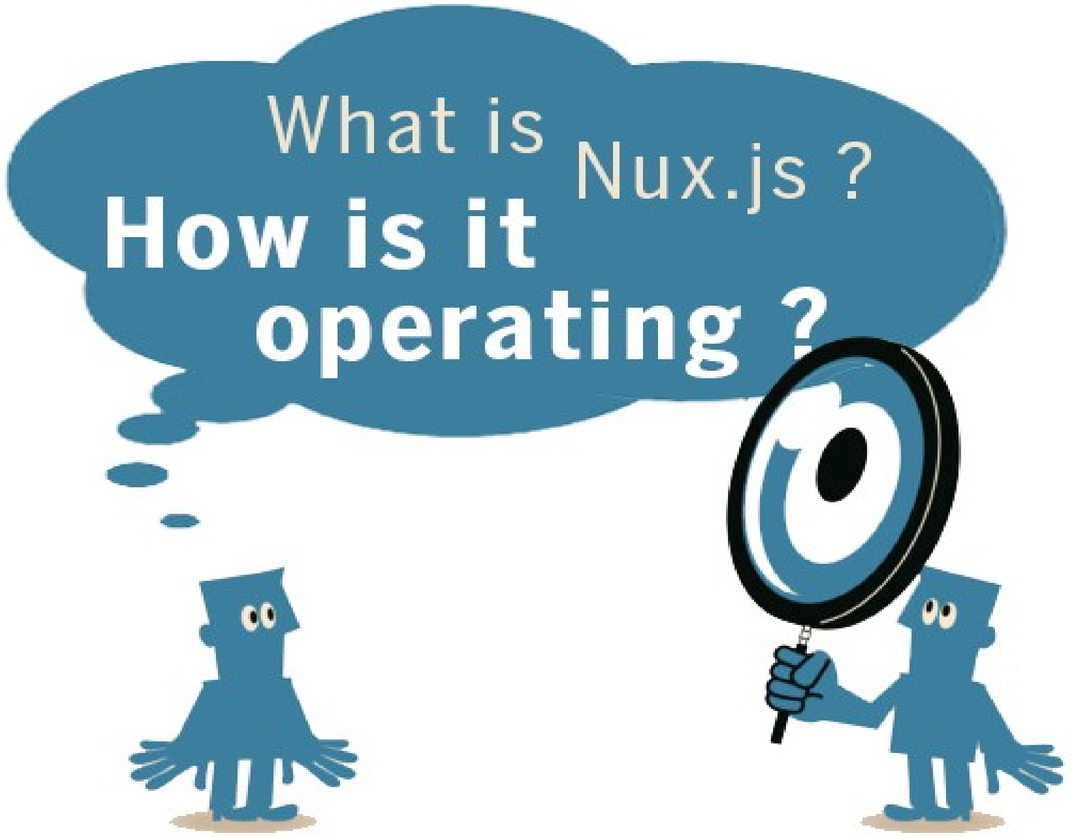
levi nine
Technology Services

LEVI9 ACADEMY

Nuxt.js lecture 4: Nuxt.js and its main features

University Team
Ukraine



NUXT

Nuxt.js is a framework built on top of Vue with the aim to make development easy and powerful.

