyándonos en <u>Docker Hub</u>.

Pyhon - Django:

La implementación de micro servicios fue desarrollada con <u>Django</u> en el cual tenemos un acceso administrativo para la modificación de datos en duro por así decirlo las credenciales solo serán entregadas a CoworkChillan y administradas por Quality Net.

Módulos que proporciona el administrador:

Tokens(llaves de seguridad y accesos)

#### Bookings

Books: Reservas

- Calendars: gestión de calendario para el ingreso de reservas

- Tasks: tareas para planificación de reservas

#### Clients

- Business: Registro de empresas

- Persons: Personas naturales, representantes legales y personas asociadas a una empresa

#### Contracts

- Contracts: Contratos dirigidos a empresas o persona natural

- Type: Tipos de contratos para establecer versiones de estos mismos

#### Services

- Categories: Categorías de servicios

- Services: Creación de servicios y sus detalles

- Valorations: Valoraciones de Servicios

#### Memberships

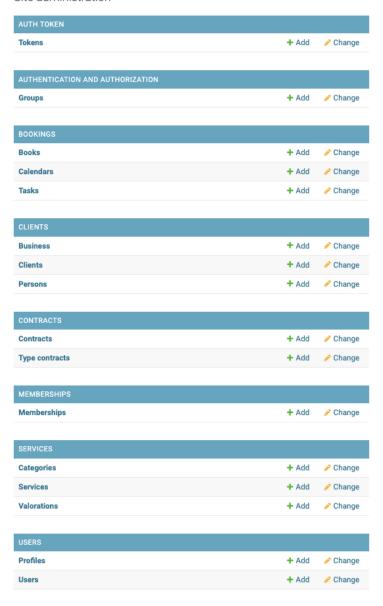
- Memberships: creación de membresías a raíz de un cliente

Estos módulos contemplan su usabilidad de forma general, esto quiere decir que a futuro se pueden integrar sin problemas características como un área cliente y ocupar nuestros procesos internos.

