SOFTWARE REQUIREMENTS SPECIFICATIONS (SRS) Functional Requirements

System Overview: The system enables users to control the mouse cursor and draw on screen using hand gestures captured via webcam. It replaces traditional input devices with gesture-based interaction for improved accessibility and innovation.

Core Features and Operations:

- Real-time gesture recognition using computer vision

- Cursor movement based on hand position

- Drawing functionality triggered by specific gestures

- UI interface for mode switching (e.g., draw, erase, clear)

- Feedback display for gesture detection status

Support for Project Objectives: Each function promotes hands-free interaction, supports accessibility for users with limited mobility, and demonstrates human-computer interaction principles in software engineering.

Workflow Example:

1. User opens the application and activates webcam

2. System detects hand gesture and maps it to cursor movement

3. Specific gesture (e.g., closed fist) triggers drawing mode

4. User draws on canvas using hand motion

5. UI buttons allow switching modes or clearing canvas

Use Case Example:

Use Case Name: Draw with Gesture

Actor: User

 Precondition: Webcam is active

Main Flow:

- User performs "draw" gesture

- System enters drawing mode

- User moves hand to draw **Postcondition**: Drawing appears on canvas

Acceptance Criteria:

- Cursor responds within 0.5 seconds of gesture input

- Drawing activates only with correct gesture

- UI buttons function correctly across modes

- System runs without crashing for at least 30 minutes of continuous use

Non-Functional Requirements

Performance:

- Gesture recognition latency ≤ 500 ms

- System supports continuous input for up to 60 minutes without lag

- Efficient processing using OpenCV and MediaPipe

Reliability:

- System maintains stable performance across lighting conditions

- Error rate in gesture detection ≤ 10%

- Recovery from minor glitches without restart

Integrity/Security:

- No sensitive data stored; webcam access is local only

- User permissions required for camera activation

- No external data transmission

Usability:

- Simple UI with clear buttons and feedback

- Accessible design for users with limited motor control

- Visual cues for gesture recognition status

Maintainability:

- Modular code structure for easy updates

- Clear documentation for each module

- Version control via Git for tracking changes

Scalability:

- System can be extended to support multi-hand gestures

- Future integration with voice commands or touchscreen input

- Potential deployment on mobile or tablet platforms