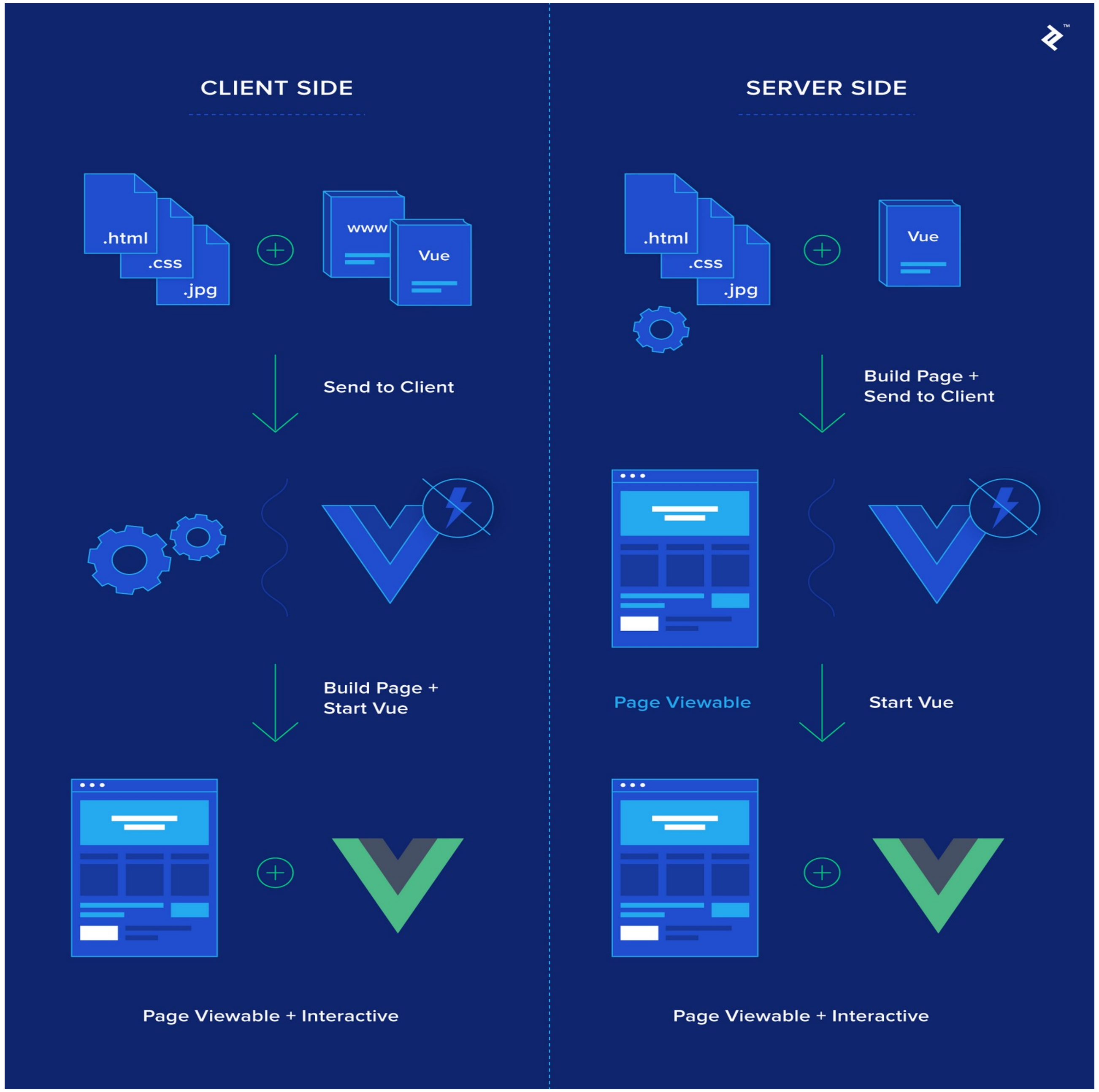
What are the benefits of Nuxt.js?

1. Works faster.

