Project Proposal

Title: Simulated Sensor Placement Optimization in Smart Buildings with Context-Aware Access and Control
Application Name: SensorPlacement
Group: 6
Group Members: David Abiola - 100974509, Chao Meng - 100997810, Suhas Sunder - 100548159
Topic from list of projects proposed by professor: Context-aware services and Contextual Access Control

1. **Objective Summary**
   This smart building project is a sensor placement simulation tool that will be hosted live and provides interaction with virtual smart devices within a smart building. Users will be able to select floors within a smart building, which will reveal a 2D top-down layout of the entire floor. Once the layout is present, users can add/drag/remove sensors into the layout to decide optimal placement and ensure proper connectivity and data transmission between devices overlapping with the sensor's coverage area. This would not only reduce costs by eliminating unnecessary sensors or selecting appropriate coverage ranges, but also help reduce unnecessary overlap, signal interference, and other issues related to the mobility management of wireless devices.

   It will be an interactive web application connected to a database while performing context-aware operations and device control. It leverages context data such as location, time, and smart home sensor readings to make intelligent decisions for smart home applications within simulated rooms.

2. **Motivation**
   With the rapid growth of Internet of Things (IoT) and smart home technologies, managing and controlling multiple interconnected devices in a building has become increasingly complex. Real-world deployment and testing of smart building systems can be costly, time-consuming, and limited by hardware availability. This system provides a flexible platform for experimentation and learning without requiring physical hardware. Additionally, the simulation facilitates understanding of pervasive computing principles, enabling users to explore intelligent automation and context-driven decision-making in a controlled, cost-effective manner. Having read several well written and reputable research papers highly relevant to this topic, it makes perfect sense to pursue this idea and potentially come up with improvements and contributions that are in line with furthering this research.

3. **Key Features**
   a. **Simulated Smart Devices:**
      i. Virtual representations of common smart building devices such as thermostats, lights, security cameras, door locks, motion sensors, and smoke detectors.
      ii. Devices will mimic realistic behaviors and support state changes (e.g., on/off, locked/unlocked, temperature adjustments).
      iii. Sensors can be added/removed and dragged around to determine best placement within a given floor layout.
   b. **Context-Aware Automation:**
      i. Devices dynamically react to contextual information, including:
         1. Location: Device or user presence in specific rooms.
         2. Time: Scheduled events and time-based triggers.
         3. Sensor readings: Temperature, ambient light levels, occupancy status.
      ii. Sensor placement can be adjusted to optimize coverage and save costs by reducing redundancy.

c. **Real-Time Device Status Updates:**
    i. Real-time UI updates to reflect device state changes, such as lights toggling on/off, doors locking/unlocking, and temperature changes.
d. **Data Logging & Analytics**
    i. Persistent recording of device state changes and sensor data over time.
    ii. Analytics dashboard to display trends and usage patterns (**If time permits**)
e. **Open-Access Usability:**
    The simulation tool requires no authentication, allowing anyone to experiment with smart building contexts and device behaviors freely.
f. Hosted live on the web for users to access and perform simulations.

4. Tools & Frameworks

a. Front-end: React & TailwindCSS
b. Back-end: Flask (Python) & SQL/NoSQL Database
c. Hosting: Digital Ocean & Cloudflare CDN

5. Testing

a. Functional Testing:
    i. Verify that simulated devices behave as expected (e.g., lights toggle, doors lock/unlock, thermostat adjusts temperature).
    ii. Confirm that context-aware automation triggers device actions correctly based on location, time, and sensor inputs.
b. Usability Testing:
    i. Gather user feedback on the interface ease-of-use and clarity of device state representations.

6. Deliverables

a. System Architecture and Design Documentation
    i. Diagrams and explanations covering the system modules, data flow, and interaction between frontend and backend components.
b. Functional Prototype
    i. Interactive web application simulating smart devices and context-aware automation.
    ii. Backend implementation handling context inference, device state management, and notifications.
c. User Interface
    i. A responsive web UI allowing users to simulate context (location, time, sensor inputs) and control devices, with real-time updates.
d. Data Logging and Analytics Module
    i. Implementation of logging device states and sensor data with a basic dashboard for usage trends (if feasible within timeline).
e. Testing and Evaluation Report
    i. Documentation of correctness tests, response time measurements, and any user testing feedback.
f. Final Presentation / Demonstration
    i. Clear demonstration of functionality and testing of the system.
g. Final Report
    i. Comprehensive report including methodology, implementation details, challenges faced, evaluation results, and potential future work.