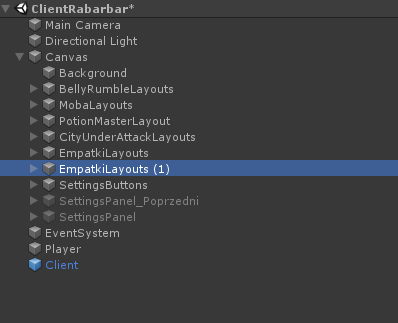
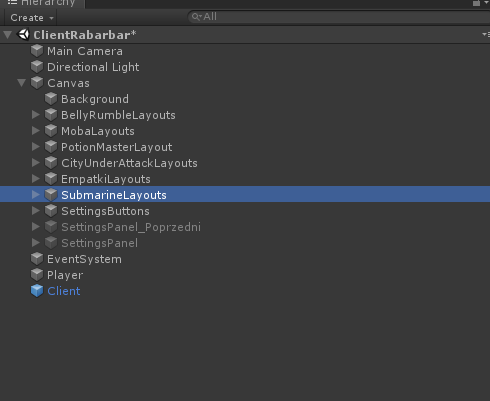
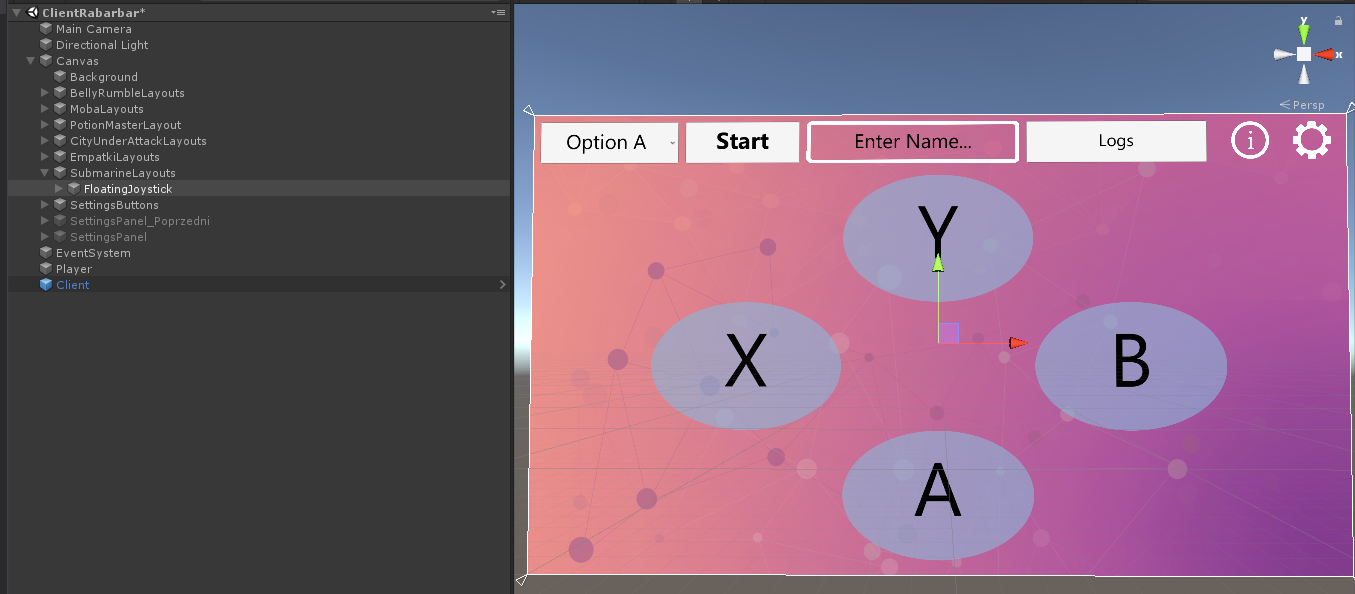
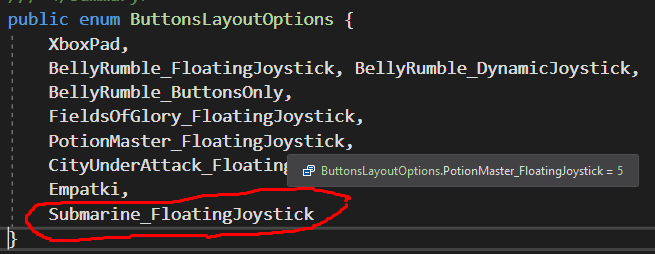
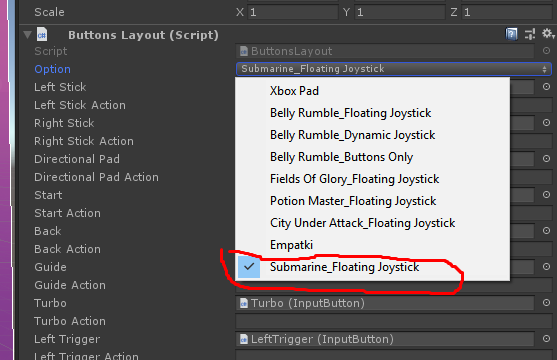
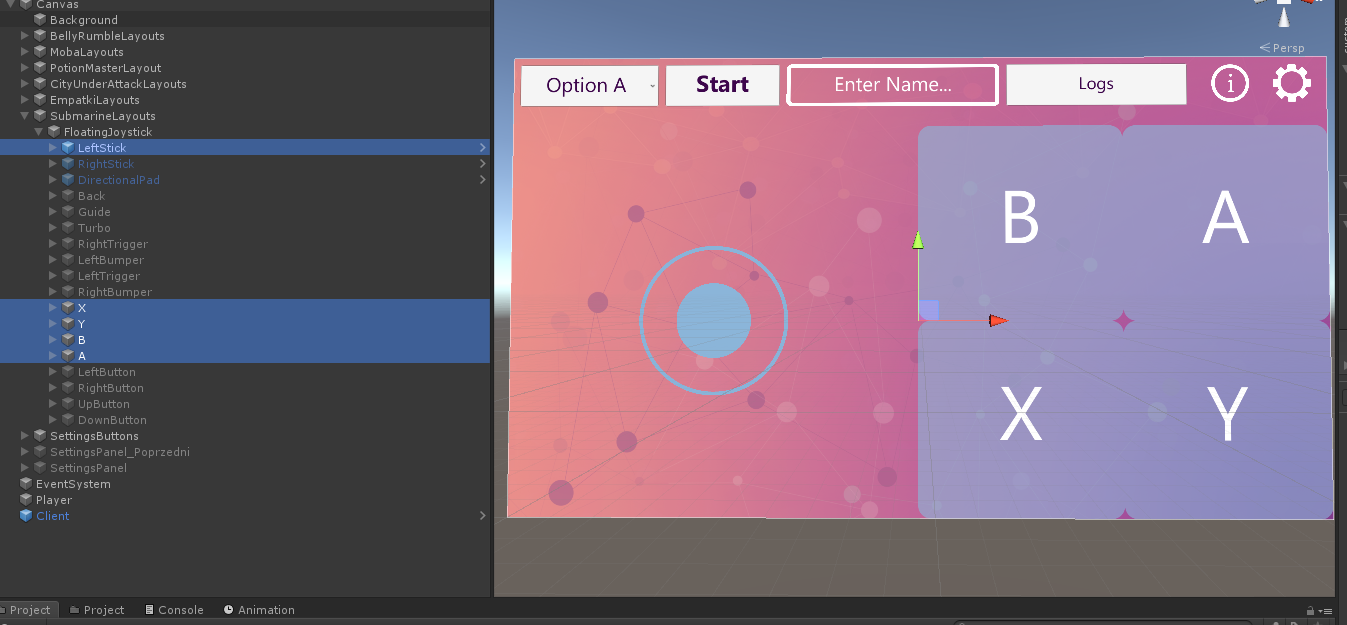