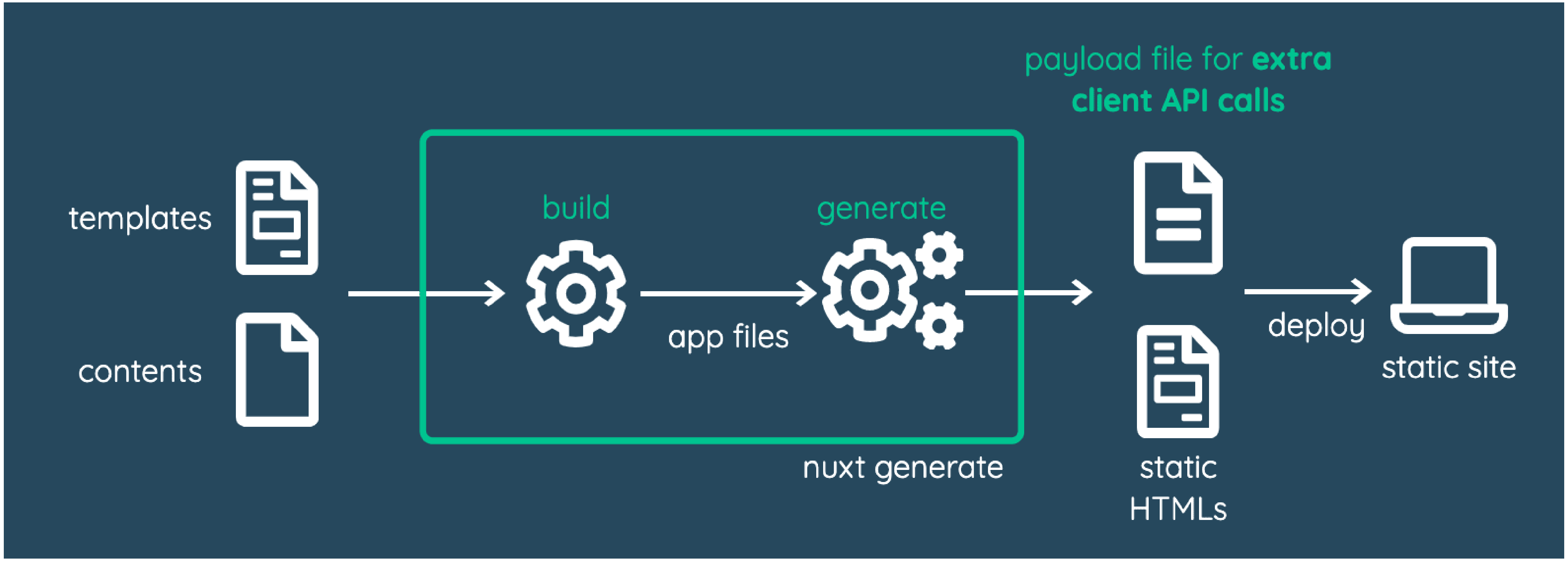
2. Create universal apps without the hassle.

