**Tips, NextJs**

Table of Contents

[**1.** **Create SSH keys** 2](#_Toc111650314)

[**2.** **Tạo ổ đĩa cài win 10 UEFI** 2](#_Toc111650315)

1. **Create SSH keys**

* Open bash (window)

ssh-keygen -t ed25519 -C "<comment>"

ssh-keygen -t rsa -C "tan\_laptop”

* MacOS

tr -d '\n' < ~/.ssh/id\_ed25519.pub | pbcopy

* Git Bash on windows

cat ~/.ssh/id\_ed25519.pub | clip

cat ~/.ssh/id\_rsa.pub | clip

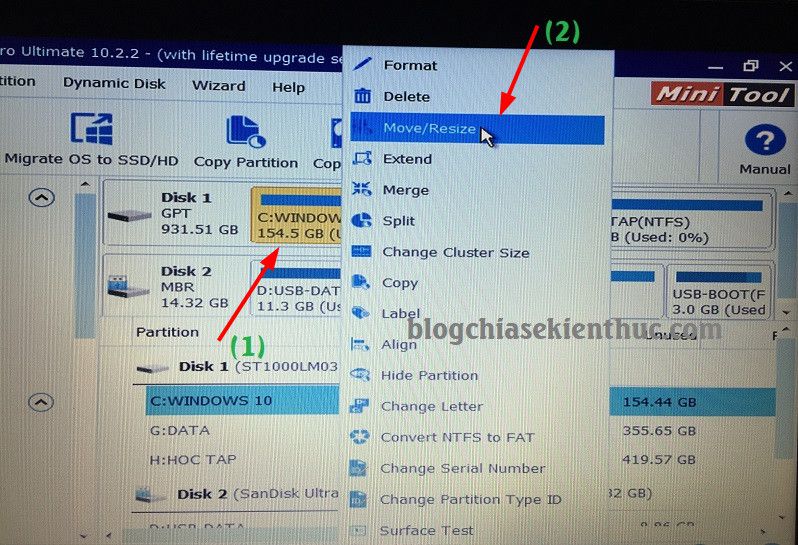
1. **Tạo ổ đĩa cài win 10 UEFI**

**Note:**

* Partition Winzard phải từ bản 10 trở lên mới có change Type ID,

- Các phân vùng khởi động của máy tính chuẩn UEFI đều phải nằm trước phân vùng hệ điều hành.

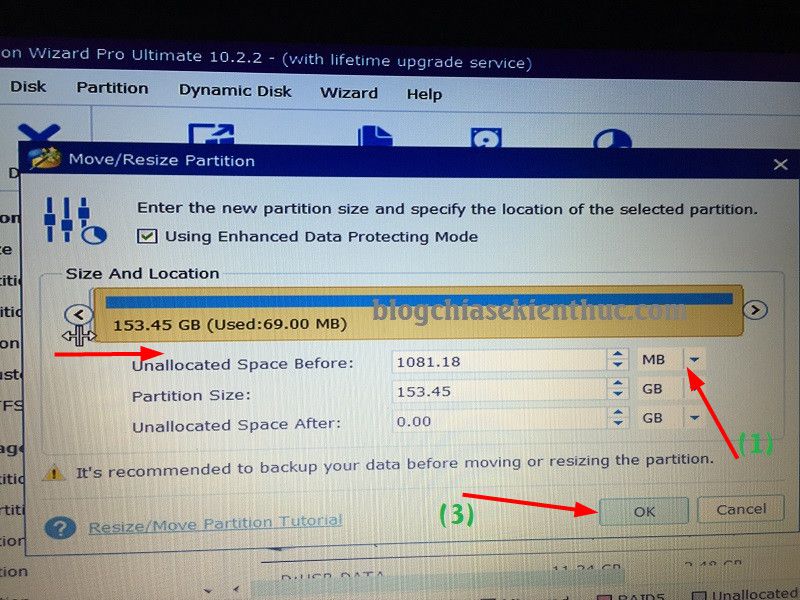
Bạn mở phần mềm Partition Winzard ra => nhấn chuột phải vào phân vùng ổ C (phân vùng chứa hệ điều hành) => chọn Move/ Resize