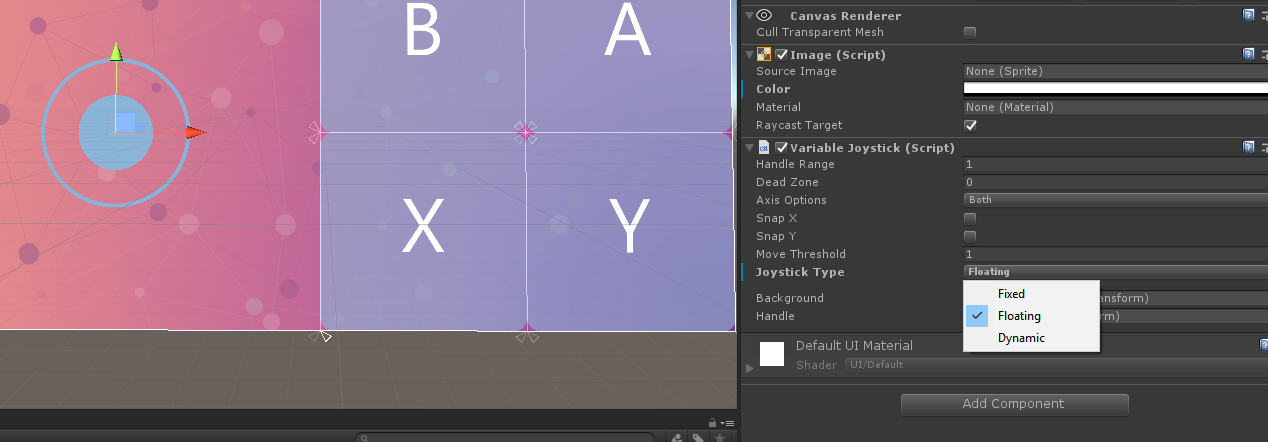
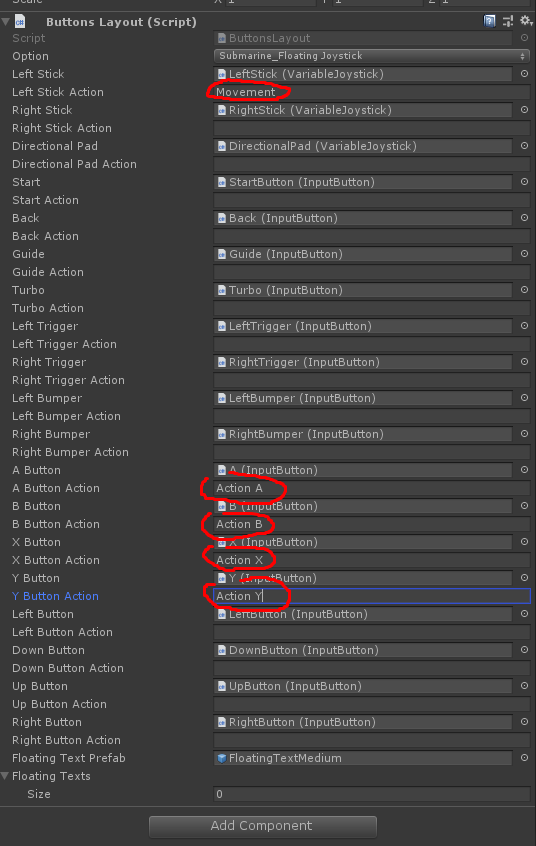
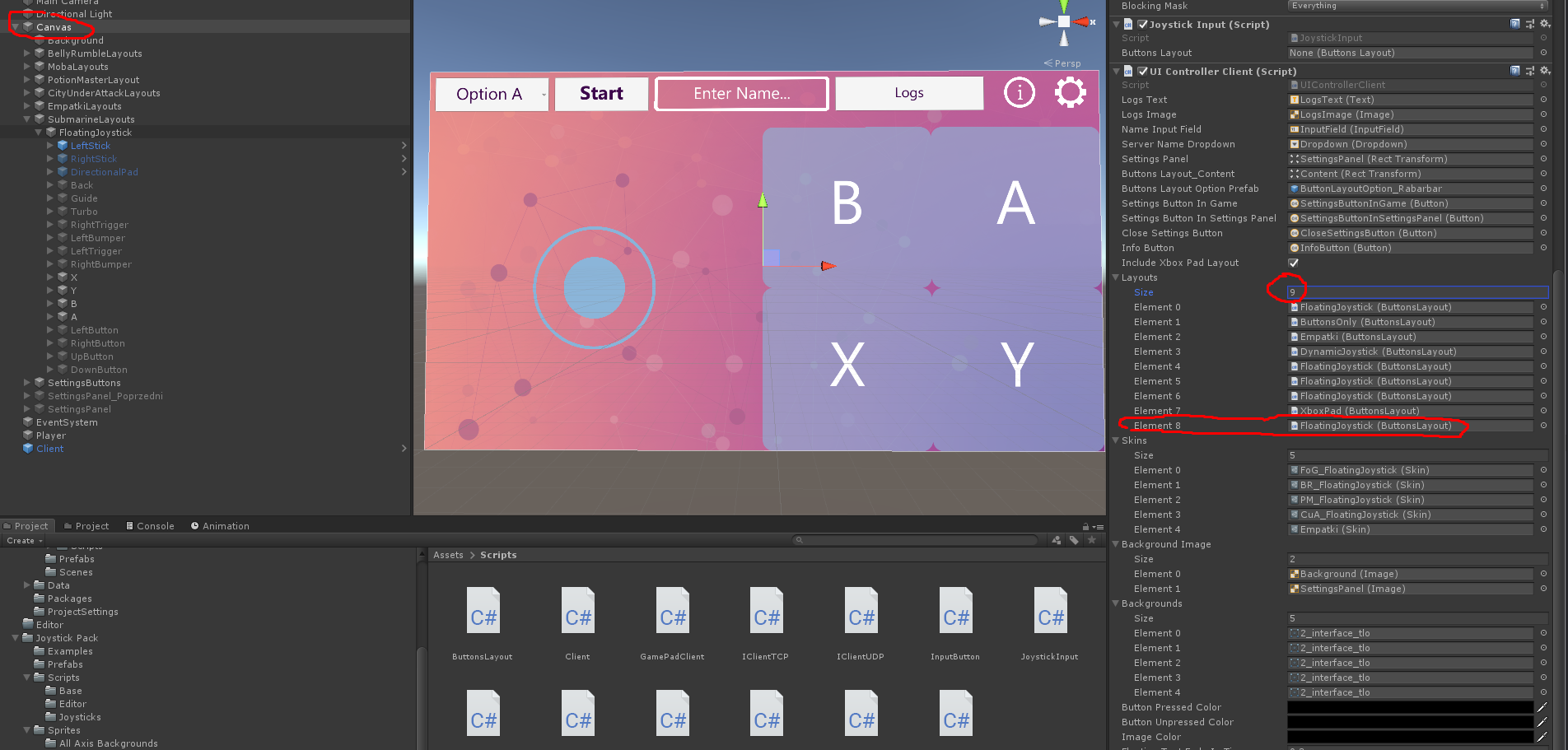
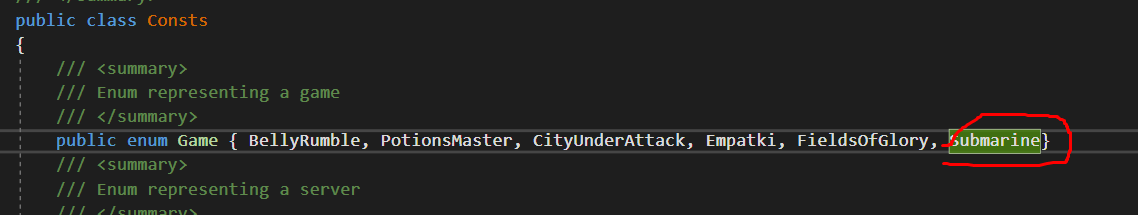
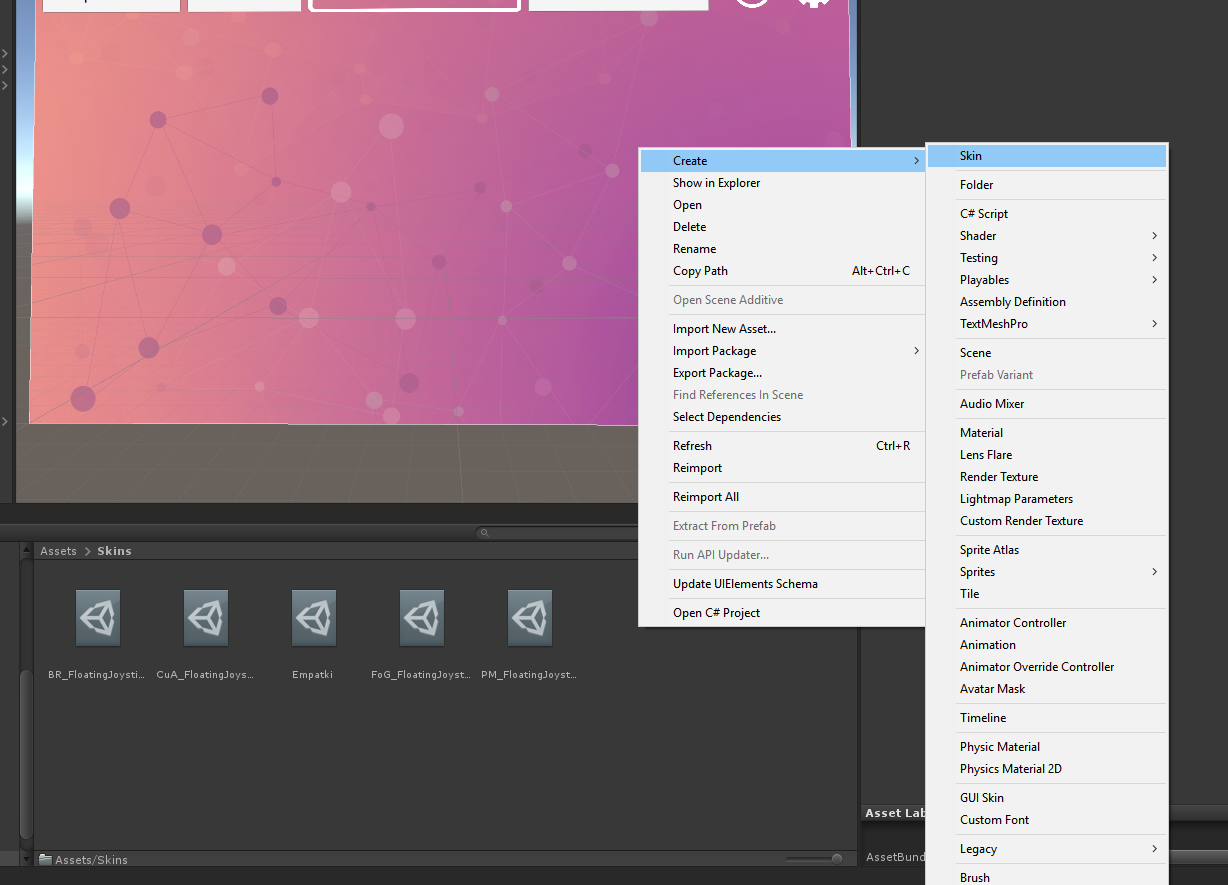
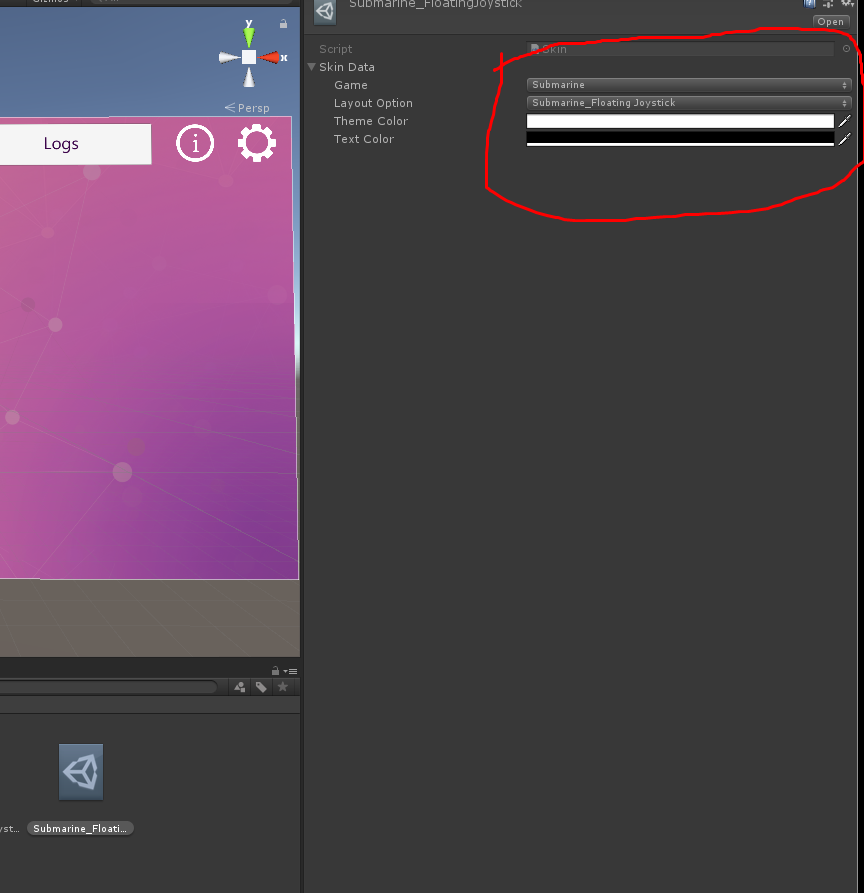