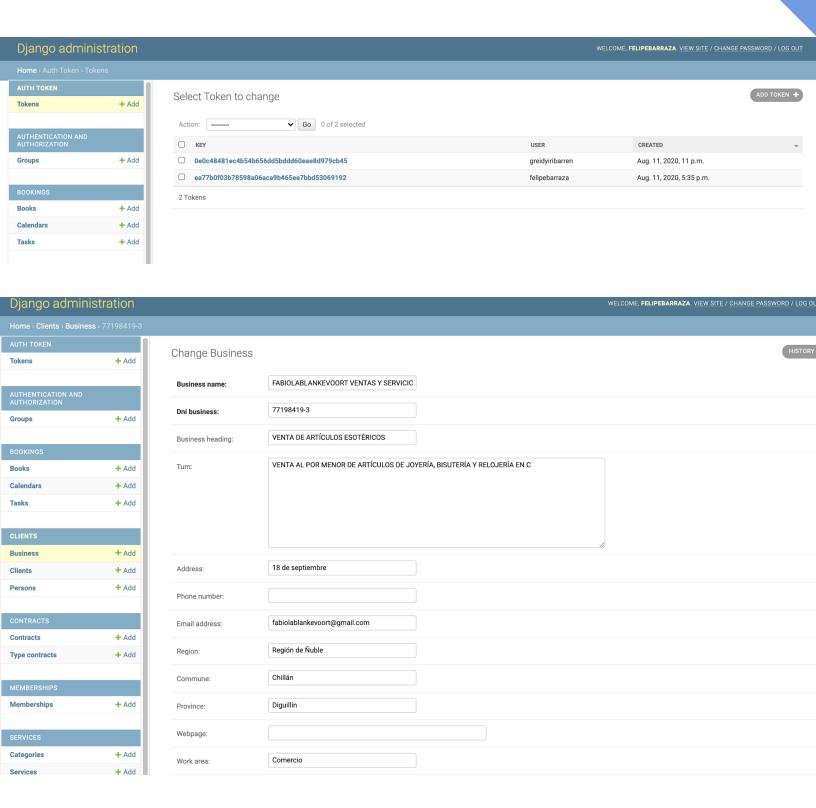
Se adjuntan capturas de ejemplo

#### Django administration

#### Site administration



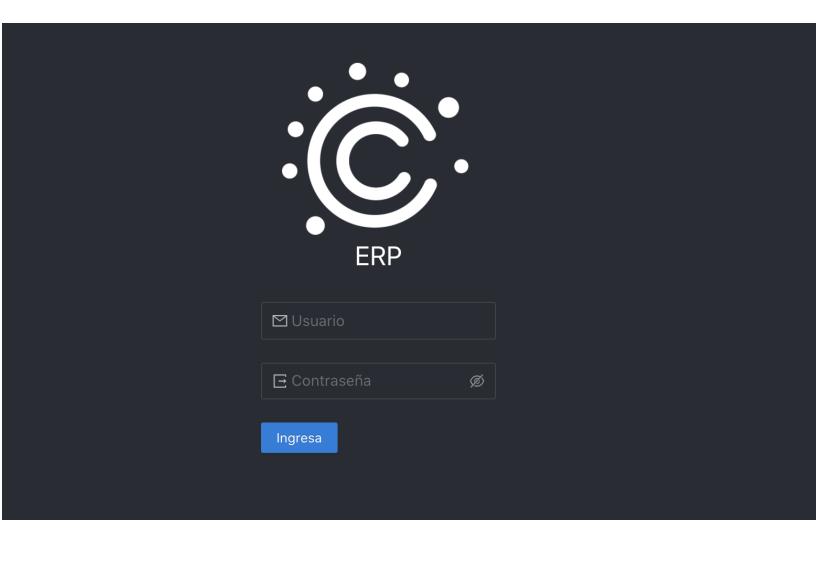
# My actions + asdfasfdsaf Service @ greidyiribarren Profile @ greidyiribarren Profile x greidyiribarren User



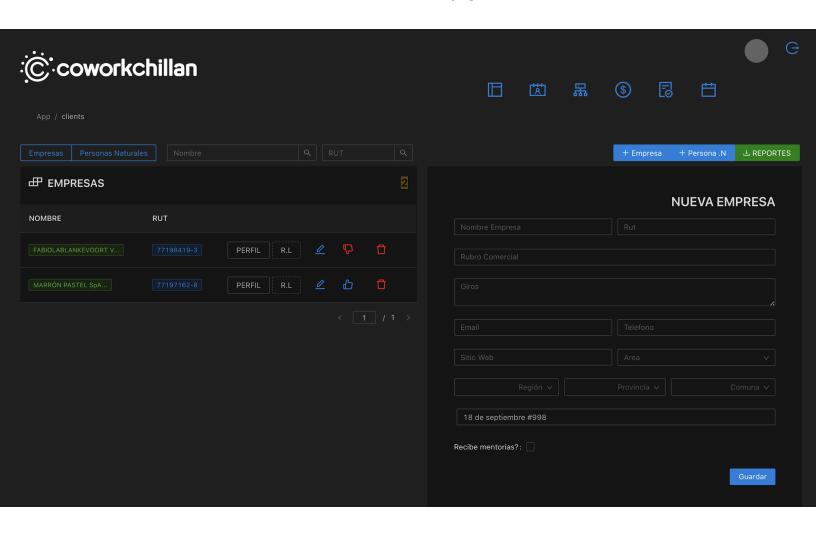
#### App Web:

Esta app o cliente se proveerá de nuestro servicio en django donde integraremos gráficamente cada modulo señalado:

Control de inicio de sesión;



Módulo de clientes/servicios/membresías/pagos/incidencias/calendario



#### Sección y área cliente para certificación B