Bạn hãy chia ra khoảng 700 MB – 1000 MB bằng cách:

1. Tại phần **Unallocated Space Before**: Ban chọn đơn vị là MB.
2. kéo từ trái sang phải, lấy khoảng 700 – 1000 MB.

**Lưu ý:**  Chính vì thế các bạn hãy làm chính xác bước này nhé.



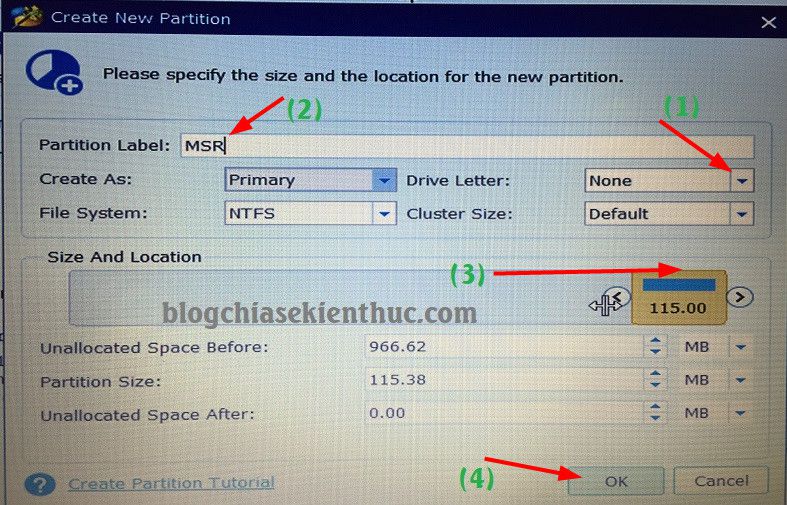
Okay, một phân vùng mới vừa được tạo ra có dung lượng 1.1 GB (1000 MB) như hình bên dưới.

#### **+ Bước 1: Tạo phân vùng MSR**

Bạn nhấn chuột phải vào phân vùng vừa mới được chia đó => chọn Create để tạo mới 1 phân vùng.

Tiếp theo, ở cửa sổ Create New Partition bạn nhập các thông tin như hình bên dưới.

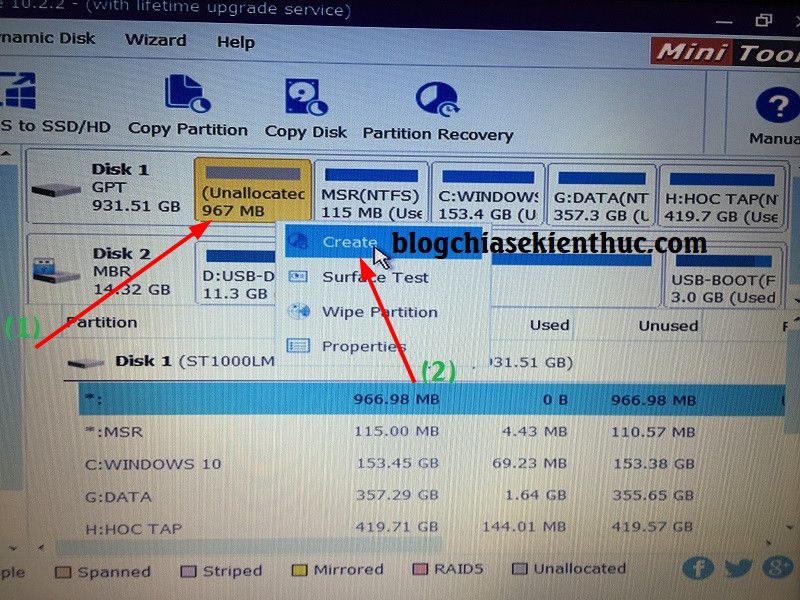
* **Driver Latter:** Bạn để là None.
* **Partition Label:** Bạn đặt tên là MSR.
* **Size And Location:** Bạn đặt con trỏ chuột để nó xuất hiện mũi tên 2 chiều như hình bên dưới =>  kéo từ trái qua phải và giữ lại tầm 20 - 100 MB thôi.