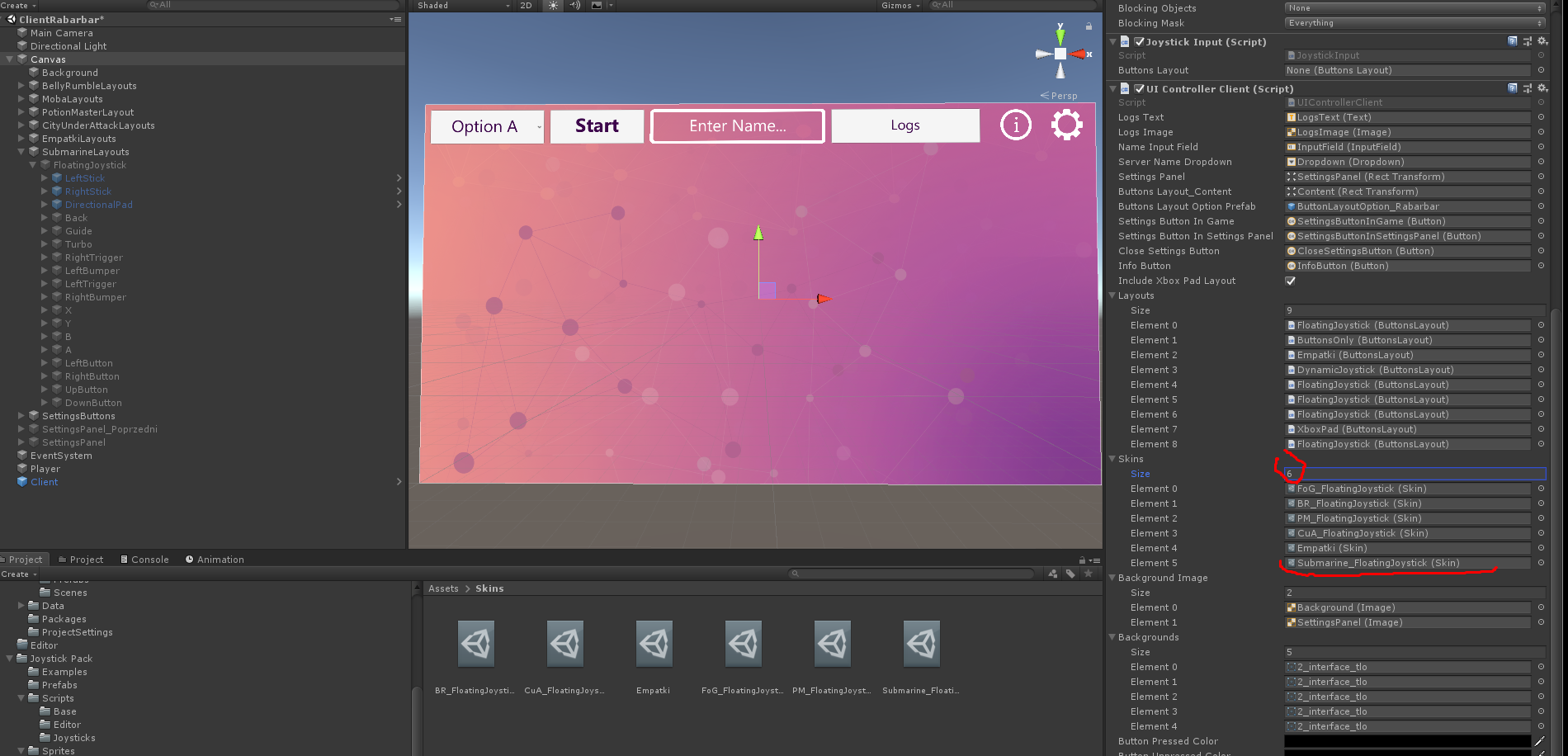
# Jak dodać nowy ButtonLayout

* Skopiuj jedną z istniejących grup layout’ów



* Nazwij adekwatnie
* Każde z dzieci grupy layout’ów jest osobnym layout’em np. FloatingJoystick, ButtonsOnly. Nazwij layout adekwatnie i aktywuj go, aby był widoczny
* Otwórz skrypt ButtonsLayout siedzący na każdym layoucie i dodaj nazwę layoutu do enuma ButtonsLayoutOptions
* Dla tworzonego layoutu ustaw publiczne pole Option na przed chwilą dodane:
* Aktywuj przyciski których potrzebujesz i zmień ich rozstawienie wedle potrzeb
* Możesz zmienić typ Joysticka
* Podmień stringi Action dla przycisków, dla których chcesz aby wyświetlała się instrukcja po naciśnięciu przycisku Info
* Na obiekcie Canvas w skrypcie UIControllerClient zaciągnij nowy layout na koniec tablicy Layouts
* Dezaktywuj stworzony Layout
* Jeśli tworzysz layout do nowej gry i chcesz by był kojarzony z grą musisz stworzyć grę i skin
* Otwórz skrypt Consts i dodaj do pola enum Game nową grę
* W folderze Skins utwórz nowy skin
* Dla nowego skina ustaw pola Game, Layout option oraz kolory Theme (kolor przycisków w grze jak joystick, A,B,X,Y) i Text (kolor tekstu na przyciskach)
* Na obiekcie Canvas w skrypcie UIControllerClient dodaj nowy skin do tablicy Skins



Gotowe