

# Pages and Routing

Web Development ប្រើប្រាស់ Nuxt 4 ឬ Vue.js, Routing ដើម្បីគ្រប់គ្រងការគ្រប់ URL ទៅកាន់ទំនាក់ណាមួយ ឬ Component ណាមួយ។ Nuxt បង្កើត route ស្ថិតិផ្តើម file មាននេះក្នុង `pages/`។

- នៅពេលអ្នកវិភាគ `fff-adm.com/about`, Routing នឹងទៅកម្រិត File `about.vue` ដើម្បីបង្ហាញឱ្យអ្នករើស។
- នៅពេលអ្នកវិភាគ `fff-adm.com/contact`, Routing នឹងទៅកម្រិត File `contact.vue`។

## 1. Static Routing

- Index Route: File `app/pages/index.vue` ជាកំណាលគ្រប់ទំនាក់នៅមិនមែន `(/)`។
- Static Routes: File `app/pages/about.vue` ជាកំណាលគ្រប់ផ្ទាល់ទៅកាន់ `/about`។
- Nested Routes: បើអ្នកបង្កើត Folder `app/pages/blog/index.vue` វាបានកំណាលគ្រប់ `/blog`។

## 2. Dynamic Routing - ការប្រើប្រាស់ Parameters ជាមួយកំណាលគ្រប់

ដើម្បីប្រាជែងកំណាល URL ត្រូវមាន Parameters ដូចជា ID ឬ ឈ្មោះ (ឧទាហរណ៍: `/posts/1` ឬ `/posts/abc`)។ យើងប្រើប្រាស់ Square Brackets `[]` ដើម្បីបង្កើតវិញ្ញាន dynamic route។

- File: `app/pages/users/[id].vue`
- URL: `/users/123`
- របៀបបង្កើតនៃយុទ្ធសាស្ត្រ URL: យើងប្រើ `useRoute()`។

ឧទាហរណ៍:

```
-| pages/  
---| users/  
-----| [id].vue
```

```
<script setup>  
const route = useRoute()  
const userId = route.params.id  
</script>  
  
<template>  
<div>  
  <h1>User Profile</h1>  
  <p>User ID គឺជាអ្នកគិត: {{ userId }}</p>  
</div>  
</template>
```

## 3. NuxtLink

ដើម្បីផ្តល់ទៅកាន់ទំនាក់រដ្ឋធម្មជាតាប់ Reload Page យើងប្រើ Component `<NuxtLink>` ដែលសម្រួល Tag `<a>`។

```
<template>  
<nav>  
  <NuxtLink to="/">ទៅកាន់ទំនាក់នៅមិនមែន</NuxtLink>  
  <NuxtLink to="/about">ទៅកាន់about</NuxtLink>  
  <NuxtLink :to="/users/${userId}">ទៅកាន់ Profile</NuxtLink>  
</nav>  
</template>
```

## 4. 404 Page

- File: `app/pages/[...slug].vue`

edited article by Mr. limvichet, contact 012856917/067878118.

- វានិងបង្ហាញកល់ពេលដែល User រាយ URL ផ្តល់។

```
<script setup>
const route = useRoute()
const pathArray = route.params.slug
</script>

<template>
<div class="container">
<h1>ទាំងអស់ Catch-all</h1>
<p>អ្នកកំណុងស្ថិតិថ្នាក់ខាងក្រោម: <strong>/{{ pathArray.join('') }}</strong></p>

<div class="debug">
<h3>ទីន៍ណែនាំ Slug Array:</h3>
<ul>
<li v-for="( folder, index ) in pathArray" :key="index">
    ជាទី {{ index + 1 }}: {{ folder }}
</li>
</ul>
</div>
<NuxtLink to="/" class="btn">ត្រឡប់ទៅទាំងអស់</NuxtLink>
</div>
</template>

<style scoped>
.container { padding: 20px; font-family: sans-serif; }
.debug { background: #f4f4f4; padding: 10px; margin-top: 20px; }
.btn { color: blue; text-decoration: underline; }
</style>
```

## 5. useRoute

`useRoute` ត្រូវបានប្រើបាបទាញយកពី `route` URL ដើម្បីកំណុងបង្ហាញនៅលើ Browser បច្ចុប្បន្ន។ អ្នកមិនអាចប្រើបាបជីវិះបញ្ជីព័ត៌មានទេ តើសម្រាប់តែ "មីនា" ប៉ុណ្ណោះ។

- `params`: តីម្លៃប្រចាំលក្ខុង URL ទាំងអស់: `id` គឺជា `/user/[id]`
- `query`: តីម្លៃបន្ទាប់ពីសញ្ញាស្ថា ? ទាំងអស់: `search` គឺជា `?search=nuxt`
- `path`: ជាបន្ទាប់ពី URL
- `name`: លើរបស់ Route

```
<script setup>
const route = useRoute()

// ប្រើបាប URL តី /blog/my-post ?lang=km
console.log( route.params.slug ) // លទ្ធផល: "my-post"
console.log( route.query.lang ) // លទ្ធផល: "km"
</script>

<template>
<p>អ្នកកំណុងមីនាមួយ: {{ route.params.slug }}</p>
</template>
```

## 6. useRouter

```

edited article by Mr. limvichet, contact 012856917/067878118.

<script setup>
const router = useRouter( )

const goToContact = ( ) => {
  // ចូរទៅទីតាំង contact
  router.push( '/contact' )
}

const goBack = ( ) => {
  // ចូលរក្សាយ ទៅមុន
  router.back( )
}
</script>

<template>
<button @click="goToContact">ទៅកាន់ Contact</button>
<button @click="goBack">ចូលរក្សាយ</button>
</template>

```

## 7. navigateTo

ទៅបីជាវើដអាប៊បី `router.push()` បានកើតឡើយ បុំនែកគួង Nuxt 4 គឺណានាឱ្យបី Utility មួយណាយៈថា `MapsTo()` ដំឡើស វិញ ព្រមទាំងផែកការប្រាកបដោយការផ្តល់ព័ត៌មានទៅក្នុង Server-side និង Client-side។

```
await navigateTo('/dashboard')
```

### របៀបការប្រើប្រាស់

ចូរអ្នកបង្កើត Project តូចចូលយកដែលមានរបនាសម្ព័ន្ធ Routing ដូចខាងក្រោម៖

- `app/pages/index.vue` ទំនើបដើម
- `app/pages/books/[id].vue` ទំនើបព័ត៌មានលម្អិតសៀវភៅករ
- `app/pages/search.vue` ទំនើបស្រួលរក
- `app/pages/...slug.vue` ទំនើបសម្រាប់បាប់ URL ដែលមិនត្រឹមត្រូវ