

# Idea de proyecto

## Servicios

- Webservice Eventos  
Un servicio donde se pueden agregar eventos, borrarlos, editarlos y consultarlos
- Webservice Notas  
Un servicio donde se pueden agregar notas a los eventos, borrarlas, editarlas y consultarlas
- \* Webservice Autenticación  
Un servicio que tendrá las cuentas de usuario que podrán ser usadas en la agenda.
- Webservice Asistente  
Un servicio al que se le mandan comandos (texto) y retorna la intención del usuario (Para realizar acciones dentro del cliente web)
- Cliente Web  
El sistema con el cual el usuario final interactúa y se encarga de consumir los servicios anteriores

Cada uno de los servicios se ejecutaría en un contenedor diferente.

**costos:**

un servidor vps cuesta 20 pesos mensuales

un dominio ronda entre 50 y 300 pesos

en tiempo nos costara aproximadamente 40 horas

	CPU	RAM	SSD	Precio
<b>VPS S</b>	1 vCore	512 MB	10 GB	<b>\$20</b> /mes Solo <a href="#">Configurar</a>
<b>VPS M</b>	2 vCores	2 GB	80 GB	<b>\$40</b> /mes 8 meses después \$100/mes <a href="#">Configurar</a>
<b>VPS L</b>	2 vCores	4 GB	120 GB	<b>\$100</b> /mes 6 meses después \$200/mes <a href="#">Configurar</a>

**norahub.mx**

2 años por 1

\$700/año

**Gratis**

por 1 año

[Seleccionar](#)

**norahub.com.mx**

2 años por 1

\$500/año

**Gratis**

por 1 año

[Seleccionar](#)

**norahub.co**

6750/año

**\$130/año**

por 1 año

[Seleccionar](#)

[Mostrar más dominios](#)

**norahubmexico.mx**

2 años por 1

\$700/año

**Gratis**

por 1 año

[Seleccionar](#)

**norahubmexico.com**

2 años por 1

\$500/año

**Gratis**

por 1 año

[Seleccionar](#)

**norahubmexico.com.mx**

2 años por 1

\$500/año

**Gratis**

por 1 año

[Seleccionar](#)

[Mostrar más dominios](#)

# Descripción de los servicios

## Servicios

- Webservice Eventos  
Un servicio donde se pueden agregar eventos, borrarlos, editarlos y consultarlos
- Webservice Notas  
Un servicio donde se pueden agregar notas a los eventos, borrarlas, editarlas y consultarlas
- \* Webservice Autenticación  
Un servicio que tendrá las cuentas de usuario que podrán ser usadas en la agenda.
- Webservice Asistente  
Un servicio al que se le mandan mensajes en lenguaje natural (texto), solicitando una acción y retorna la intención del usuario (Para realizar acciones dentro del cliente web)  
Las intenciones posibles que puede retornar son:
  - add-event
  - add-event-today
  - add-event-tomorrow
  - get-event-today
  - get-event-tomorrow
  - get-event-todos
  - get-event-fecha
  - get-event-day
  - delete-event
  - message (Esta es la intención que se retorna cuando ninguna de las anteriores fueron identificadas)
- Cliente Web  
El sistema con el cual el usuario final interactúa y se encarga de consumir los servicios anteriores

# Instrucciones de instalación

## Requerimientos

- Sistema Operativo Windows 10+, Mac OS 10.0.1 +, linux
- Docker y docker-compose (version 3.3)
- Almacenamiento recomiendo 10GB

En la raíz del repositorio se encuentran ya dos scripts de instalación (Install.sh para sistemas Linux y Mac, y wininstall.bat para sistemas windows). Este script solicita todas las variables de configuración.

NOTA: Para ejecutar el script wininstall.bat es necesario hacerlo desde powershell.

## Instrucciones de uso

Para levantar el servicio se requiere ejecutar el comando docker-compose up en la raíz del proyecto.

## Para el uso del cliente web

En caso de no usar los puertos predeterminados de los servicios de usuarios y/o eventos, será necesario modificar manualmente las consultas desde los archivos .js. Esto debido a que el cliente es representativo de lo que se puede llegar a hacer.

Los archivos en cuestión son:

Noral-lends/cliente/app/src/js/login.js

```
$( "#login-form" ).submit( async ( event ) => {  
    event.preventDefault();  
    ocultarAlerta();  
    let url = `${host}:8080/login`; // Aquí se cambia por el puerto seleccionado en usuarios-api  
    console.log(url);  
    let usernameInput = $( "#username-input" ).val();  
    let passwordInput = $( "#password-input" ).val();  
    try {  
        let { data: usuario } = await axios.post(url, {  
            username: usernameInput,  
            password: passwordInput,  
        });  
  
        location.href = `/functions/save_session.php?email=${usuario.email}&id=${usuario.id}&nombre=${usuario.nombre}&token=${usuario.token}&username=${usuario.username}`;  
    } catch ( error ) {  
        console.log(error);  
        if (!error.response) {  
            // alert("algo salió mal, reintente de nuevo más tarde")  
            mostrarAlerta("algo salió mal, reintente de nuevo más tarde");  
            return;  
        }  
        if (!error.response.data) {  
            mostrarAlerta(error.message);  
        }  
        if (error.response.data.code !== 0) {  
            mostrarAlerta(error.response.data.error);  
        }  
    }  
});
```

