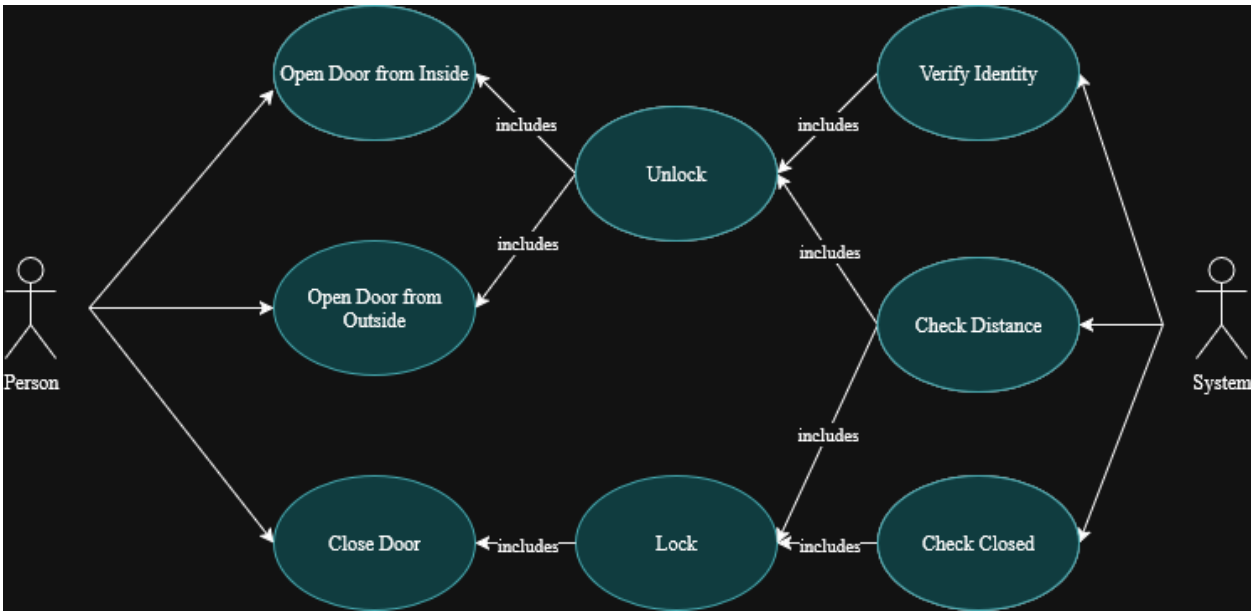
Ryan Antcliff
Dr. Mohammed
CSE 321
30 September 2025

Phase 1: Smart Automatic Door Lock

It is a reoccurring issue that when I leave my apartment in the morning, I forget to lock the front door. Additionally, sometimes I forget my keys and become locked out when my roommates lock the door. A solution to this I wish to propose is an automatic locking device that can be attached to a standard Hadley Village apartment door lock that is able to take care of locking and unlocking the door on its own. The device must be able to sense that it's closed to initiate itself to lock the door. It must also be able to sense a person approaching the door from inside the apartment to unlock it. On the exterior of the door, it needs to be able to detect a person approaching, like the way it does on the inside, but additionally, it would require a camera and facial recognition technology to unlock the door for only authorized persons that wish to enter.

UML Case Diagram



CRC Cards

Class: InsideUnitController	
Responsibility:	**Collaborations:**
• Manage all inside-hardware • Coordinate lock/unlock actions triggered from inside presence or external authorization • Ensure door only unlocks when magnetic sensor confirms closed	• UltrasonicSensorInside • MagneticSensor • ServoMotorLock • BluetoothComm • SafetyManager

Class: OutsideUnitController	
Responsibility:	**Collaborations:**
• Manage all outside-hardware • Run facial recognition logic for access control • Send authorization signal to InsideUnitController	• UltrasonicSensorOutside • CameraModule • FaceRecognition • BluetoothComm • SafetyManager

