

Documento de Diseño

TriBot

**Joshua Campos Chiny
Erick Cobo Enríquez**

ULACIT

Componentes

Cliente

El cliente está conformado por tres páginas web, el signup, el login y routes. Estas páginas fueron desarrolladas en HTML5, CSS3, y jQuery para realizar las solicitudes web.

El cliente es independiente del servidor, por lo que se usa solicitudes cross-domain para comunicarse con él.

Para la autenticación se utiliza el usuario y la contraseña únicamente en la página de login; posteriormente, el cliente recibe un token, el cual guarda en el Session Storage. Cada solicitud que realice, enviará dentro de la autenticación este token.

Servidor

El servidor está conformado por un solo script desarrollado en Python, utilizando el micro framework de Flask. Se implementó CORS, para manejar las solicitudes cross-domain, y HTTPBasicAuth con tokens para implementar una capa básica de seguridad.

Se utilizó la librería de NetworkX para crear los grafos, los cuales son recorridos utilizando el algoritmo de Dijkstra.

El servidor atiende solicitudes para crear usuarios nuevos, para iniciar una sesión, para la cual crea y envía un token de autenticación, así como verificar las mejores rutas, tomando en cuenta el grafo creado anteriormente.

Base de Datos

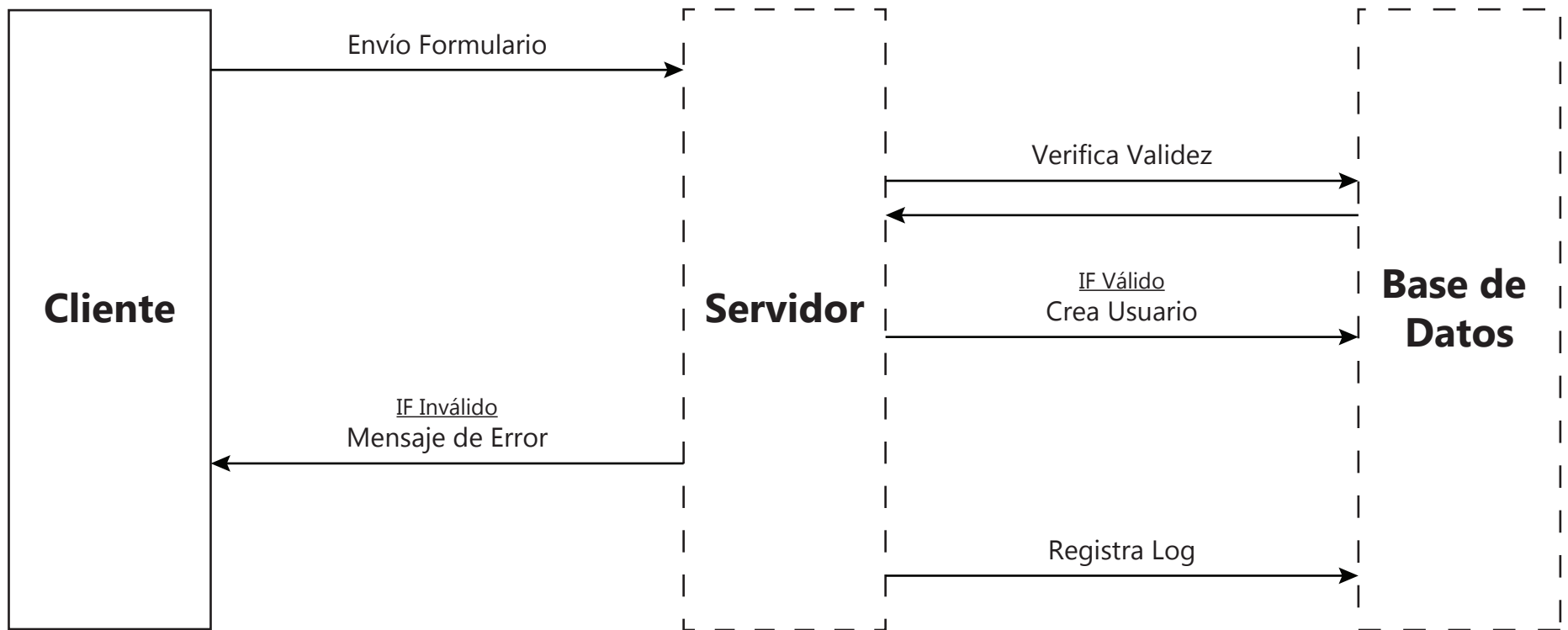
La base de datos se implementó utilizando MongoDB, la cual se ubica de manera local en el servidor. Dentro de ella, se crearon dos colecciones, una para los usuarios y otra para los logs de operaciones.

Los usuarios se manejan por medio de una identificación única que genera la base de datos, y posteriormente, se guarda el nombre, apellido, correo electrónico, nombre de usuario y contraseña. Los logs de operaciones poseen una identificación única creada también por la base de datos, así como el usuario, la fecha y la acción realizada.

