**Purpose:** Provide a layer over the servo classes provided by the FTC libraries. Simmilar to ‘com.lmrobotics.devices.Motor’, this will be useful for both allowing us to future-proof the code and creating stub methods for testing purposes.

|  |  |
| --- | --- |
| **Priority:** High | **Reason:** Servos will be used for many of the manipulators used for scoring. |

**Primary Programmers:**

* Brian

**Public Constants (public static final):**

* int MAX\_POS
  + Purpose: Provide the maximum position value a servo can be set to.
* int MIN\_POS
  + Purpose: The lowest position value a servo can be set to. (Could be ≤ 0)

**Constructors (called when an object instance is created):**

* public Servo(1 arguments)
  + Must Initialize: Whatever needs to be initialized in the FTC servo library for a servo to work, and store the information to access the servo, which could be a reference to the servo instance from the FTC libraries.
  + Arguments:
    - <type> port
      * The port or location the servo is connected at

**Interface Instance Methods (used on an instance of this class):**

* public synchronized void setTarget(1 argument)
  + Purpose: To set the position the servo will move to. This should return immediately after setting the servo target; it should not wait for the servo to stop moving.
  + Priority: Very High (Needed for basic control of the servo)
  + Arguments:
    - int targetPos
      * The position to move the servo to (see also: MIN\_POS and MAX\_POS)
  + Returns: N/A
* public synchronized int getCurrentPos(0 arguments)
  + Purpose: Get what position the servo is at when this method is called.
  + Priority: Medium (not vital to using a servo, but could easily be used for tracking a servo during autonomous and possibly teleop)
  + Arguments: N/A
  + Returns: (int) the current position of the servo
* public synchronized boolean isMoving(0 arguments)
  + Purpose: Determine if the servo is still moving to reach the target position; servo must be within a few degrees of the target position before this will return true.
  + Priority: High (very likely to be used to score during autonomous and possibly during teleop)
  + Arguments: N/A
  + Returns: (boolean) if the servo is still moving to the target position