

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.

Detalles Académicos

Para esta entrega cada encargado de grupo deberá hacer uso de la base de datos MongoDB que ya usaron en la entrega anterior.

Introducción

Llegó el final de proyecto y es el momento de atacar a los piratas de la pesca con datos y análisis. Por un lado, tenemos la aplicación PHP que maneja puertos y barcos, y por el otro el sistema de análisis de los mensajes. La idea es poder tener un sistema completo que pueda ser utilizado por los usuarios y además por los investigadores de la PDI detrás del caso.

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**¹.

¹Te 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

Funcionalidades básicas

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.

Funcionalidades para la PDI

LA PDI necesita, además, lo siguiente. Dado un id de usuario, un rango entre dos fechas y una o más palabras clave, se debe visualizar un mapa con:

- Los lugares en los que ese usuario ha enviado o recibido mensajes (según la geolocalización de los mensajes, campos `lat` y `long`).
- Si ese usuario es un jefe de instalación, el puerto donde trabaja.
- Si ese usuario es un capitán, los puertos en los que ha estado el barco donde capitanea.

Para el mapa puedes usar cualquier librería que permita manejar mapas en entornos web. Un ejemplo es `Leaflet`².

Importante: Toda esta parte del trabajo se realiza en `codd.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.

Instrucciones de Entrega

- La entrega es para el **jueves 10 de Diciembre, a las 20:00**.
- Los archivos de su aplicación deben ser subidos al servidor `codd.ing.puc.cl`, específicamente en la carpeta **Entrega5** del grupo con el menor índice.

²<http://leafletjs.com/>

- La nueva versión de la página debe estar ubicada en el directorio `/home/grupoXX/Sites/`. Esto significa que el homepage debe ser `/home/grupoXX/Sites/index.php`. Usar cualquier otra ruta arriesga que el proyecto no sea corregido.
- Para facilitar la corrección **deben** 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. Los ayudantes se reservan el derecho a descontar décimas en una entrega en la que se haya dificultado la corrección.