Class Name	Description	Status
SensorManager	Handles all sensor inputs (FSRs, potentiometers)	To be written
MotorController	Controls the servos/stepper motors for finger movement	Written
GestureRecognition	Processes input signals to recognize hand gestures	To be written
SafetyHandler	Implements safety mechanisms (force limits, error handling)	To be written
FeedbackSystem	Manages haptic feedback and alerts	To be written
MainProgram	Controls the entire system flow, interacting with all other classes	To be written

SensorManager

Need commands for sensor inputs and filtering of noise.

- read_FSR() → float – Reads force sensor data.
- read_Potentiometer() → float – Reads potentiometer values.
- filter_data(sensor_data: float) → float – Applies a noise filter.

MotorController

Controls/Regulates servos or stepper motors.

- move_finger(finger_id: int, angle: float) → void – Moves a finger to a specific angle.
- set_grip_strength(force: float) → void – Adjusts grip force dynamically.
- stop_movement() → void – Stops all motors in case of emergency.

SafetyHandler

Ensures safe usage of the prosthetic hand/accounts for when the hand needs to be immediately shut down.

- check_force_limits(force: float) → bool – Returns True if force is within safe limits.

- trigger_emergency_stop() → void – Activates emergency stop.

FeedbackSystem

Manages feedback mechanisms/ wants a mechanism that lets the user know that the sensor is receiving information.

- vibrate_motor(intensity: int) → void – Activates haptic feedback.
- display_alert(message: str) → void – Sends a warning message.

MainProgram

Orchestrates the entire system/ allows the calling of all classes.

- initialize_system() → void – Initializes hardware components.
- run_control_loop() → void – Runs the main program loop.

```
+-----------------------+
|   Main Program        |
|  (System Controller)   |
+-----------------------+
         |
         v
+-----------------------+
|  SensorManager         | <-- Reads input from sensors (FSR, EMG, IMU, Potentiometer)
+-----------------------+
         |
         v
+-----------------------+
|  MotorController       | <-- Sends PWM signals to servos/stepper motors
+-----------------------+
         |
         v
+-----------------------+
|  SafetyHandler        | <-- Triggers emergency stop and adjusts for the limits of force
+-----------------------+
         |
         v
+-----------------------+
|  FeedbackSystem        | <-- Provides feedback from the sensor
+-----------------------+
```