Use Case U1S1: Manage Countdown Time Events
Scope: Timer
Level: User Goal
Primary Actor: User
Scenario: Start the Countdown Timer
Related Use Cases: N/A
Stakeholders and Interests:

- User: Wants an accurate countdown time

Preconditions: The Countdown Timer is up, a time is entered, (See *Use Case 1S4: User Enters the Countdown Time*), the Countdown Timer is currently not running.
Success Guaranties: (Postconditions) The Countdown Timer is running, and accurately counting down the time (to 0 seconds).

Main Success Scenario:

User	**System**
1. Indicates to start the timing countdown.	
	2. Records the Countdown time (See *Use Case 1S4: User Enters the Countdown Time*).
	3. Subtracts (Counts Down) the time in one-second increments.
	4. Displays the new time (currenttime = previous time-1sec).
	5. Repeats 3 & 4 until the currenttime is 0
	6. Alerts the User the countdown is complete

Alternative Flows:
*a. If at any time, the System cannot measure the one second increment for the countdown, then the System informs the User the Countdown Timer has stopped working properly.

Technology and Variations List:
2b-6b. The Actual accuracy of the timer is dependant and will vary based on the task schdulers of both the Operating System and the language used for implementation.

Frequency of Occurrence: Multiple Countdown Timers as needed.

Open Issues:

- Is the accuracy dependant on the Operating System?
- Is the accuracy dependant on the Hardware?
- Multitasking issues?
- Multithreading issues?
- How does language affect the accuracy of the Countdown Timer?
- How does a “fully loaded System” affect the accuracy of the Countdown Timer?