

# Entrega 5: Deploy y consumo de la API

Esta entrega busca conectar la aplicación que desarrollaron en PHP con la API desarrollada por ustedes en la entrega anterior. Además deberán agregar ciertas funcionalidades a su aplicación. En concreto, deberán:

- Montar la API en un servidor de heroku.
- Consumir los JSON de la API en la aplicación web en PHP.
- Hacer una página de estadísticas y resumen general.

## Detalles Académicos

Para esta entrega cada encargado de grupo recibirá los datos de una base de datos MongoDB hosteada en un servidor del DCC.

## Introducción

Para finalizar la aplicación de la Dirección Secreta de Turismo, lo único que queda es juntar ambas componentes. Por un lado, tenemos la aplicación PHP, y por el otro el sistema de mensajería. La idea es poder tener un sistema completo que pueda ser utilizado por los usuarios.

## Deploy de la Web API

En esta entrega deben consumir la API desde tu aplicación en PHP, por lo que deberás subirla a un servidor para poder accederla desde cualquier computador. Para esto vas a usar **Heroku** y una base de datos MongoDB para que te entregaremos las credenciales. Esto significa que la API desarrollada en su entrega anterior debe ser **subida a producción**<sup>1</sup>.

---

<sup>1</sup>Te sugerimos revisar el siguiente tutorial <https://medium.com/the-andela-way/deploying-a-python-flask-app-to-heroku-41250bda27d0>. De todas formas, hay muchas formas de hacer esto y queda a tu criterio la manera de proceder.

## Consumir la API desde la aplicación PHP

Una vez subida la API a Heroku, debe ser posible consumirla desde la aplicación en PHP. **Una vez que un usuario inicia sesión en la aplicación en PHP**, debe ser capaz de:

- Revisar todos sus mensajes recibidos.
- Revisar todos sus mensajes enviados.
- Enviar un mensaje a un usuario en particular.
- Buscar mensajes por texto utilizando las funcionalidades de la búsqueda por texto de la entrega anterior.

Además el usuario debe ser capaz de ingresar un rango entre dos fechas. Para cada mensaje emitido por el usuario entre esas dos fechas, se debe visualizar en un mapa los lugares en que ha estado ese usuario (según la geolocalización de los mensajes). Para esto puede usar cualquier librería que permita manejar mapas en entornos web. Un ejemplo es **Leaflet**<sup>2</sup>.

**Importante:** Toda esta parte del trabajo se realiza en `bases.ing.puc.cl`. Las rutas implementadas en la aplicación de php quedan a criterio tuyo. Cada una de estas funcionalidades no deberían estar todas en una sola página, para que no quede demasiado colapsada. No es necesario implementar autenticación del lado de la API.

## Implementar un dashboard/página resumen

A los usuarios se les va a hacer muy útil una página donde puedan saber los hoteles más reservados, las regiones con más tour, el número de cepas de vino por región, etc. Para esto te pedimos que muestres:

- Un gráfico de barras que muestre el número de habitaciones reservadas por cada región.
- Un gráfico que muestre el número de cepas de vino por región.
- Al menos otras dos estadísticas que quedan a tu criterio. Pueden ser mostrados con gráficos u otra forma de resumir. La idea es que sirva para resumirle la información de la aplicación a un usuario que está navegando por la página.

Para esto puede usar cualquier librería que permita manejar gráficos en entornos web. Un ejemplo es **C3.js**<sup>3</sup>.

---

<sup>2</sup><http://leafletjs.com/>

<sup>3</sup><https://c3js.org/>

## Detalles adicionales

Para facilitar la corrección se solicita indicar a los ayudantes en un **readme** toda la información necesaria para comprender de manera más rápida cualquier aspecto de la entrega. Es **obligación** que el **readme** contenga la información para correr la aplicación y probar cada una de las funcionalidades solicitadas. Los ayudantes se reservan el derecho a descontar décimas en una entrega en la que se haya dificultado la corrección. **La entrega es para el jueves 27 de junio, a las 23:59**