# Chapter X: Methodology

The methodology followed in the development of the Arduino-based Automatic Glass Door System was systematic and iterative, focusing on integrating hardware components with efficient software logic. The steps involved are as follows:  
  
1. \*\*Requirement Analysis\*\*: Identifying the need for automation, safety, and reliability in a glass door system, especially in environments like offices and institutions.  
  
2. \*\*Component Selection\*\*: Choosing appropriate components such as DC motor with encoder, Cytron MD10C motor driver, inductive proximity sensor, PIR sensor, relays, push buttons, and Arduino Uno.  
  
3. \*\*System Design\*\*: Creating a block diagram and layout for wiring, ensuring optimal placement and functionality of each component. The control logic was designed to support homing, emergency stops, PIR-triggered operations, and encoder feedback-based motor control.  
  
4. \*\*Circuit Assembly\*\*: Wiring all components on a breadboard, handling ground and power lines efficiently, and ensuring proper interfacing of sensors and actuators with Arduino Uno.  
  
5. \*\*Code Development\*\*: Programming the Arduino using C/C++ with logic for homing, speed control using encoder feedback, PID integration, relay handling, PIR motion detection, and button-based interrupts.  
  
6. \*\*Testing & Iteration\*\*: Each part of the system was tested individually and then integrated. Serial monitor outputs were used extensively to debug encoder values, relay logic, sensor triggers, and motor performance.  
  
7. \*\*Final Integration\*\*: After successful testing, all components were merged into a single system with serial output for monitoring and fault diagnosis.

# Chapter X: Testing and Validation

To ensure the reliability and accuracy of the system, several tests were conducted. These tests helped validate both the hardware functionality and software logic.  
  
1. \*\*Homing Sensor Testing\*\*: Verified that the inductive proximity sensor correctly detects the door’s initial position. The encoder resets to zero upon detection.  
  
2. \*\*Encoder Feedback\*\*: The encoder output was monitored during door movement to verify correct counting and direction. Negative values during closing and positive during opening confirmed logical accuracy.  
  
3. \*\*Relay Response\*\*: Ensured relays operated as expected, cutting off power or switching control based on motor direction and safety logic.  
  
4. \*\*Button Operations\*\*: Tested open/close/emergency-stop logic under different states. Buttons were debounced using INPUT\_PULLUP and tested under various user conditions.  
  
5. \*\*PIR Sensor Validation\*\*: Checked PIR detection sensitivity, range, and its ability to reset auto-close timers. Confirmed its responsiveness to human presence.  
  
6. \*\*Motor Driver Control\*\*: Validated motor speed and direction using the Cytron MD10C. PWM and DIR signals were tested at different duty cycles.  
  
7. \*\*Auto-Close Logic\*\*: Timers were verified for automatic door closing, with checks to extend the delay upon PIR detection or repeat open request.

# Chapter X: Results

The Arduino-based automatic glass door system functioned as intended, fulfilling the design goals. Key results observed include:  
  
- \*\*Accurate Homing\*\*: The door homes precisely at startup, ensuring encoder-based control begins from a known point.  
  
- \*\*Smooth Motion Control\*\*: Encoder feedback combined with speed ramping or PID provided smooth acceleration and deceleration, avoiding sudden jerks.  
  
- \*\*Robust Emergency Control\*\*: The system stops the door when opposing commands are given, preventing hardware damage.  
  
- \*\*Efficient PIR Response\*\*: Human detection was fast and reliable, with successful extension of the auto-close period on detection.  
  
- \*\*Serial Debugging\*\*: Detailed outputs showed encoder values, speed, and trigger events, enabling easy diagnosis.  
  
- \*\*Stability\*\*: The system remained stable under continuous cycles of open and close operations.  
  
These results demonstrate that the integration of sensors, encoder logic, and relay-based control can produce a reliable and safe automatic door solution.

# Chapter X: Limitations and Future Scope

Despite the success of the project, certain limitations were identified:  
  
- \*\*No Obstacle Detection\*\*: The system currently lacks IR or ultrasonic sensors for obstacle detection mid-operation.  
- \*\*Fixed Motion Profiles\*\*: While smooth, the system doesn't adapt its motion dynamically based on weight or resistance.  
- \*\*Single Access Point\*\*: The design supports one door; multi-door integration would need communication modules.  
- \*\*No Security Integration\*\*: Features like biometric or RFID authentication are not included.  
  
\*\*Future Enhancements\*\*:  
  
- Add obstacle detection using ultrasonic sensors to stop the door if an object is detected.  
- Implement adaptive PID control for varying load conditions.  
- Introduce mobile app control or voice command integration.  
- Expand the system to support multiple entry points using wireless communication.  
  
These additions would significantly enhance the robustness, flexibility, and user-friendliness of the system, preparing it for commercial or institutional deployment.