A universal app is used to describe JavaScript code that can execute both on the client and the server side (SPA and SSR).

Link: https://nuxtjs.org/docs/concepts/server-side-rendering


CLIENT SIDE
SERVER SIDE
.html
.css
.jpg
www
Vue
Send to Client
Build Page +
Send to Client
Build Page +
Start Vue
Page Viewable
Start Vue
Page Viewable + Interactive
Page Viewable + Interactive

3. SSG (static site generation).

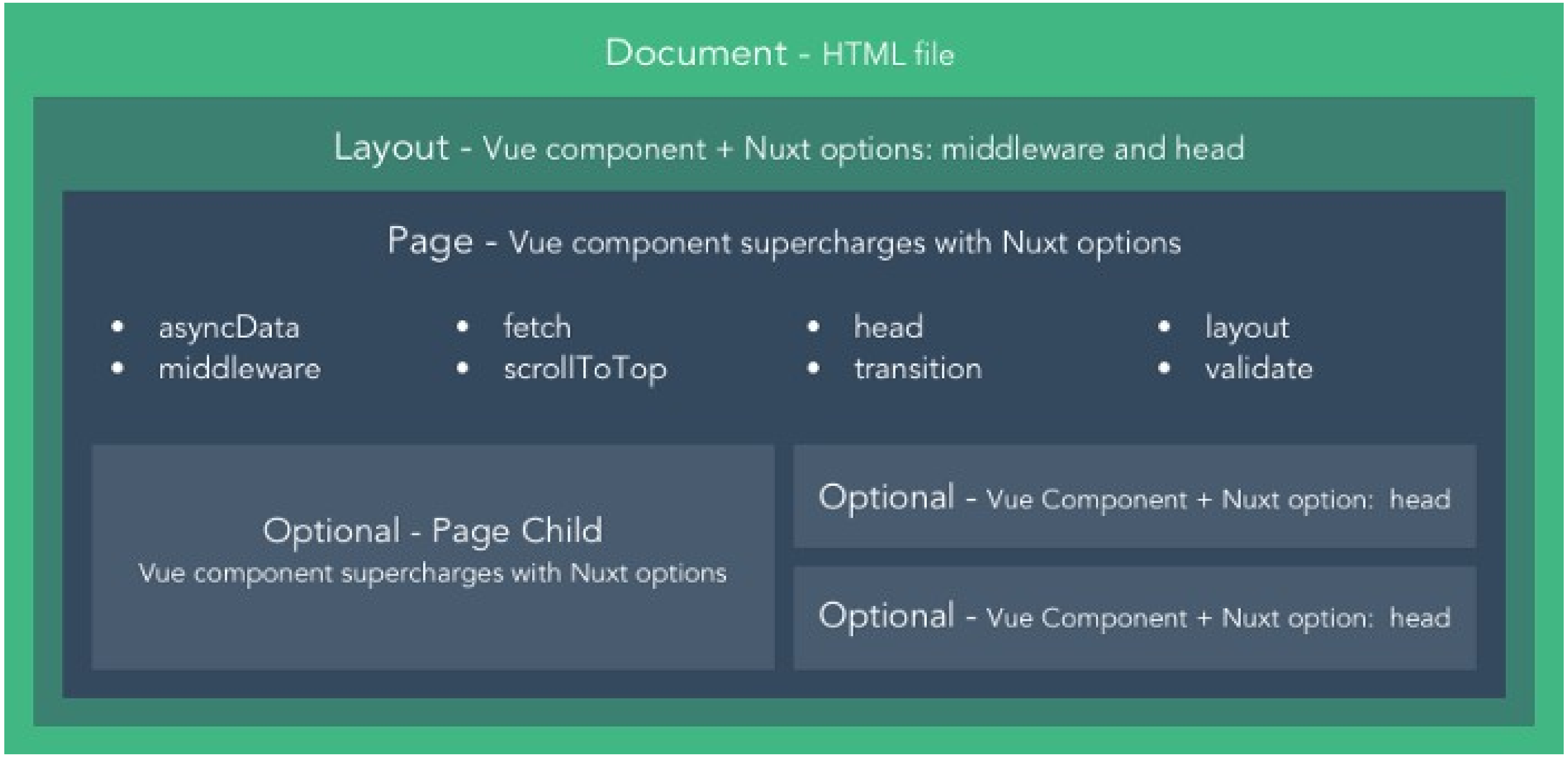


Link: https://nuxtjs.org/docs/concepts/static-site-generation

4. SEO support.

Link: https://nuxtjs.org/docs/features/meta-tags-seo

5. Get great project structure by default.


Document - HTML file
Layout - Vue component + Nuxt options: middleware and head
Page - Vue component supercharges with Nuxt options
• asyncData
• middleware
• fetch
• scrollToTop
• head
• transition
• layout
• validate
Optional - Page Child
Vue component supercharges with Nuxt options
Optional - Vue Component + Nuxt option: head
Optional - Vue Component + Nuxt option: head

Here are a few of the main directories that it sets you up with:

- **components** — a folder so you can organize your individual Vue components.
- **layouts** — a folder to contain your main application layouts.
- **pages** — a folder to contain your app's routes. Nuxt.js reads all the .vue files inside this directory and creates the application router.
- **store** — a folder to contain all of your app's Vuex Store Files.

Link: https://nuxtjs.org/docs/get-started/directory-structure/

Drawbacks of Nuxt.js:

1. NuxtJS has poor TypeScript support.
2. Getting Custom Libraries to Work with Nuxt with Can Be Challenging.
3. Debugging It Can Get Painful.

NUXTJS

Life Cycle

Link: https://nuxtjs.org/docs/concepts/nuxt-lifecycle

NUXTJS

1. Configuration

Link: https://nuxtjs.org/docs/features/configuration

2. The difference between assets and static folders

Assets directory

The **assets** directory contains your un-compiled assets such as Stylus or Sass files, images, or fonts.

Link: https://nuxtjs.org/docs/directory-structure/assets

Static directory

The **static** directory is directly mapped to the server root () and contains files that likely won't be changed. All included files will be automatically served by Nuxt and are accessible through your project root URL.

Link: https://nuxtjs.org/docs/directory-structure/static

3. File system routing

Nuxt automatically generates the **vue-router** configuration based on your file tree of Vue files inside the **pages** directory. When you create a .vue file in your pages directory you will have basic routing working with no extra configuration needed.

Links:
https://nuxtjs.org/docs/features/file-system-routing
https://nuxtjs.org/docs/features/nuxt-components

4. Layouts (+ error page)

Layouts are a great help when you want to change the look and feel of your Nuxt app. Whether you want to include a sidebar or have distinct layouts for mobile and desktop.

