Deep Work Guardian

A Multi-Agent AI System for Protecting Focus, Health & Privacy

Team: Muhammad • Abdulla • Shabaz • Peshawa • Rasyar

Python | OpenCV | PyAudio | Pygame | psutil | Tkinter | Multi-Threading

The Problem

😩 Students sit too close to the screen → eye strain & back pain

👀 Someone looks at your screen → privacy breach

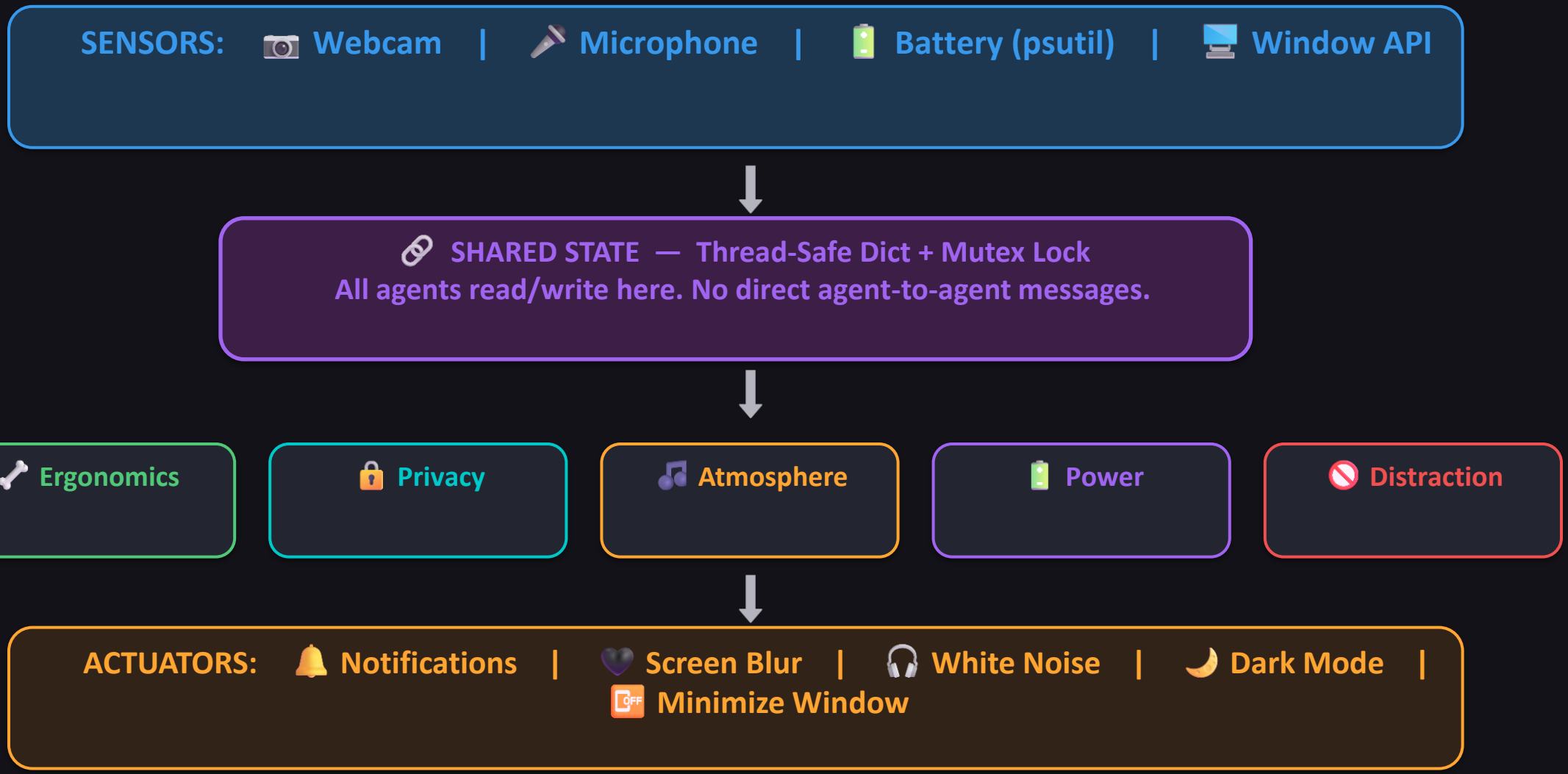
📢 Noisy study environment → loss of concentration