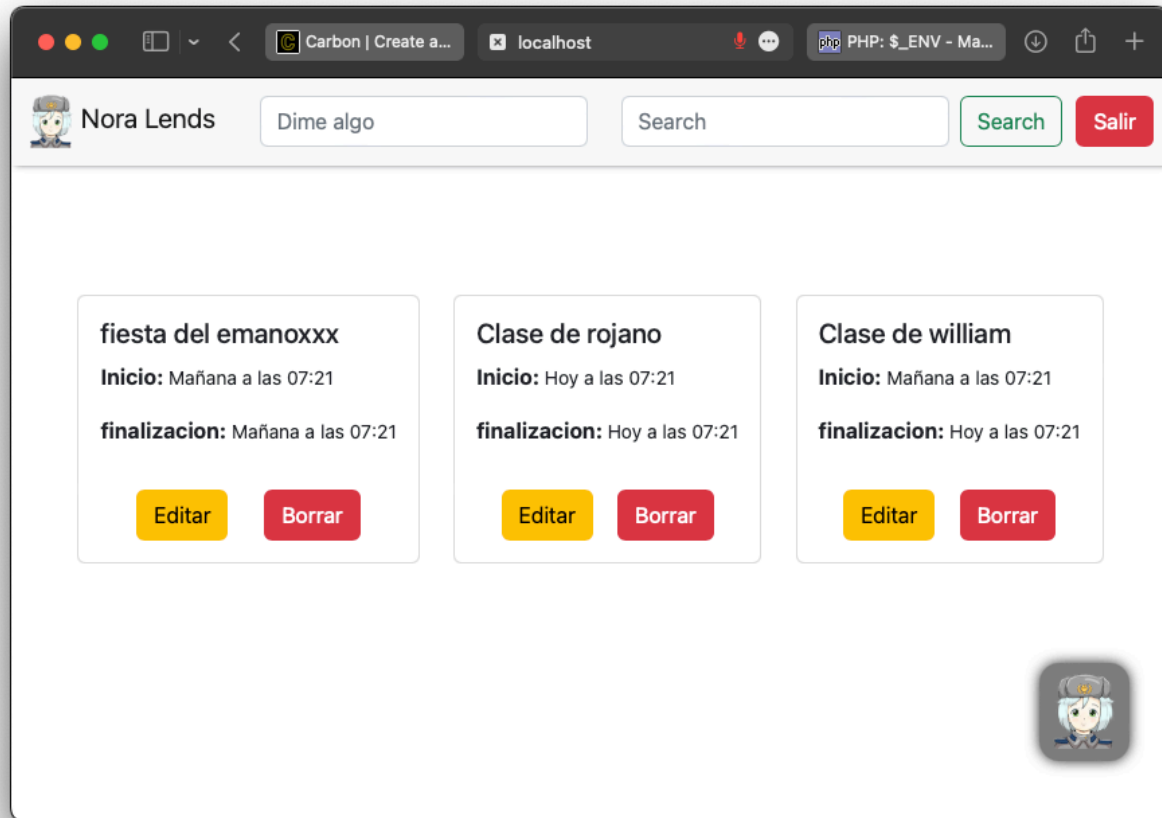
Noral-lends/cliente/app/src/js/main.js

```
1 const host = `http://${window.location.host}`;  
2 var eventos = [];  
3 var eventosFiltrados = eventos;  
4  
5  
6 if (token !== undefined) {  
7   let url = `${host}:8081/eventos`; // Aquí se cambia por el puerto seleccionado de eventos-api  
8   axios  
9     .get(url, {  
10      headers: {  
11        Authorization: `Bearer ${token}`,  
12        "Access-Control-Allow-Origin": "*",  
13      },  
14    })  
15    .then((response) => {  
16      eventos = response.data;  
17      eventosFiltrados = eventos;  
18      // eventos.forEach((evento) => {  
19      //   insertaEvento(evento);  
20      // });  
21      loadEventos();  
22    })  
23    .catch((error) => {  
24      alert(error);  
25    });  
26 }  
27  
28
```

## ¿Cómo usar el cliente?

Cuando se entra al cliente será necesario iniciar sesión con una cuenta (creada con alguna solicitud al servicio de usuarios, véase documentación para el consumo de la api )

Una vez que se inicia sesión habrá una pantalla como la siguiente:



Donde se mostrarán nuestros eventos, se puede cerrar sesión, buscar algún evento en el campo de Search, o pedirle al asistente que realice una acción.

---

## Utilizando el asistente

Hay dos formas de usarlo, ya sea escribiendo en el campo ("Dime algo") y presionando ENTER par realizar la acción, o con comandos de voz.

Para los comandos de voz solo hay que pedir la acción como a continuación:

**Nora** muestra mis eventos de hoy

(Cada acción solicitada debe empezar con la palabra Nora ya que es como empezará a escuchar)

# **Documentación para el consumo de la api**

<https://documenter.getpostman.com/view/18410004/UyxohiZm>