Link: https://nuxtjs.org/docs/directory-structure/layouts

5. Plugins

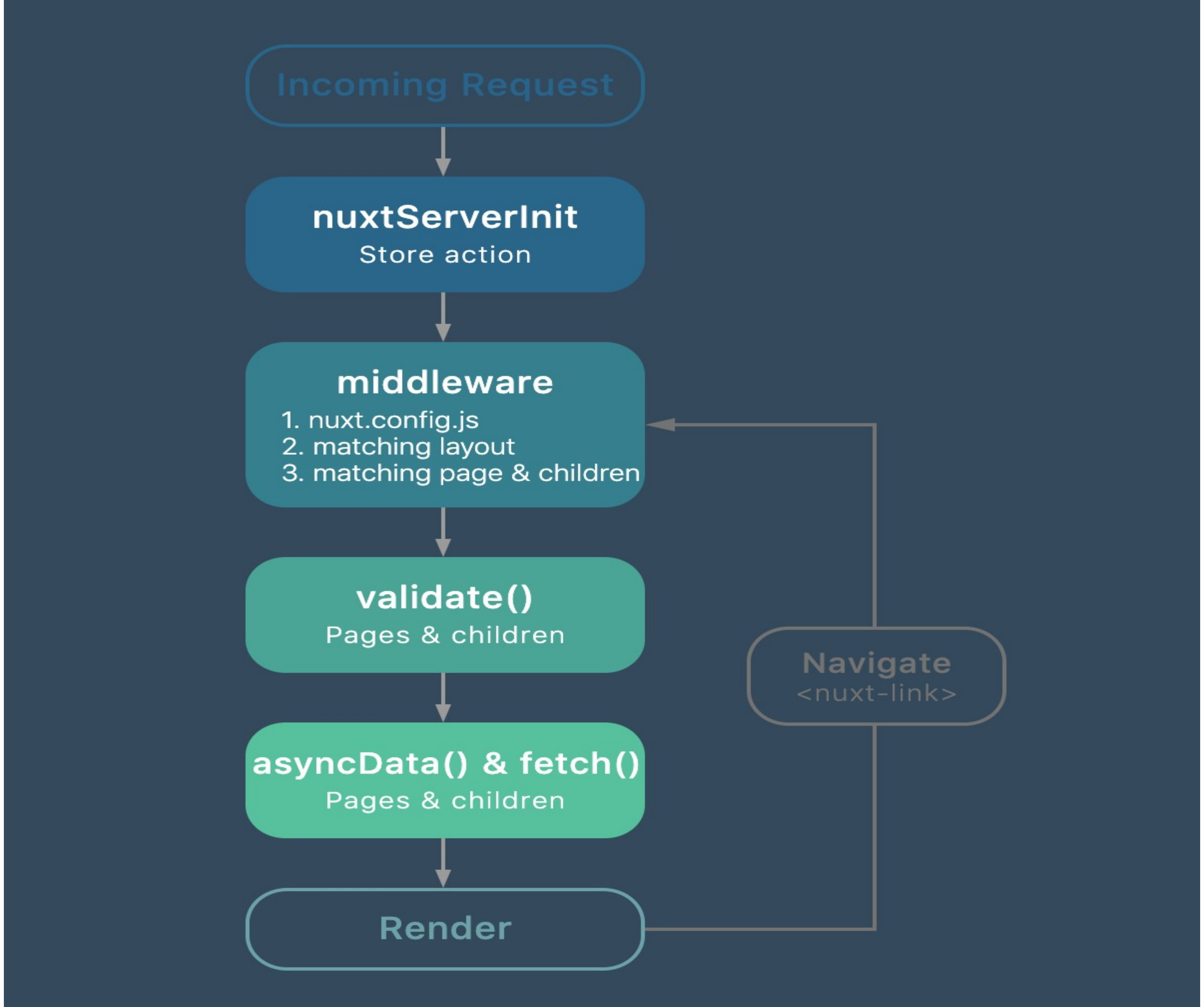
The plugins directory contains your Javascript plugins that you want to run before instantiating the root Vue.js Application.

Link: https://nuxtjs.org/docs/directory-structure/plugins

6. Store

The **store** directory contains your **Vuex Store** files. The Vuex Store comes with Nuxt out of the box but is disabled by default. Creating an **index.js** file in this directory enables the store.

Link: https://nuxtjs.org/docs/directory-structure/store


Incoming Request
nuxtServerInit
Store action
middleware
1. nuxt.config.js
2. matching layout
3. matching page & children
validate()
Pages & children
asyncData() & fetch()
Pages & children
Render
Navigate
<nuxt-link>

‘nuxtServerInit’

It is a reserved store action available only in **store/index.js** file and if defined will be called on server-side before rendering requested routes.

Link: https://nuxtjs.org/docs/directory-structure/store/

‘middleware’

The **middleware** directory contains your application middleware. Middleware lets you define custom functions that can be run before rendering either a page or a group of pages (layout).

Link: https://nuxtjs.org/docs/directory-structure/middleware

‘validate method’

Nuxt lets you define a validator method inside your dynamic route component.**Validate** is called every time before navigating to a new route. It will be called server-side once (on the first request to the Nuxt app) and client-side when navigating to further routes. This method takes the **context** object as an argument.

Link: https://nuxtjs.org/docs/components-glossary/validate/

‘Data fetching’

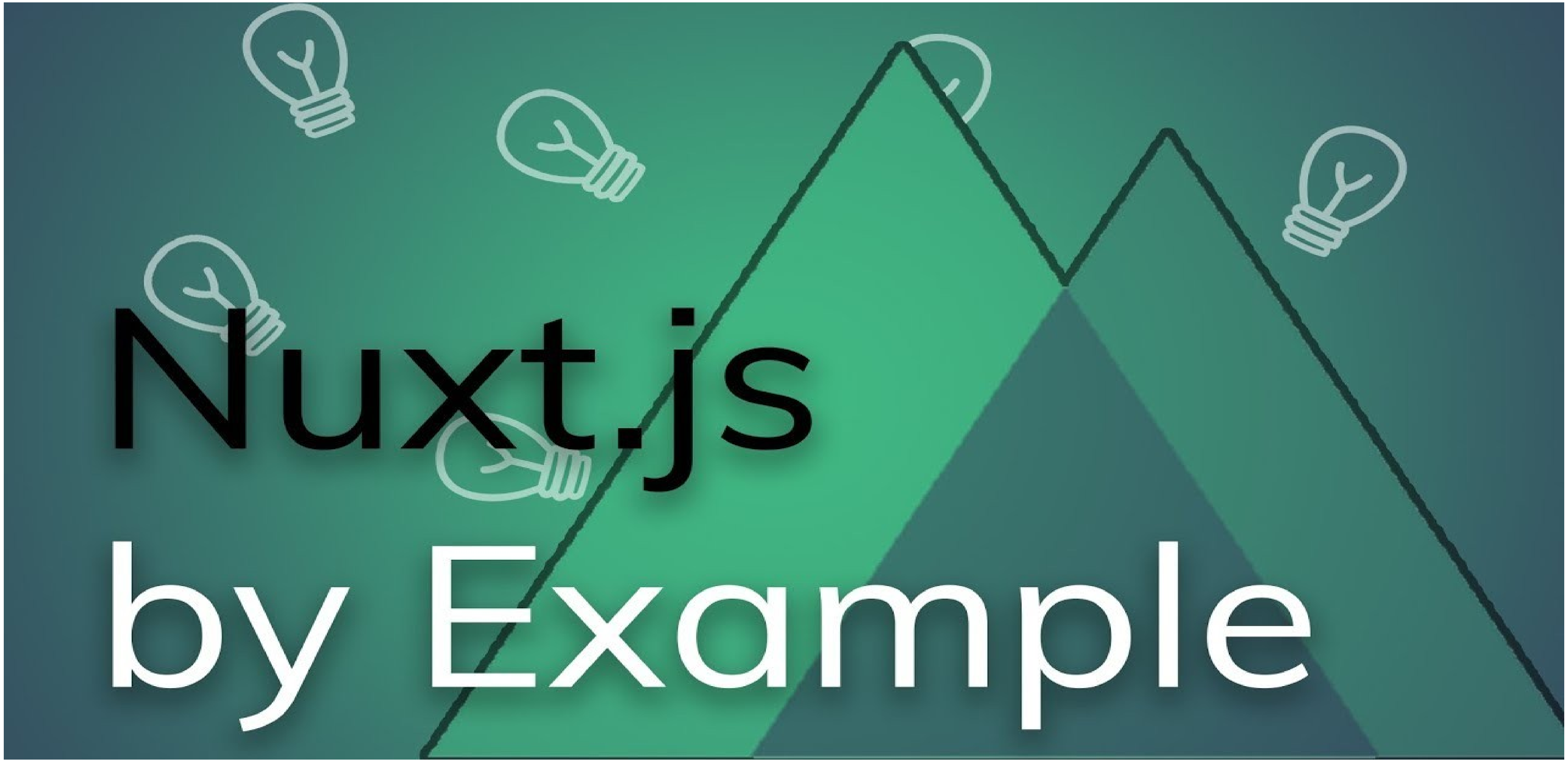
In Nuxt we have 2 ways of getting data from an API. We can use the **fetch** method or the **asyncData** method.

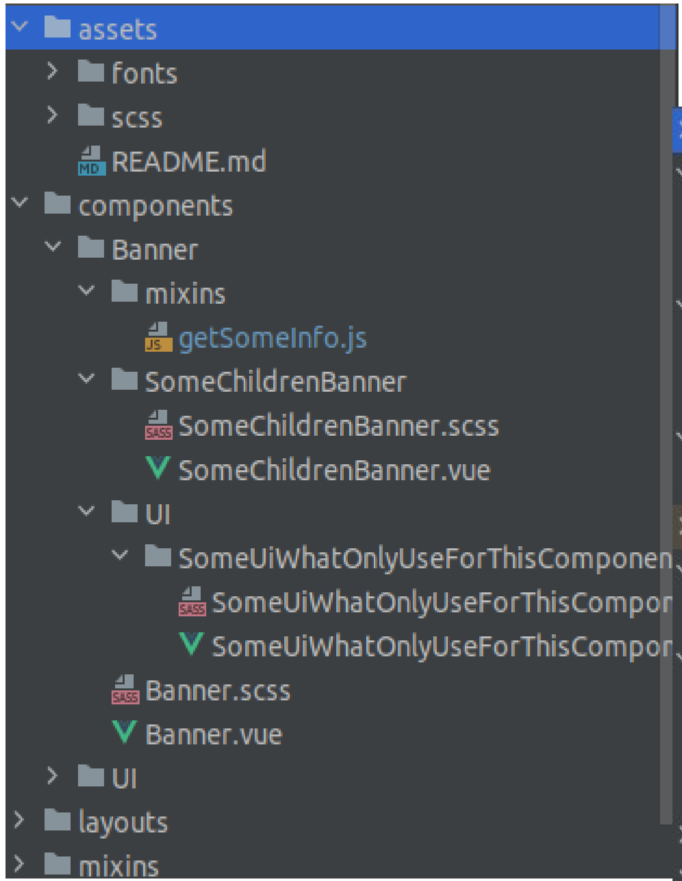
Nuxt has two hooks for asynchronous data loading:

