

# Okos önkiszolgáló kassza

## Feladat leírása:

A kamera látóterében található, valamilyen módon előre definiált termékek felismerése, független a termék által felvett póztól. Kasszáknál történő kiszolgálásnál különböző előre csomagolt termékeket vásárolunk. A termékeket QR-kóddal látják el amivel a termék beazonosítható. A beadandó keretében nem QR-kód alapján azonosítja be a gép a termékeket hanem egy kamera képe alapján. Ez a módszer hatékony kiegészítő módszer lehet ha a terméken lévő QR-kód megsérül, a nem valós vagy a beolvasó nem működik. Program feladata egy képi forrás alapján eldönteni hogy milyen termékek találhatóak rajta. Minimum 6 fajta termék felismerése a cél.

## Elméleti háttér:

## Megvalósítás:

Programozási környezetnek JavaScript-et választottam nagy kompatibilitási és hordozhatósági képessége miatt. A program "Single Page" (egylapos) applikáció lesz.

## Fejlesztői dokumentáció:

### Kosár függvények:

- `setProduct(name, count, price, unit)`: Termék beállítása a kosárban
  - `name` : string - A termék neve
  - `count` : integer - a termék darabszáma
  - `price` : integer - a termék ára (opcionális)
  - `unit` : string - egység (db, kg stb.) (opcionális)
  - Visszatérési érték: Ha sikerült a végrehajtás akkor igaz különben hamis
- `addProduct(name, count)` : Termékszám módosítása a kosárba
  - `name` : string - A termék neve
  - `count` : integer - a hozzáadandó darabszám
  - Visszatérési érték: Ha sikerült a végrehajtás akkor igaz különben hamis

### Videó függvények:

- `VIDEO`: videó HTML objektum elérés
- `async listVideo()`: Kilistázza a videóforrásokat

- Visszatérési érték: Promise objektumot ad vissza, utána igaz ha sikeres a listázás különben hamis
- `async setVideo(index)`: beállítja a megadott videó forrást, listázás után elérhető
  - `index` : integer - videó forrás indexe
  - Visszatérési érték: Promise objektumot ad vissza, utána igaz ha sikeres a beállítás különben hamis
- `async setVideoFile()`: betölt egy kiválasztott videó fájlt és beállítja forrásként.
  - Visszatérési érték: Promise objektumot ad vissza, utána igaz ha végzett a metódus
- `removeVideo()`: eltávolítja az aktuális videóforrást és felszabadítja a lefoglalt memóriát
- `async startVideo()`: Videó első indítása (inicializálása), ellenőrzi a támogatást, betölti a függőségeket, kilistázza a videókat és a legutolsó forrásra állítja
  - Visszatérési érték: Promise objektumot ad vissza, igaz ha sikeres a betöltés különben hamis
- `drawCanvas(name, x, y, width, height)`: Kirajzol egy dobozt az objektum nevével
  - `name`: string - Objektum neve
  - `x`:integer - X koordináta (bal felső)
  - `y`: integer - Y koordináta (bal felső)
  - `width`: integer - objektum szélessége
  - `height`: integer - objektum magassága
- `clearCanvas()`: törli az összes kijelzett objektumot

#### Objektum detektálás:

- `async loadDetection()`: tárgyfelismerés betöltése
  - Visszatérési érték: Promise objektumot ad vissza, igaz ha sikeres a betöltés különben hamis
- `startDetection()`: detektálás elindítása
- `stopDetection()`: detektálás leállítása

## Tesztelés:

Videó forrás kiválasztása után betölthetővé válik egy előre felvett videó.

## Felhasználói leírás:

A program betöltése után a felhasználó a kamera képét és mellette vagy alatta a kosár tartalmát láthatja. Kamera képe alatt egy legördülő menüből lehet kiválasztani a program által fogadott kép forrását.

## Irodalomjegyzék: