



EDUCACIÓN
SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO
NACIONAL DE MÉXICO

INSTITUTO TECNOLÓGICO DE OAXACA

INTEGRACION DE PROCESOS DE DESARROLLO DE SOFTWARE

GRUPO: 9SA

HORARIO: 07:00-08:00

DOCENTE: Espinosa Pérez Jacob

ALUMNOS:

Méndez Mendoza Luisa Michel

Martínez García Yahir Omar

Pérez de Jesús Edith

López García Lourdes Gloria

PRESENTA:

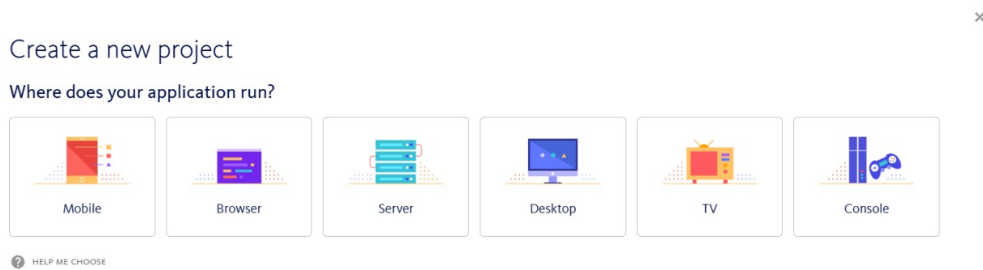
ANALISIS DE LOGS

CREACIÓN DEL PROYECTO EN BUGSNAG

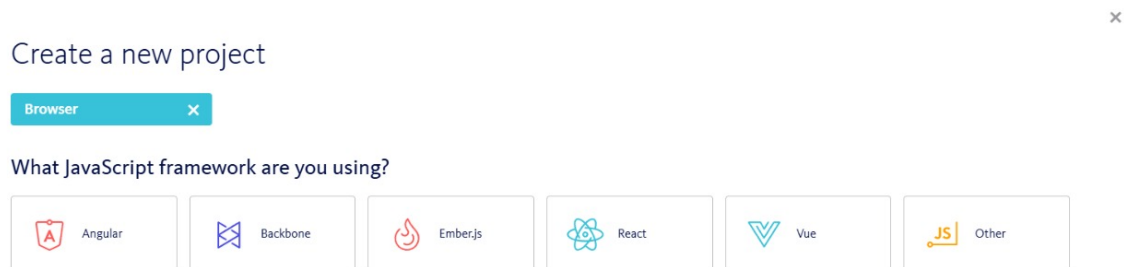
El proceso inicia con el inicio de sesión en [Bugsnag.com](https://bugsnag.com)

Se debe crear o usar un proyecto existente para el *frontend*.

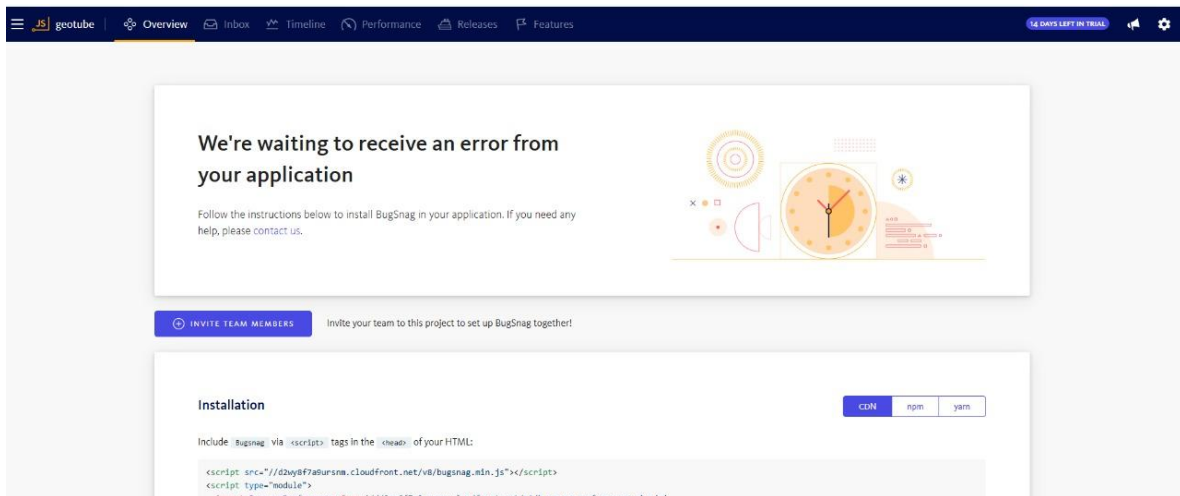
Paso 1: Se selecciona la opción "Browser" para ejecutar la aplicación