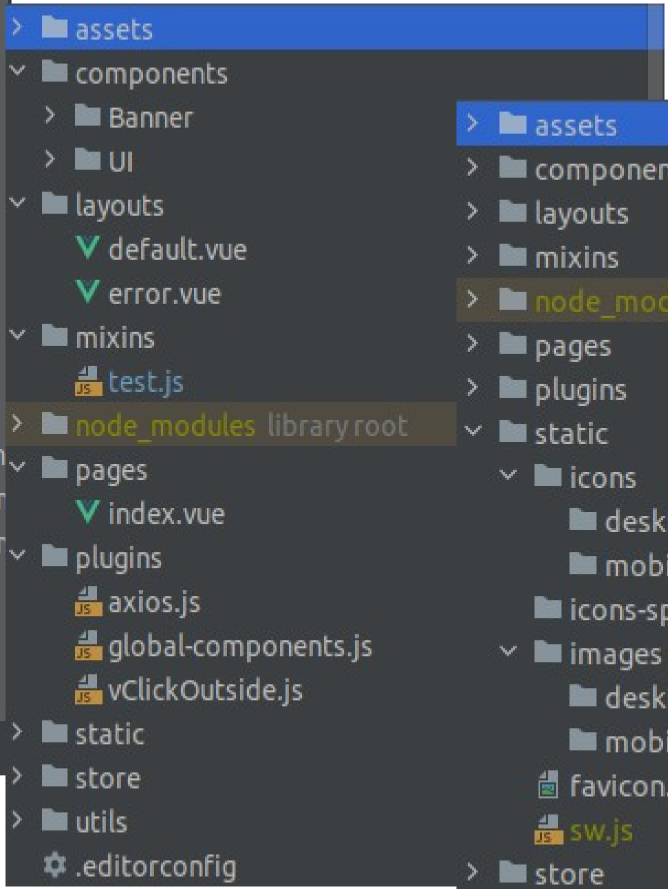
- **asyncData** hook is a way to fetch data server-side. It waits for its promise to be resolved before rendering the page and directly merges the return value to the local state.
- **fetch** is a hook called during server-side rendering after the component instance is created, and on the client when navigating. The fetch hook should return a promise (whether explicitly, or implicitly using **async/await**) that will be resolved:
  1. On the server, before the initial page is rendered
  2. On the client, some time after the component is mounted

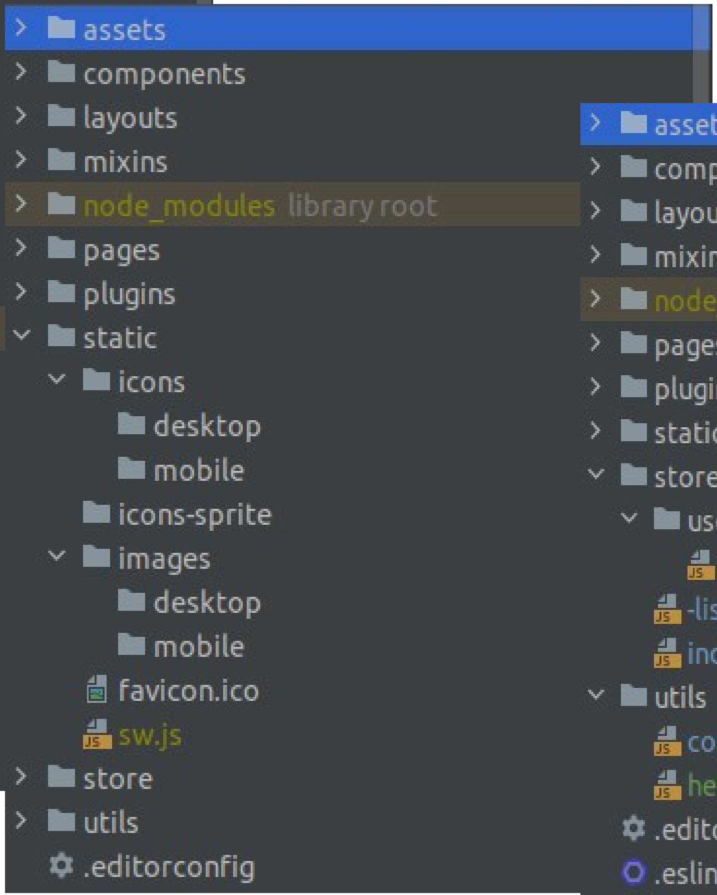
Link: https://nuxtjs.org/docs/features/data-fetching
Example: https://codesandbox.io/s/wqxe3?file=/pages/async-data/_id.vue