#### **+ Bước 2: Tạo phân vùng boot EFI cho chuẩn UEFI**

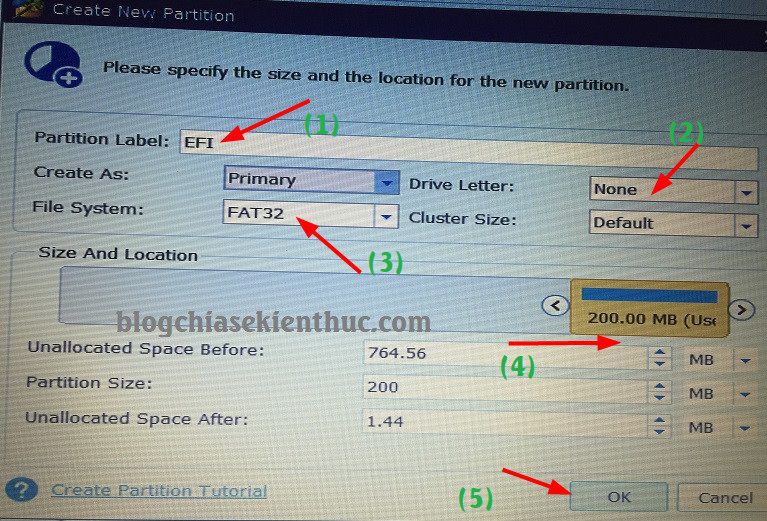
**Thực hiện:**

Nhấn chuột phải vào phân vùng Unallocated => chọn Create.



Bây giờ bạn hãy thiết lập chính xác như sau:

1. **Partition Label:** Bạn đặt tên phân vùng này là EFI.
2. **Create As:** Bạn để là Primary.
3. **Drive Letter:** Bạn để là None.
4. **File System:** Bạn để là FAT32.
5. Kéo theo hướng mũi tên từ trái qua phải, bạn lấy khoảng 200 MB làm phân vùng boot.
6. Sau đó nhấn OK để đồng ý.

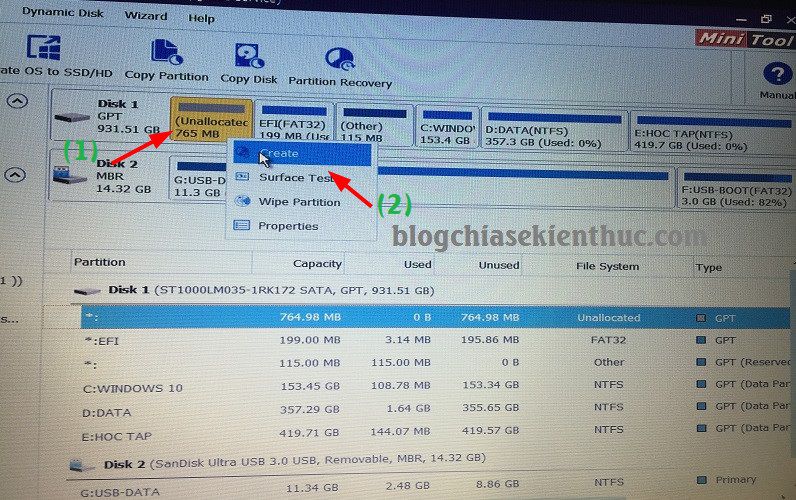


#### **+ Bước 3: Tạo phân vùng Recovery cho ổ cứng GPT**

Tiếp tục, ở phân vùng trống (chưa được định dạng) còn lại, chúng ta sẽ để làm phần vùng Recovery. Phân vùng này sẽ giúp bạn [Reset lại Win](https://blogchiasekienthuc.com/thu-thuat-may-tinh/cach-reset-windows-10.html) và [sửa lỗi khởi động khi bị lỗi Windows](https://blogchiasekienthuc.com/thu-thuat-may-tinh/sua-loi-may-tinh-khong-khoi-dong-vao-duoc-windows.html).

