This Matlab app models how a spring-damper system behaves when controlled using a PID controller. Using input parameters (mass, damping, spring constant) and input PID gains (kP, kI, kD) we control how the system responds to a setpoint. The PID controller adjusts the force applied to the mass based on the error between the current position and the desired one. The app runs a simulation and plots the system's position over time, allowing you to see how different parameters affect the system's stability, speed, and accuracy.

To find the best PID values, I tested different combinations while keeping the system's mass, damping, and spring constant within realistic ranges. Through trial and error, I noticed a clear pattern: increasing kP makes the system respond faster, but too much causes overshooting and oscillations; increasing kI reduces long-term error but can lead to instability if it's too high; and increasing kD helps smooth out the motion and reduce overshoot, though it can slow the response. By balancing these three values, I found that kP between 30–50, kI between 0.1–0.5, and kD between 5–10 gave the best overall results for most setups.