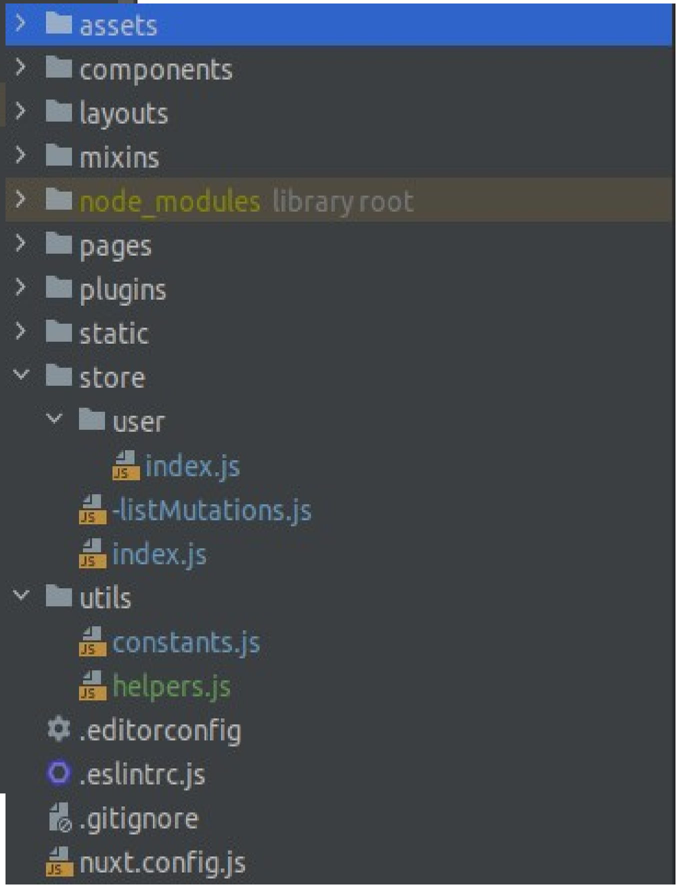
Correct structure of the Nuxt:




```
v assets
  > fonts
  > scss
    README.md
v components
  v Banner
    v mixins
        getSomeInfo.js
    v SomeChildrenBanner
        SomeChildrenBanner.scss
        SomeChildrenBanner.vue
    v UI
      v SomeUiWhatOnlyUseForThisComponen
          SomeUiWhatOnlyUseForThisCompor
          SomeUiWhatOnlyUseForThisCompor
      Banner.scss
      Banner.vue
  > UI
> layouts
> mixins
```


```
> assets
v components
  > Banner
  > UI
v layouts
    default.vue
    error.vue
v mixins
    test.js
> node_modules  library root
v pages
    index.vue
v plugins
    axios.js
    global-components.js
    vClickOutside.js
> static
> store
> utils
  .editorconfig
```


```
> assets
> components
> layouts
> mixins
> node_modules  library root
> pages
> plugins
v static
  v icons
      desktop
      mobile
    icons-sprite
  v images
      desktop
      mobile
    favicon.ico
    sw.js
> store
> utils
  .editorconfig
```


```
> assets
> components
> layouts
> mixins
> node_modules  library root
> pages
> plugins
> static
v store
  v user
      index.js
    -listMutations.js
    index.js
v utils
    constants.js
    helpers.js
  .editorconfig
  .eslintrc.js
  .gitignore
  nuxt.config.js
```

?

QUESTIONS

levi nine

Technology Services

USEFUL LINKS:

1. https://nuxtjs.org/docs/get-started/installation – official documentation
2. https://habr.com/ru/post/336902/ – habr article
3. https://www.youtube.com/watch?v=lm9olMCRCIc – Nuxt JS (fast course in 70 minutes)

THANK YOU

Levi9 Ukraine
Kiev / Lviv

levi nine
Technology Services