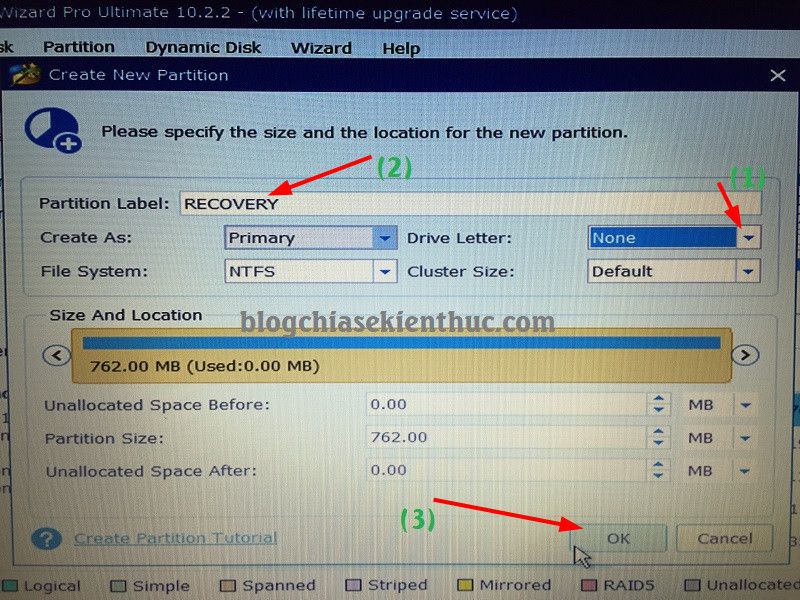
**Thực hiện:**

Bạn nhấn vào phân vùng Unallocated => chọn Create như hình bên dưới.



Bạn thiết lập như sau:

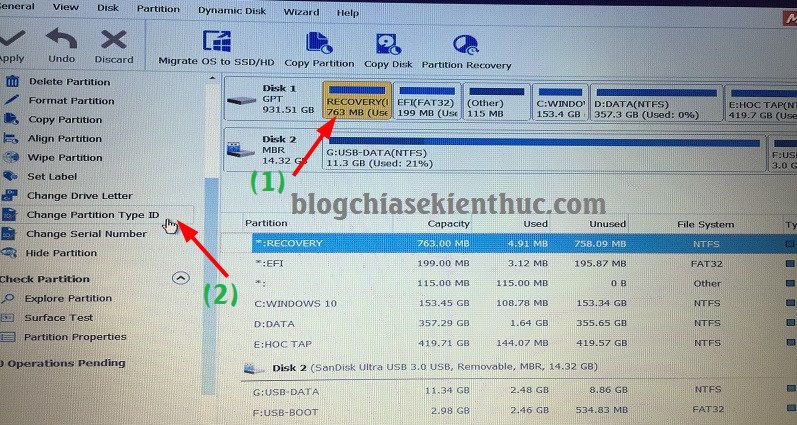
* **Partition Label:** Bạn đặt tên là RECOVERY.
* **Drive Letter:** Bạn để là None
* Nhấn OK để đồng ý.



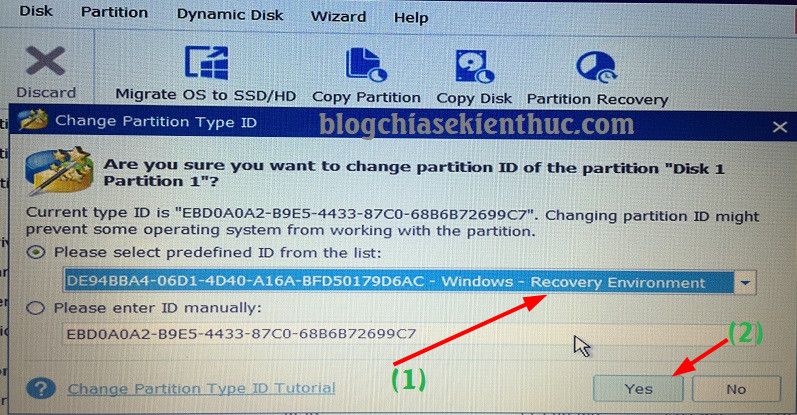
**+ Bước 5:** Bây giờ chúng ta sẽ thay đổi ID cho từng Partition mà chúng ta vừa tạo để nó về đúng với định dạng và chức năng của nó.

