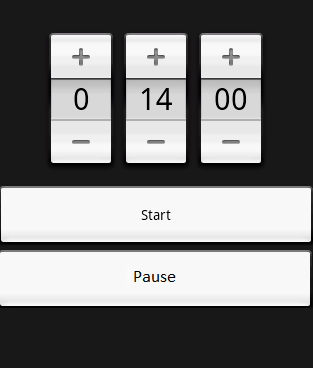
**Assignment # 3**

For this assignment, your Android program is required to implement three activities: (1) An Activity that serves as main menu; (2) A Tick-Tack-Toe game; (3) A Timer. The interface for your timer should look as shown below:



**Requirements:**

The Main Menu activity should make use of Intents to switch/select between the two other Activities (Tick-Tack-Toe, Timer)

**For the Timer**

* The Timer is a countdown timer. When it gets to 0, it should present a Toast message (the message is up to you). You can also look how to make the device to vibrate (optional). The buttons at the top should help you to change the amount of time to countdown. The rightmost is seconds, the middle is minutes, and the other is hours.
* As the timer starts, it should update the values accordingly in the interface (e.g., after one second, the interface here should show 13 in the middle and 59 on the right.
* When the user hits pause, the timer should stop and keep the current time. At that point the user can change the values of the timer, hit start and it should continue to execute.
* Don’t worry if the timer stops working if the user closes the timer screen. The goal with this project is to practice the update of UI from threads/AsynchTasks/Handlers.
* Don’t use the Timer class from Android for this program. The whole idea is to practice threads.

Main Activity

Tic-Tac-Toe-Activity

Timer Activity

**For the TicTacToe**

This part of the program should have the option to play two players or one player against the computer.

If the user selects a two player game, alternating between each player, assume that the players “pass” the phone back and forth. Your code will need to determine winner, loser, tie. You will also be able to play again (play another game) without having to restart the activity.

If the user selects option for the other player to be an AI player. The user will determine if they want to be X or O. The AI does not need to be very smart, but it must be “smarter” then just picking a random

location. You will also be able to play again (play another game) without having to restart the activity.

**Rubric:**

Implementing the Timer: 30%

Implementing the rest of the project: 70%