Paso 2: Seleccionamos la opción "React" y se crea un proyecto para el Frontend



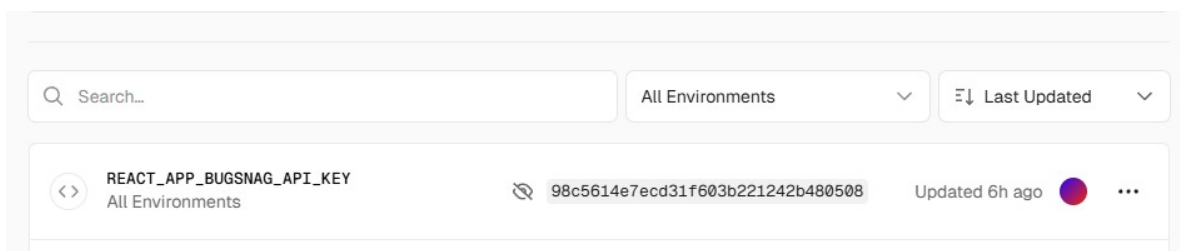
Obtención del API Key: Después de la creación, se obtiene la API Key de Bugsnag



CONFIGURACIÓN DE VARIABLES DE ENTORNO EN VERCEL

Se debe configurar la API Key en el entorno de despliegue, en este caso Vercel:

- Iniciamos sesión en Vercel.com.
- Navegamos al proyecto de *frontend* en Vercel.
- Accedemos a "Settings" > "Environment Variables".
- Añadimos la siguiente variable de entorno:
 - Name: REACT_APP_BUGSNAG_API_KEY
 - Value: 44b9f77d87102021c70d41b15a1b59a9
 - Se guarda y Vercel inicia un *redeploy*.
 - La API Key en Vercel es: 98c5614e7ecd31f603b221242b480508



MODIFICACIÓN DEL ARCHIVO PRINCIPAL (FRONTEND/SRC/INDEX.JS)

Se modifica el archivo de entrada principal para inicializar Bugsnag y envolver la aplicación con `ErrorBoundary`:

1. Importación de Bugsnag y plugins:

```
import Bugsnag from '@bugsnag/js';
import BugsnagPluginReact from '@bugsnag/plugin-react';
import BugsnagPerformance from '@bugsnag/browser-performance';
```

2. Inicialización de Bugsnag (Errores) : La `apiKey` se cargara desde la variable de entorno de Vercel/local (`process.env.REACT_APP_BUGSNAG_API_KEY`).

```
Bugsnag.start({
  apiKey: process.env.REACT_APP_BUGSNAG_API_KEY,
  plugins: [new BugsnagPluginReact(React)], // Pasa React al plugin
  appType: 'frontend', // Opcional, ayuda a clasificar en Bugsnag
  releaseStage: process.env.NODE_ENV === 'production' ? 'production' : 'development'
});
```

3. Inicialización de Bugsnag Performance (Opcional):

```
BugsnagPerformance.start({
  apiKey: process.env.REACT_APP_BUGSNAG_API_KEY, // Usar la misma API Key
  aunque se puede añadir más opciones si se necesita
});
```

4. Uso de `ErrorBoundary`:

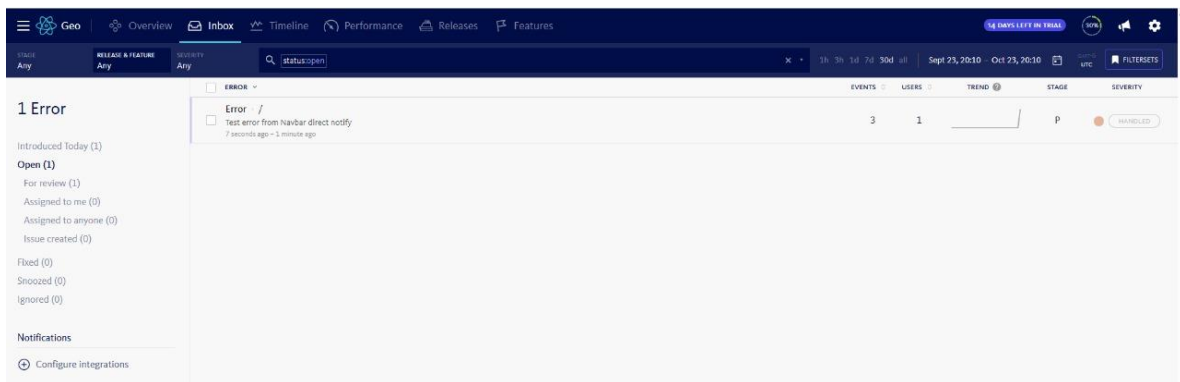
```
const ErrorBoundary = Bugsnag.getPlugin('react').createErrorBoundary(React);
// ...
root.render(
  <React.StrictMode>
    {/* 5. Envuelve tu componente principal (AppWrapper) con el ErrorBoundary */}
    <ErrorBoundary>
      <AppWrapper />
    </ErrorBoundary>
  </React.StrictMode>
);
```

MONITOREO DE ERRORES EN BUGSNAG

Una vez configurado y desplegado, Bugsnag comenzará a reportar los errores:

- Error Registrado: Se observa un error con la descripción "Test error from Navbar direct notify".

- Estado: El error está con status:open.
- Detalles del Error (Primer Reporte):
 - Eventos: 3
 - Usuarios: 1
 - Etapa (Stage): p (producción)
 - El error fue introducido "Today" y hay 1 abierto (Open (1))



Detalles del Error (Timeline):

- Eventos: 4
- Usuarios: 1
- App Types: 100.0% frontend
- Release Stages: 100.0% production
- El error ocurrió hace 5 horas en la vista de *Timeline*