Class: UltrasonicSensorInside	
Responsibility:	**Collaborations:**
• Detect a person approaching from the inside • Trigger auto-lock/unlock readiness when someone approaches	• InsideUnitController • Scheduler

Class: UltrasonicSensorOutside	
Responsibility:	**Collaborations:**
• Detect a person approaching from the outside • Wake system to prepare for face scan	• OutsideUnitController • Scheduler

Class: MagneticSensor	
Responsibility:	**Collaborations:**
• Detect if door is physically closed or open • Provide safety check before locking	• InsideUnitController • SafetyManager

Class: ServoMotorLock	
Responsibility:	**Collaborations:**
• Physically rotate lock into locked/unlocked position • Obey safe operation (don't engage if door is open)	• InsideUnitController • SafetyManager

Class: CameraModule	
Responsibility:	**Collaborations:**
• Capture facial image/video for recognition • Pass raw data to FaceRecognition	• OutsideUnitController • FaceRecognition

Class: FaceRecognition	
Responsibility:	**Collaborations:**
• Process image data to detect and recognize faces • Match face against stored database of authorized faces • Return result to OutsideUnitController	• CameraModule • OutsideUnitController • DatabaseManager

Class: BluetoothComm

Responsibility:

- Establish and maintain link between inside and outside Arduino units
- Transfer access decisions, lock states, and sensor signals
- Handle error recovery if connection drops

Collaborations:

- InsideUnitController
- OutsideUnitController
- ErrorHandler

Class: DatabaseManager

Responsibility:

- Store list of authorized users/faces
- Provide lookup for FaceRecognition

Collaborations:

- FaceRecognition
- AdminInterface

Class: SafetyManager

Responsibility:

- Verify safety before locking (door closed, servo not obstructed)
- Override system if unsafe condition is detected
- Fail-safe default: keep door unlocked if system error occurs

Collaborations:

- ServoMotorLock
- MagneticSensor
- InsideUnitController
- ErrorHandler

Class: Scheduler

Responsibility:

- Schedule periodic sensor reads (ultrasonic, magnetic)
- Prioritize real-time tasks (face recognition, lock actuation)

Collaborations:

- InsideUnitController
- OutsideUnitController
- UltrasonicSensorInside
- UltrasonicSensorOutside

Class: ErrorHandler

Responsibility:

- Handle system errors (Bluetooth failure, servo jams, face rec. timeout)
- Provide fallback behavior
- Notify user/admin via logs or communication interface

Collaborations:

- BluetoothComm
- ServoMotorLock
- SafetyManager
- AdminInterface

Class: AdminInterface

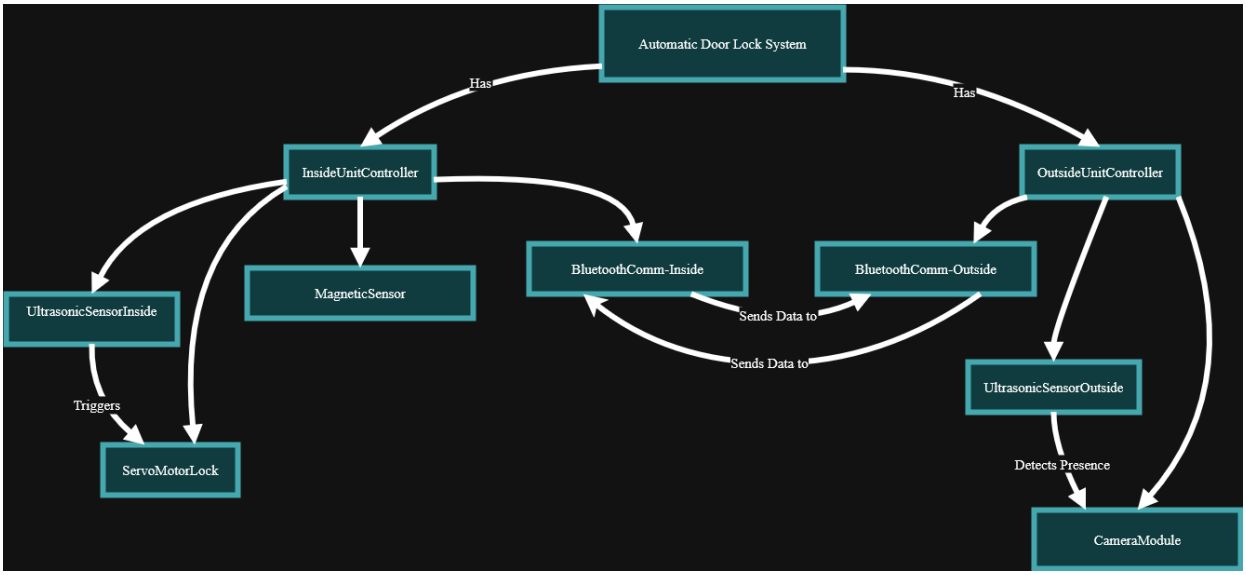
Responsibility:

