POCKET GUARD

IoT-ready, Low-cost Personal Belongings Monitoring

Daniel Fernando Pigatto
Christian Becker Pepino
Guilherme Dias
Federal Technological University - Paraná
MONITORING SYSTEMS
WHAT IS POCKET GUARD?

- A portable monitoring module
- Can be coupled to different objects
- "Micromonitoring"
- GSM, Bluetooth and GNSS
- Mobile App
USE CASES

- Purses, backpacks, bags
- Cars, trucks
- Lockers
- Personal belongings in general
- Reduce theft and make the user feel confident
HOW DOES IT WORK?

Configure and activate tracking module

Put the tracker on/in object of interest

Suspicious movement Detection

Notification + Global Position Tracking
CONFIGURATION

Pocket Guard

Username

Password

Sign in

Not registered yet?

Welcome back Christian, things look alright.

backpack

distance: 19 km
alarm: on

last updated: 45s ago

+

Looking for trackers

Pair

Cancel

pair manually
POSITIONING
SUSPICIOUS MOVEMENT

- Owner’s absence (Bluetooth)
- Movement (Accelerometer)
LOCATION TRACKING

Device details

backpack

alarm: on  distance: 19 km

last update: 57s ago

Deactivate alarm

Remove Device
SYSTEM OVERVIEW

Web Server

Pocket Guard Device

Pocket Guard App

Apple Push Notification Service

Internet

Bluetooth
WEB SERVER

- Centerpiece of the system
- Handles HTTPS requests
- Stores and retrieves user and device data from database
- Encapsulates notifications rules
- Handles authentication (token)
WEB SERVER - Technologies

- High performance
- Ease of use
- NPM Package Manager
- Scalability

- Cloud hosting service
- Supports many programming languages
- Custom command line Tool
WEBSEVER - Architecture

- RESTful
- HTTP Methods: GET, POST, PUT, DELETE.
- HTTP Codes (404, 201, 500)
- JSON

Source: http://phpflow.com
WEBSERVER - Architecture

- GET /devices handler
- JSON responses
HARDWARE: ARDUINO UNO

- Popular, low-cost, open-source, widely available
- Low specs (8-bit Processor, 16 MHz Clock, 2 KB Ram, 1 KB EEPROM)
- Available modules (Bluetooth 4.0, GSM, GPS, Accelerometer)
TRACKER STATE MACHINE

- **Proximity**
  - bluetooth connected
  - already paired
  - not paired

- **Active**
  - moved suspiciously
  - alarm untriggered

- **Triggered**
  - paired

- **Pairing**
  - bluetooth disconnected
typedef enum {active, pairing, triggered, proximity} state;
state deviceState;

switch (deviceState) {
  case active:
    deviceActive();
    if (alarmTriggered && alarmOn){
      deviceState = triggered;
      prepareForTriggered();
    }
    if (bluetoothConnected) {
      deviceState = proximity;
      prepareForProximity();
    }
    if (alreadyPaired == false) {
      deviceState = pairing;
      prepareForPairing();
    }
    break;
  case pairing:
    devicePairing();
    if (alreadyPaired == true) {
      leavePairing();
      deviceState = active;
      prepareForActive();
    }
    break;
DIFFICULTIES

- Fake components
- Limited RAM
- Faulty wiring

Condemn the copycat company copied behavior on HM-10!!!!!!
If you buy a fake, please apply for a refund guarantee your legitimate rights and interests
FUTURE WORK

- 3D Printed Case
- Reduced size
- Reduced Latencies
- Android App
- Authorized contacts
- Battery lifetime statistical validation
QUESTIONS