<u>Road Map</u>

There are 2 main files for this program:

1. MainWindow.cs

- <u>Description</u>: This is where I wrote all of my code that was not for the UI. This handles all of the functionality of the program.
- <u>Important Functions:</u>
    + Event Handlers Triggered By Users: **FreestyleOnClick, StartFreestyleBtnClick, BackBtnClicked, MoveOnClick, StartBtnClick, ReturnBtnClick, EndRoutineBtnClick, RoutineOnClick, PartnerBtnClick**
    + **WindowLoaded** - Detects the Kinect sensor, sets event handler for detecting skeletons
    + **SensorSkeletonFrameReady** - When a skeleton is detected *and* can be tracked, this method calls DrawBonesAndJoints to create the skeleton. This function is an event handler that fires every time a skeleton is detected.
    + **InitializeMoveMaking** - The backbone of "Bust A Move," this function handles checking if a user is ready to start recording a dance move and then stores these coordinates in a series of lists. It then checks if a user is done recording. If so, it calls WritingJointsToFile to write the data into text files.
    + **DrawBone** - Draws the bones of the skeleton to the screen and also changes the colors of the skeleton.
    + **dtClockTime_Tick & dtClockTime_TickPartner** - Redraws the stored moves onto the screen, called from the **RoutineOnClick** and **PartnerBtnClick** event handlers, respectively

2. MainWindow.xaml.cs

- This file handles the UI elements, including the home screen, instructions, and the five circles in "Bust A Move"
- Due to the WPF style application, there is only one page, with UI elements being hidden and reappearing when necessary.

Libraries:

I utilized the Kinect library from a hollowed-out version of the **SkeletonBasics WPF** application. I removed a lot of code from this application and reduced it, but I maintained the library so that I would not have to worry about what I was including.