



Visionnaire validó el código existente y estructuró la evolución modular para completar el desarrollo de un CRM

A mediados de 2020, Britânia (que hoy también es propietaria de la marca Philco en Brasil) estaba desarrollando con otro proveedor un nuevo CRM (del inglés *Customer Relationship Management* - Gestión de Relaciones con Clientes) para centralizar la relación con clientes y *leads* (datos de contacto, historial de interacciones, oportunidades, propuestas, pedidos y soporte), pero el desarrollo fue interrumpido.

Como resultado, la empresa necesitaba una consultoría para evaluar lo que ya existía y confirmar la viabilidad de continuar el proyecto a partir del código entregado.

El desafío ocurrió durante la pandemia (y, por lo tanto, requería soluciones online) y durante la transición de Britânia hacia prácticas ágiles, lo que exigía equilibrar velocidad y gobernanza.

Además, era necesario comprender una herramienta de referencia, levantar requisitos sin “reinventar la rueda”, respetar los estándares de desarrollo de la empresa para facilitar mantenimiento y la futura internalización, y definir una arquitectura robusta y modular, pese a que el alcance seguía poco detallado y existía riesgo de estimaciones imprecisas.

Para abordar esto, Visionnaire inició con un diagnóstico del código fuente producido por el proveedor anterior, bajo cuatro perspectivas: Arquitectura, UI/UX, Frameworks (Node.js/React) y Gestión. El diagnóstico incluyó reuniones entre equipos, demostraciones del sistema en su estado actual, sesiones virtuales y presenciales, visitas a las fábricas en Joinville-SC y acompañamiento en sitio del uso de herramientas, culminando en presentaciones de conclusiones. Al final, se confirmó que era posible continuar el desarrollo a partir del código existente. Con la validación y el trabajo de requisitos/definición técnica, el proyecto avanzó a prototipado y desarrollo, con Visionnaire apoyando al equipo interno de Britânia hasta que el cliente asumió la continuidad. Como resultado, Britânia logró concluir el CRM que estaba detenido, ahorrando al reutilizar lo ya construido y con una base más segura para evolución futura. El caso también destaca ganancias de productividad y reducción de costos gracias a un equipo especializado y procesos ágiles y basados en la Nube, utilizando *Scrum* y *sprints* semanales. Tecnológicamente, el sistema utilizó JavaScript (Node.js) con NestJS, React y Redux; base de datos Microsoft SQL Server (con uso puntual de NoSQL); integración con TOTVS; y alojamiento en los servidores de Britânia.