**Thực hiện:**

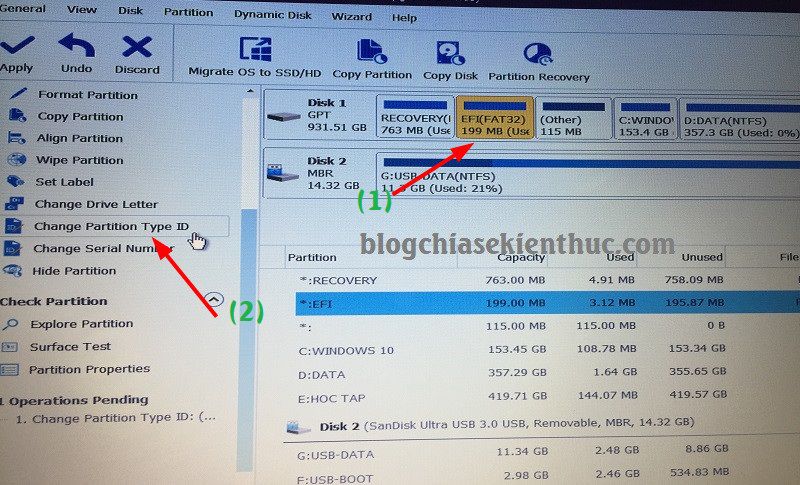
Chọn phân vùng RECOVERY => chọn Change Partition Type ID như hình bên dưới.



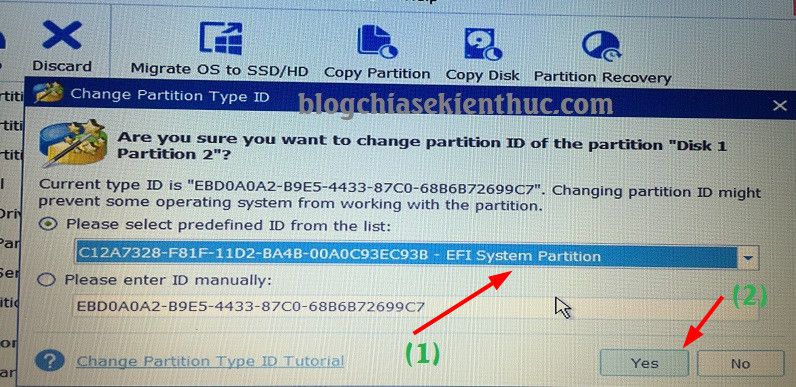
Cửa sổ **Change Partition Type ID** xuất hiện, đối với phân vùng RECOVERY thì bạn chọn ID là Windows - Recovery Environment => chọn YES.



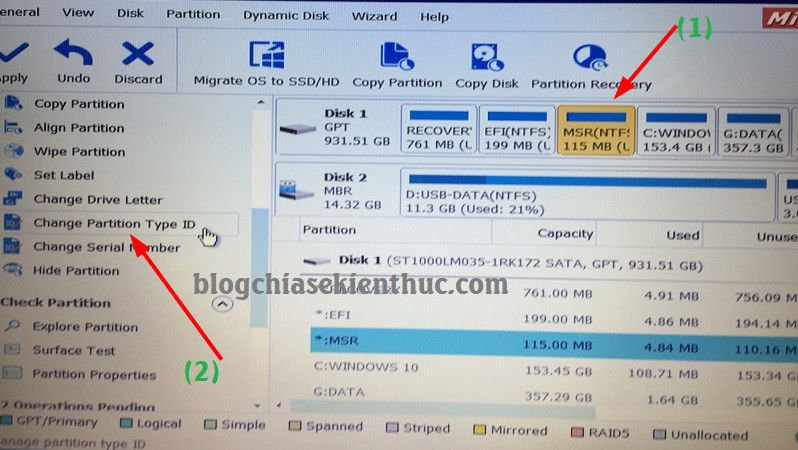
**+ Bước 6:** Tiếp tục, chọn phân vùng EFI => nhấn vào Change Partition Type ID



