# Note Buổi 2.

## UI System (Canvas)

\_ Canvas có 3 Render Mode:  
+ World Space: hệ quy chiếu với Unity Unit. ()

+ Screen Space - Overlay: hệ quy chiếu với Pixel màn hình. (dp trong Android)

+ Screen Space – Camera: dùng để tạo HUD (Head Up Display)

\_ Canvas Scaler: Dùng để scale Canvas tùy theo các cỡ màn hình khác nhau.

\_ Pivot: Điểm thuộc Object Child để xác định vị trí ghim Object Child. (của con)

\_ Anchor: Điểm thuộc Object Child để biết vị trí của ghim Object Child lên Object Parent (của bố)

Ví dụ: A là con của B. Thì trong A sẽ có thông số:

+ Pivot là vị trí ghim trên A .

+ Anchor là vị trí ghim A lên B.

\_ Luôn luôn Set Native Size khi đưa ảnh vào Canvas để giữ Ratio ( tỉ lệ ) của ảnh gốc. Sau đó điều chỉnh cho hợp lý.

## Unity tổng quát

\_ Khi kéo 1 object trong Hierarchy vào Tab Project thì sẽ tạo ra 1 Prefab (Bản mẫu) và có thể kéo thả cái Prefab này ra Scene để tái sử dụng (tạo ra Object Clone).

\_ Bất kỳ chỉnh sửa nào trong Bản mẫu sẽ thay đổi những Object Clone từ Bản mẫu.

\_ Vòng đời của game:

+ **Start()** làm nhiệm vụ khởi tạo khi đối tượng vừa được đặt vào màn hình

+ **Update()** làm nhiệm vụ cập nhật đối tượng liên tục trong vòng đời của game

\_ Reference: khi khởi tạo 1 biến Public GameObject ở trong 1 Script gắn trong 1 Object, sẽ có 1 ô để kéo thả Object khác vào hiện ra trong Component Script đó. (dùng để Binding Object trong Scene với biến trong Script)

\_ Khởi tạo các biến Public với kiểu dữ liệu đơn giản (int, float, string) thì trong tab Inspector ngoài cửa sổ Editor sẽ hiện ra các biến có thể thay đổi giá trị (giúp việc phối hợp vs Game Designer).

\_ Lưu ý khi sử dụng TKLib để chuyển Scene thì nhớ kéo các Scene muốn chuyển vào trong Build Setting.

\_ Muốn ẩn hiện (Active/Deactive) Game Object thì có thể sử dụng GameObject.SetActive(true/false);

\_ Muốn lấy 1 Component trong 1 GameObject thì có thể dùng hàm GetComponent<Tên Component>();