* 12 Estado Aplicación Vuex
  + Links;
    - https://carlosazaustre.es/como-funciona-flux
  + Proviene de Flux o Redux , tecnologia de acceso unico a la info
  + Estado de la aplicación intenta “centralizar” las peticiones de las variables publicas en un solo sitio
  + Se incorpora el concepto de store donde se aglutina todo el estado de la app
    - State: donde almacenamos los datos
    - Getters: funciones de acceso al dato
    - Mutations: funciones de actualizacion (sincronica) del dato
    - Actions: Invocaciones asincronicas de la actualizacion del dato
    - Modules: permiten “disgregar” el store en varios modules para darle modularidad
  + Al unificarlo en un solo sitio , se dota de mas seguidad y seguimiento
    - Validaciones antes de almacenar un valor
    - Registros de logs
    - Etc
  + El acceso a la variable debe hacerse mediante una funcion encapsulada ( getter)
  + La actualizacion (setter) debe hacerse mediante una mutacion
  + Esa actualizacion puede ser
    - Sincronica: mutations ( commit )
    - Asincronica: actions ( dispatch )
  + Consejos:
    - Poner las mutaciones en constantes y mayusculas , asi sabremos en un solo sitio aquellas mutaciones que se pueden hacer en nuestra app

Instalacion

Npx vue create 17\_estado\_app\_vuex

Npx vue add vuex

Inyecta en el main store

Crea la carpeta store con un index con un store vacio

Ejercicio 2

Creamos un vuex con http (axios)

Npx vue create 17\_vuex\_http

Npx vue add vuex

Npx vue add axios

Usamos la url de github

* + - <https://api.github.com/users/Unitech/repos>