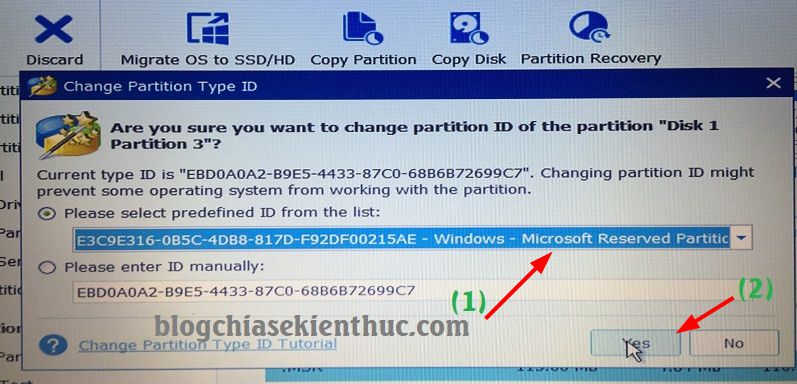
Tiếp tục gán ID cho phân vùng EFI là EFI System Partition => chọn Yes để đồng ý.



**+ Bước 7:** Chọn phân vùng MSR => chọn tính năng Change Parition Type ID



Bạn gán ID cho phân vùng này là Windows - Microsoft Reserved Partition => chọn Yes.



**+ Bước 8:** Bạn hãy nhấn vào nút Apply như hình bên dưới để áp dụng toàn bộ thay đổi bên trên.

1. **Convention trong nextjs**

Nextjs là 1 framework nên sẽ có các quy định và các rules để tuân theo

* **Folder structure**:
  + [id] là chỉ các dynamic route,
  + [...segments] là chỉ catch-all nested routes: admin.com**/user/uhy133dqo/them-moi**
  + ở nextjs 13 cách đặt tên file cũng khác. Mặc định là **page.tsx** (tương đương như index.tsx)
  + ngoài ra còn có các loại tên file: loading.tsx, not-found.tsx
* **Cách đặt tên**:
  + \_components: \_ là private folder, nextjs sẽ ko xem các folder này là route
  + (routes): group route, vẫn coi các folder là route

1. **SEO trong nextjs**

Nextjs nổi bật nhất là cái SEO. Nên hỗ trợ mấy cái metadata, site-map, robust rất tốt.

1. **Cách export trong nextjs**

Có 2 kiểu named export và default export

* **Default export**: nên dc sử dụng cho các **Page Component, Layout Component.** Nhằm để nextjs nhận bít hàm default đó và sử dụng như mặc định.1 vài trường hợp khác có thể dùng **Default export:** Làm Layout Component để sử dụng <**Suspense>,** export default new Class()

*Note****:*** *khi import để xài phải gắn* ***alias*** *cho file đó*

* **Named export**: nên dc sử dụng cho các UI Components. Nhằm để quản lý các components tránh trùng tên, dễ quản lý  
  *Note: có thể* ***export \* from ‘../../’***

1. **Server component và client component**

|  |  |
| --- | --- |
| Server component | client component |
| * No directive * Có thể viết function call API ở đây * Ko thể dùng event * Gắn data của api và truyền prop xuống cho component. * Component có thể async await và trả về only phần JSX/html * Chỉ có thể truyền xuống, chứ ko thể nhận lên event/handler function.   https://stackoverflow.com/questions/77183720/passing-events-from-client-component-to-server-component-to-handle-it | * “use client” * có thể dùng các hook như viết code React * có thể dùng các event: onClick,onChange * truyền prop theo bất cứ type nào cũng dc (function, obj, array,...) * ko thể truyền ngược lên event/handler function cho server component nên có thể gọi api để xử lý tiếp |