- Allow admin to add/remove authorized users (faces) via Bluetooth or wired connection
- Provide diagnostics and logs

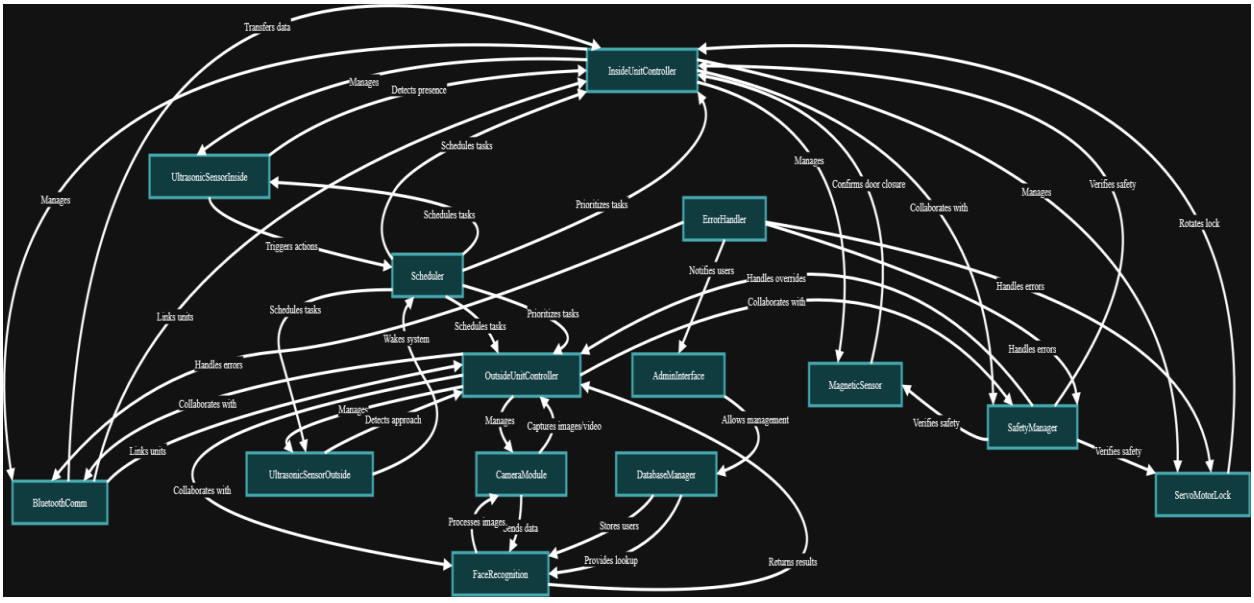
Collaborations:

- DatabaseManager
- BluetoothComm
- ErrorHandler

Architectural Block Diagram



Flow Chart



Component List

Component Name	Item Code	Quantity	Status
Ultrasonic Sensor	HC-SR04	2	On-Hand
Arduino	Uno R3	2	Not Ordered
Bluetooth Module	HC-05	2	Not Ordered
Servo	TBD	1	Not Ordered
Camera (for facial recognition)	Link	1	Not Ordered
Magnetic Contact Sensor	Link	1	Not Ordered