

Travel Agency Inc. Business case

TL;DR;

Este caso de negocio estudia la incertidumbre en proyectos de ingeniería de software complejos con un volumen alto de ingenieros y equipos de ingeniería, toma de decisiones, objetivos, métricas y gestión de personas. Presentamos distintos aprendizajes y posibles soluciones.

Contexto

Travel Agency Inc. (TA) es una agencia de viajes que organiza viajes en todo el mundo. Múltiples destinos, paquetes turísticos, ticketing y hospitality. Sus servicios son variados pero todos ellos relacionados con el transporte de personas alrededor del mundo. Tiene sucursales en todos los continentes. La empresa nació hace más de 10 años en un entorno 100% digital desde sus inicios y ha tenido un rápido crecimiento. Su plantilla es de 1000 empleados, 600 personas en ingeniería y 400 personas en operaciones, marketing, HR y otros departamentos.

La empresa dispone de Apps nativas en iOS y Android y la versión en Web. Capta clientes B2C por las tres plataformas. Además, en web también gestiona clientes B2B.

El departamento de ingeniería trabaja en equipos (squads) y aplica metodologías ágiles para organizar su trabajo. Cada squad goza de suficiente autonomía para tomar sus decisiones dentro de su ámbito de actuación. Los squads están repartidos por todas las oficinas alrededor del mundo, es decir, trabajan en distintas zonas horarias. Algunos squads son sólo ingenieros de backend, otros de front-end y otros cross-functional. Cada equipo cuenta con 1 Engineering Manager y 1 Product Manager ambos liderando el equipo. Un equipo cross-functional tiene como ingenieros: 3 Backends, 2 Web engineers, 2 Android y 2 iOS engineers. Además el equipo cuenta con un soporte adicional de 1 Data Scientist a tiempo completo. A tiempo parcial colaboran 1 User Research y 1 Diseñador especializado en UI y usabilidad.

El departamento de ingeniería dispone de un career ladder habitual en el sector de las empresas tecnológicas de producto. Consiste en seis niveles, empezando en L1 como junior hasta L6 como principal engineer/director of engineering (alineada con Radford levels). El departamento organiza 2 ciclos de performance review, donde los ingenieros presentan sus desempeños/logros, los managers piden feedback y se entrega una valoración individual con la Performance Review. Esta valoración sigue una escala entre: needs improvement, sometimes meet expectations, meet expectations y outstanding. Los ingenieros que se clasifican como needs improvement deben participar en un Performance Improvement Plan (PiP) durante no más de 6 semanas. Al final de este período, si no hay cambios destacados se despiden a la persona por bajo rendimiento.

Ingeniería se divide en varias Business Units independientes, cada una de ellas con sus Director de ingeniería y Director de Producto, además pueden existir varios Senior EM y Senior PMs que agrupan equipos y dominios de negocio. Cada BU cuenta con varios equipos. Las BUs de producto son Growth, Cars, Flights, Train and Hotel. Y la BU de Platform donde se organizan los equipos que dan soporte a los equipos de producto. Platform opera la infraestructura tecnológica de bajo nivel para que los equipos de producto puedan centrarse en desarrollar nuevos negocios.

La BU de Flights es una de las más complejas. Gestionan billetes de avión, tanto el searching como el booking/check-in y se da soporte a los clientes ante eventuales circunstancias en los transfers entre vuelos. La BU cuenta con un total de 10 equipos. Es una de las más grandes del departamento de ingeniería.

Situación

El pipeline de la BU de flights consta de 5 etapas:

- 1) Acquisition: 3 squads, gestionando la API de la empresa para recibir precios de billetes de avión de partners y múltiples scrappers para capturar precios de otras aerolíneas.
- 2) Storage: 2 squads, para almacenar precios y depurar errores.
- 3) Integration: 2 squads, para integrar con el booking/api de la aerolínea (cuando hay acuerdo comercial).
- 4) Searching: 2 squads, para indexar los precios de forma que sea accesible por los clientes, aplicando lógicas complejas como por ejemplo para evitar conexiones imposibles entre vuelos.
- 5) Booking: 1 squad, integrar el booking de la aerolínea dentro del sistema de la agencia y gestionar el soporte al cliente durante el vuelo.

En la primera etapa (acquisition), se ha creado recientemente el tercer equipo (con todo ingenieros de backend) para mejorar la calidad de los precios ingestados. Dicha API se comparte entre los tres equipos. El leadership de la BU ha acordado medir el rendimiento del equipo con el siguiente KPI: número de precios ingestados en el sistema. Siguiendo la misma lógica que los otros dos equipos. El Goal asignado al equipo en base al KPI fue alcanzar 100M de precios ingestados en 6 meses. A más a más el leadership motiva al equipo para que eviten mapear la estructura de datos aerolínea por aerolínea manualmente, el equipo debe ser capaz de crear una solución escalable. Es decir, el coste de integrar una nueva aerolínea debe ser marginal. El producto se desarrolla desde cero y no hay deuda técnica.

El PM esboza el plan de trabajo, crea las primeras épicas y stories en JIRA y organiza los dos primeros sprints para empezar a trabajar.

El equipo se organiza en sprints de una semana, standup diaria excepto el lunes que hay sprint planning. El jueves backlog refinement para el siguiente sprint. El viernes sprint review y demo. Retrospectiva cada dos semanas.

Aparecen las dificultades

Después de los primeros 2 meses de trabajo se integra la primera aerolínea y los precios aparecen en el buscador de vuelos. Pero aparecen las primeras preocupaciones en los equipos de Storage, el sistema no podrá soportar la cantidad de precios nuevos ingestados previstos. El leadership decide no publicar los precios hasta que se estabilice el storage y pide al equipo que no publiquen precios en el sistema pero que sigan ingestando precios a la misma velocidad.

Después de los primeros 6 meses, el KPI alcanzado es muy bajo. No llega a 1M precios ingestados. Cada vez que se resuelve una dificultad se abren varias de nuevas que obligan a modificar el plan de trabajo. El equipo está desmoralizado. Se decide mantener el KPI pero en vez de ingestar precios de distintas aerolíneas se decide apostar por un agregador de precios que proporcionará 20 aerolíneas con una sola integración. Se reduce el scope del proyecto para focalizar la integración de una sola API y se abandona el proyecto inicial. Se modifica el equipo para añadir más ingenieros de backend.

Al año, se alcanzan a duras penas los 10M de precios ingestados en el sistema. El equipo tiene una rotación alta de ingenieros y el ambiente en el squad no es sano. El leadership ha perdido la confianza con el team management.

This work is licensed under a [Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License](https://creativecommons.org/licenses/by-nc-nd/4.0/).