Smart Parking System Architecture

Vom avea 3 module:

* Un modul va genera date – vom folosi un Raspberry Pi pt a transmite si a afisa imagini preluate de o camera pe un IP local (Raspberry Pi Camera);
* Un modul va fi dashboard-ul – prin care vom afisa datele (locuri de parcare disponibile, locuri ocupate, preturi, etc) – android studio maybe?!
* Un modul fi worker-ul – va implementa logica care nu afecteaza UI-ul (dashboard-ul) – actualizari de locuri disponibile, ori de cate ori o masina este parcata (**event**), background worker-ul va scadea nr de locuri disponibile (background worker-ul va fi declansat de un eveniment) - <https://www.wpf-tutorial.com/misc/multi-threading-with-the-backgroundworker/>

Configurare Raspberry Pi:

Instalare sist de operare Raspbian.

Se seteaza Raspberry PI astfel incat sa accepte modulul camera;

Se ruleaza urmatoarele comenzi pt transmiterea de imagini capturate de camera pe un cont de Azure:

<https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-raspberry-pi-kit-node-get-started>

Comenzi pt transmiterea de imagini inspre un IP local:

1. sudo apt-get install motion – Această comandă va instala software-ul necesar transmisiunii live.
2. sudo nano /etc/motion/motion.conf – Această comandă deschide fișierul de configurare al softwareului unde trebuie schimbați anumiți parametri:
   1. daemon - trebuie setat pe ON
   2. framerate – trebuie setată o valoare cât mai mare (ex: 100)
   3. Stream\_port – alegeți port-ul pe care se vor transmite imaginile (8081 e prestabilit)
   4. Stream\_qualty – la fel ca la framerate, o valoare cât mai mare (ex: 100)
   5. Stream\_localhost – trebuie setat pe OFF
   6. webcontrol\_localhost – trebuie setat pe OFF
   7. quality – setat pe 100
   8. width – setat pe 640
   9. height – setat pe 480
   10. post\_capture – setat pe 5
3. sudo nano /etc/default.motion – În fișierul deschis de această comandă trebuie doar setat parametrul „start\_motion\_daemon” pe „yes”.
4. sudo service motion start – Comanda pentru pornirea transmisiei live
5. În browser, la adresa 192.168.1.4:8081 se pot vedea imaginile provenite de la Raspberry Pi. Adresa menționată diferă între modelele de Raspberry Pi și conexiunea la internet.

Constrangeri:

* Locuri de parcare pt oamenii cu nevoi speciale.
* Tipuri de masini ce trebuie parcate – camioane, masini obisnuite etc

Ideea principala:

Raspberry PI preia o imagine, determina cate locuri sunt disponibile si transmite info care va fi afisata in aplicatia mobila.

Basic dashboard:

