Creating a smart public toilet using Arduino involves several components and steps.

Components Needed:

1.Arduino board (e.g., Arduino Uno or Arduino Mega)

2.Sensors (e.g., ultrasonic sensor, infrared sensor)

3.Solenoid or servo motor for locking/unlocking the door

4.Electronic lock (optional)

5.Water pump and valves for flushing

Temperature and humidity sensors

Display screen (e.g., LCD or LED)

6.Push buttons or capacitive touch sensors for user input

7.Power supply

8.Internet connectivity module (optional for remote monitoring)

9.Enclosure for protection

Procedure:

Design the Toilet Structure:

●Design or choose a public toilet structure that can accommodate the smart features. Ensure it has a lockable door and space for the electronic components.

Install Sensors:

●Place an ultrasonic sensor or infrared sensor near the toilet entrance to detect when a person enters or exits.

●Use temperature and humidity sensors inside the toilet to monitor conditions.

Locking Mechanism:

●Implement a solenoid or servo motor to lock and unlock the door electronically.

●Optionally, use an electronic lock for added security.

Flush system:

●Set up a water pump and valves controlled by the Arduino for flushing.

●Create a user-friendly interface to trigger flushing.

Display and User Interface:

●Add an LCD or LED display to provide information to users (e.g., availability, cleanliness).

●Implement push buttons or capacitive touch sensors for user input (e.g., flushing, requesting cleaning).

Power supply:

●Ensure a stable power supply for the Arduino and other components.

Write Arduino code to:

★Monitor sensor inputs (e.g., occupancy, temperature, humidity).

★Control the locking mechanism.

Manage the flush system.

★Display information on the screen.

★Respond to user input.

★Connect to the internet for remote monitoring (if needed).

Testing and Calibration:

★Test the smart toilet to ensure sensors and components work as expected.

★Calibrate sensors for accurate measurements.

Safety Measures:

★Implement safety features, like emergency door release and automated door unlocking in case of power failure.

Enclosure and Aesthetics:

★Place all components inside a secure, weatherproof enclosure.

★Design the exterior of the toilet for aesthetics and user-friendliness.

Maintenance and Monitoring:

★Establish a maintenance plan for regular checks and cleaning.

★If you added remote monitoring, set up alerts for any issues.

User Education:

★Provide clear instructions to users on how to use the smart toilet.

★Remember that this is a complex project that may require expertise in electronics, programming, and engineering. Ensure that you follow safety regulations and consider privacy concerns when implementing such a system in public spaces.

Arduino software program for a smart public toilet:

```
#include <Servo.h> // Include the Servo library for the door locking mechanism


Servo doorLock; // Create a servo object for the door lock


int doorLockPosition = 0; // Initialize the door lock position (locked)


void setup() {
  // Initialize pins and components
  doorLock.attach(9); // Attach the servo to pin 9
  pinMode(2, INPUT); // Ultrasonic sensor input
  pinMode(3, OUTPUT); // Water pump control
  pinMode(4, OUTPUT); // Flush valve control
  pinMode(5, INPUT); // User input button
  // Initialize other components as needed
}

void loop() {
  // Check for user input
  if (digitalRead(5) == HIGH) {
    // User pressed the flush button
    digitalWrite(4, HIGH); // Activate flush valve
    delay(1000); // Flush for 1 second
    digitalWrite(4, LOW); // Deactivate flush valve
  }

  // Check for occupancy using the ultrasonic sensor
  if (digitalRead(2) == HIGH) {
    // Toilet is occupied
    // Lock the door
    doorLock.write(doorLockPosition); // Lock the door
  } else {
    // Toilet is unoccupied
    // Unlock the door
    doorLock.write(90); // Unlock the door (assuming 90)
```

Future Enhancement:

- Integration with internet connectivity for remote monitoring and maintenance.
- Data collection and analytics for usage patterns and maintenance needs.
- Enhanced user experience with touchless controls, voice commands, or mobile apps.

Conclusion:

◆Safety, security, and privacy are critical considerations when designing and implementing a public toilet system.

◆Compliance with local regulations and accessibility standards is essential.

◆Regular maintenance and monitoring are needed to ensure the system functions reliably.