Technisch verslag - AR Maaltijd checker

## Software versies

Unity: 2019.1.1f1

Vuforia: 8.1.7

Android: 4.1 ‘Jelly Bean’ (API level 16)

## Consumptie waarden web scrape

De consumptie waarden worden van een website af gescraped. [https://www.voedingswaardetabel.nl](https://www.voedingswaardetabel.nl/)

Als bepaalde consumptiewaarden opgevraagd moeten worden, wordt eerst de html van de pagina van de consumptie opgevraagd. Deze url komt overeen met de pagina van cola light: <https://www.voedingswaardetabel.nl/voedingswaarde/voedingsmiddel/?id=1025>

Zoals te zien is, hebben verschillende voedingen een id. Door het id te kennen kunnen de voedingswaarden worden opgevraagd.

Door vervolgens de html string op te splitten zijn de waarden uit de html string te halen. Hier een voorbeeld voor Kcal:

double GetKcal(string site)

{

string[] spited = site.Split(new string[] { @"<span id=""lblKcal"">" }, StringSplitOptions.None);

string[] value = spited[1].Split('<');

return Convert.ToDouble(value[0]);

}

## Voedingswaarden scoreboard

## Afbeelding met lucht Automatisch gegenereerde beschrijving

Het voedingswaarden scoreboard wordt aan de consumptie geplakt. De gele balken geven aan dat de hoeveelheid van de voedingsgroepen 50 tot 99% zijn voldaan. Als er onder de 50% is voldaan aan de voedingswaarden, dan zijn de balken groen. Afbeelding met lucht

Automatisch gegenereerde beschrijving

(In de foto hierboven werden de voedingswaarden van normale cola gekoppeld aan het object.)

Dit wordt gedaan door alle waarden uit de maaltijdlijst bij elkaar op te tellen.

De blauwe balken geven aan hoeveel de gescande consumptie toevoegt aan de huidig totaal genuttigde voeding. Dit is uiteraard ten opzichte van de vooraf aangegeven maximale fat (vet), proteïne, koolhydraten en calorieën.

De grote van de balken worden berekend met de volgende code:

float calc\_fat = (min\_balk\_grote + (max\_balk\_range \* (float)prcentvodaan\_fat));

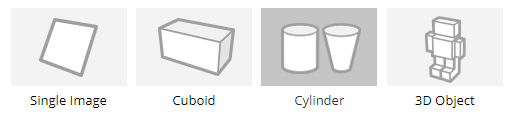
if (calc\_fat < max\_balk\_grote)

{

fatbar.transform.localScale = new Vector3(calc\_fat, fatbar.transform.localScale.y, fatbar.transform.localScale.z);

}

## Vuforia Object detectie



In de AR applicatie is gekozen voor de cilinder objecten. Dit is voor deze demo applicatie binnen de scope omdat voor 3D objecten een goed belichte setup gemaakt moet worden. Dit is wel de correcte manier om consumpties in te scannen omdat deze in alle vormen en maten komen.

Voor cola werken de cilinder objecten ook:

Afbeelding met elektronica

Automatisch gegenereerde beschrijving

Deze object database kan direct in unity worden gedownload.

Afbeelding met schermafbeelding

Automatisch gegenereerde beschrijving

Het Gameobject binnen unity.

Afbeelding met lucht

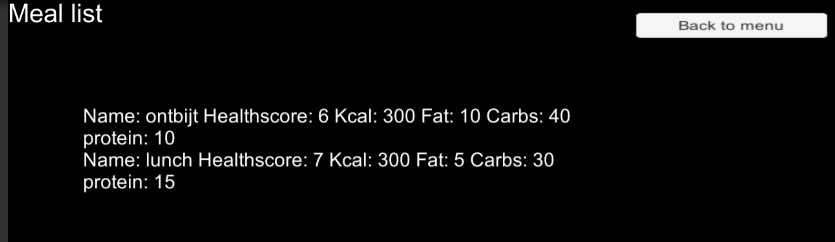
Automatisch gegenereerde beschrijving

## Menu Control

Afbeelding met schermafbeelding

Automatisch gegenereerde beschrijving

Het menu binnen de applicatie wordt door canvas objecten geregeld. Door voor elk submenu een canvas object te maken kan gemakkelijk geswitcht worden tussen verschillende menu’s.



Door aan een knop een methode toe te voegen, is er makkelijk gedrag te koppelen aan een knop.

Afbeelding met schermafbeelding

Automatisch gegenereerde beschrijving

De functie:

public void with\_to\_main\_menu() {

main\_menu.SetActive(true);

add\_meal\_menu.SetActive(false);

meal\_list\_menu.SetActive(false);

onscreen\_menu.SetActive(false);

}

## Consumptie container

De maaltijden worden opgeslagen in de klasse logic, dit is een container voor maaltijden. De meal list variable is een lijst van alle maaltijden.

public List<meal> meal\_list = new List<meal>();

Een maaltijd is opgebouwd uit een struct, deze struct bevat allen relevante data.

public struct meal

{

public string name;

public double kcal;

public double healtyscore;

public double fat;

public double carb;

public double prot;

public meal(string n, double k, double h, double f, double c, double p)

{

name = n;

kcal = k;

healtyscore = h;

fat = f;

carb = c;

prot = p;

}

}

## Final product



De applicatie herkend een blikje Fanta en geeft aan dat het de maximale aantal koolhydraten zou overschrijden na het nuttigen van deze consumptie.

Het Unity project is te vinden op <https://github.com/Slimmerik/ARD>.