Sign Up

Para tener acceso a la aplicación, es necesario haberse registrado previamente. La página de Signup existe para realizar esta acción, en la cual se puede observar un formulario que pide el nombre, apellido, correo, usuario y contraseña. En esta página todos los campos son obligatorios, de lo contrario mostrará un error. Una vez que se llenan estos campos, se hace clic en el botón de Submit para enviar dicho formulario al servidor.

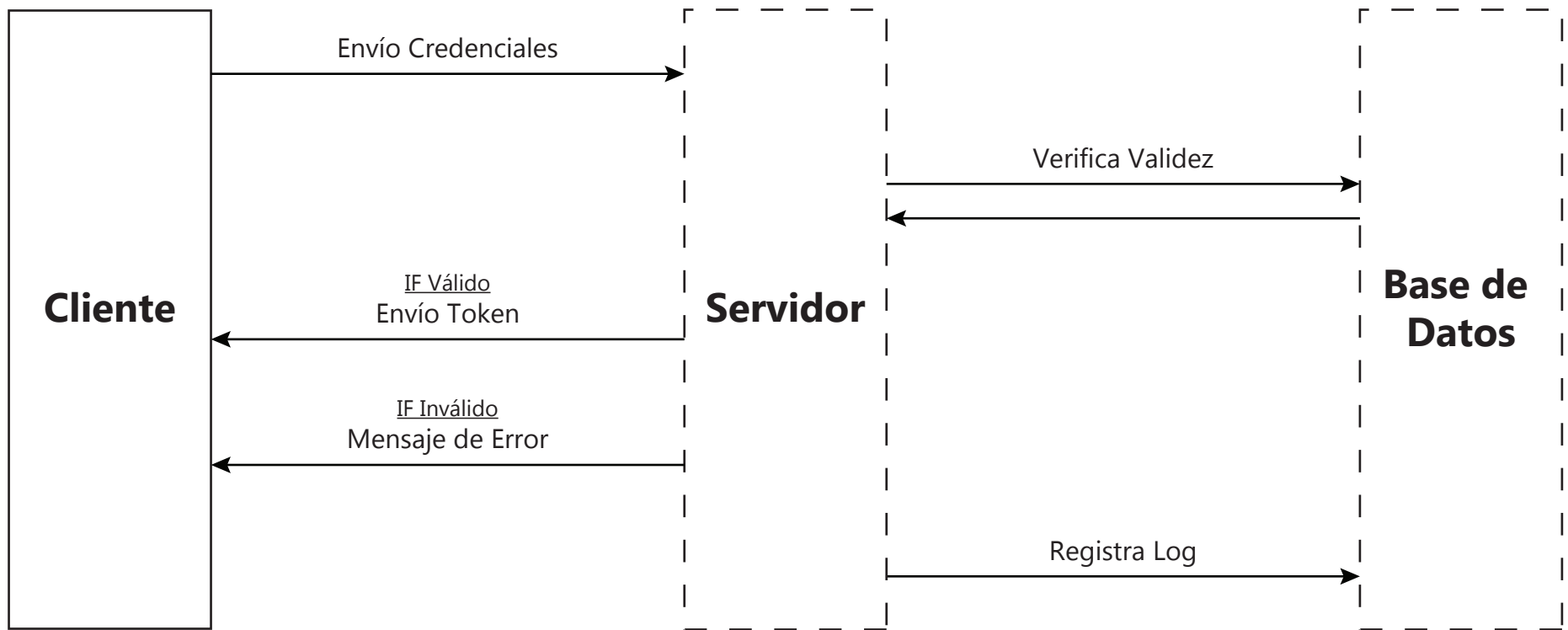
Realizado esto, el servidor recibe el formulario y verifica, mediante una consulta a la base de datos, que el usuario no exista. Si el usuario no existe, el servidor le envía la información correspondiente a la base de datos para realizar el registro; si ya existe, el servidor le envía un mensaje de error al cliente. Al final, se registra un log de lo sucedido.



Log In

Para tener acceso a la aplicación, es necesario iniciar una sesión. La página de Login existe para realizar esta acción, en la cual se puede observar dos campos, uno para el nombre de usuario y otro para la contraseña. En esta página todos los campos son obligatorios, de lo contrario mostrará un error. Una vez que se llenan estos campos, se hace clic en el botón de Login para enviar los credenciales al servidor.

Realizado esto, el servidor recibe los credenciales y verifica, mediante una consulta a la base de datos, que el usuario exista y que los datos sean válidos. Si el usuario existe, el servidor le envía un token de autenticación al cliente, para que lo utilice en las siguientes consultas; si el usuario no existe, el servidor le envía un mensaje de error al cliente. Al final se registra un log con lo sucedido.



Obtener Rutas

El objetivo principal de esta aplicación es poder elegir dos ubicaciones del mapa y que la aplicación le diga cuál es la ruta más corta entre esos dos puntos. Para esta función, utilizamos la página de Rutas en la que necesario elegir cada uno de los parámetros disponibles: Origen, Destino y Prioridad. Luego se hace clic en el botón de Get Route, en donde se hará la petición al servidor, y se muestra la respuesta en la página. Posteriormente, si el usuario lo desea, puede hacer clic en Get Alternatives para hacer otra solicitud y obtener otras cuatro alternativas. Al final de cada solicitud se registra un log en la base de datos de lo sucedido.