🔋 Laptop unplugged → battery drains fast, dies mid-session

📱 Social media & distracting apps → hours of wasted time

✅ **Our Solution: 5 AI agents that monitor sensors and automatically protect you in real-time.**

System Architecture



The 5 Agent Subsystems

🦴 Agent 1 — Ergonomics Monitor

- Sensor: Webcam (shared frame)
- Algo: Haar Cascade + Pinhole Model
- Rule: If face < 50 cm → Notify
- Cooldown: 2 min between warnings
- Owner: Muhammad

🔒 Agent 2 — Privacy Shield

- Sensor: Webcam (shared frame)
- Algo: Haar Cascade face count
- Rule: If faces > 1 → Blur + Minimize
- Actuator: Tkinter overlay + PyGetWindow
- Owner: Abdulla

🎵 Agent 3 — Atmosphere Controller

- Sensor: Microphone (PyAudio)
- Algo: RMS → Decibel conversion
- Rule: If noise > 60 dB → White Noise
- Actuator: Pygame mixer (NumPy buffer)
- Owner: Shabaz

🔋 Agent 4 — Power Optimizer

- Sensor: psutil battery API
- Algo: State transition tracking
- Rule: Unplugged → Dark Mode + 30% brightness
- Actuator: PowerShell registry + WMI
- Owner: Peshawa

🚫 Agent 5 — Distraction Blocker

- Sensor: Active window title (PyGetWindow)
- Algo: Blocklist keyword matching
- Rule: 5 min on distraction → Minimize + Notify
- Blocklist: YouTube, Instagram, TikTok...
- Owner: Rasyar

Inter-Agent Communication — Shared State

All agents communicate through a single SharedState object — a Python dictionary protected by a threading.Lock (Mutex). No agent calls another agent directly.

Ergonomics →	face_distance_cm, posture_warning_active
Privacy →	background_face_detected, screen_blurred
Atmosphere →	noise_level_db, white_noise_playing
Power →	battery_percent, is_charging, dark_mode_enabled
Distraction →	active_window, distraction_timer, app_blocked

Methods: get(key) → read with lock | set(key, value) → write with lock | get_all() → copy entire state

Key Technical Concepts

Multi-Threading	All 5 agents + webcam run as daemon threads in parallel
Mutex Lock	threading.Lock prevents race conditions on shared data
Haar Cascade Classifier	OpenCV face detection algorithm (Agents 1 & 2)
Pinhole Camera Model	distance = (known_width × focal_length) / pixel_width
RMS → Decibels	$20 \times \log_{10}(\text{RMS})$ converts audio samples to dB (Agent 3)
Windows Registry	PowerShell modifies HKCU theme keys for Dark/Light mode
State Transition Tracking	Agents act only on CHANGES, not every loop cycle

Live Dashboard Output

main.py prints a real-time status line every 5 seconds showing all agent data:

```
C:\> python main.py

================================================
   🚀 DEEP WORK GUARDIAN — Starting Up...
================================================

✅ All 5 subsystems running!
Press Ctrl+C to stop
------------------------------------------------

Face: 72.3cm | Noise: 42.5dB | Battery: 85% ⚡ 🌞  | Privacy: ✓ | Window: VS Code
Face: 38.1cm | Noise: 65.2dB | Battery: 84% 🔋 🌙 | Privacy: ✓ | Window: YouTube
Face: 71.0cm | Noise: 31.8dB | Battery: 83% 🔋 🌙 | Privacy: 🔒 | Window: [BLURRED]
```

How to Run:

1. pip install opencv-python numpy psutil pygetwindow plyer pyaudio pygame
2. python main.py
3. The dashboard updates every 5 seconds. Press Ctrl+C to stop.

Thank You!

Deep Work Guardian — Protecting Focus, Health & Privacy

Muhammad • Abdulla • Shabaz • Peshawa • Rasyar

Any Questions?