**Technisch Ontwerp**

**Vending Machine**

datum: 12-dec-2018

Versie: V0

door: Maarten Jakobs & Max van den Boom

klas: IC16AO.b

|  |  |  |
| --- | --- | --- |
| Naam: DOCENTNAAM | Naam: KLANT | Naam: AUTEUR (Student) |
| Functie: | Functie: | Functie: |
| Datum: | Datum: | Datum: |
| Goedgekeurd (handtekening) | Goedgekeurd (handtekening)  De klant kan dezelfde persoon zijn als de docent. | Goedgekeurd (handtekening) |

**Goedkeuring**

Inhoud

[Inleiding 3](#_Toc532386096)

[Eisen 3](#_Toc532386097)

[Beslissingen 3](#_Toc532386098)

[Afspraken 3](#_Toc532386099)

[Plan van Aanpak 4](#_Toc532386100)

[Ontwerponderdelen 4](#_Toc532386101)

[Web server aanmaken 4](#_Toc532386102)

[Website 4](#_Toc532386103)

[Ontwerponderdelen datahandling 4](#_Toc532386104)

[Basisopbouw dataverzending 4](#_Toc532386105)

[Stap 1 Website 4](#_Toc532386106)

[Stap 2: QR code generator 4](#_Toc532386107)

[Stap 3: QR code reader 5](#_Toc532386108)

[Stap 4: Splits byte-array in RGB 6](#_Toc532386109)

[Stap 5: patronen 7](#_Toc532386110)

[Over dit document 8](#_Toc532386111)

[Afkortingen 8](#_Toc532386112)

[Referenties 8](#_Toc532386113)

[Definities 8](#_Toc532386114)

[Gebruikte materialen 8](#_Toc532386115)

[Bijlagen 9](#_Toc532386116)

[Ascii tabel 9](#_Toc532386117)

[Arduino MEGA 2560 pinout 10](#_Toc532386118)

# Inleiding

Dit technisch ontwerp is gemaakt voor de opdracht ‘Vending machine’. Door het lezen van dit bestand moet het duidelik worden wat het product is en hoe het werk.

# Eisen

de vending machine gaat werken met een website. Je kan op de website een lijstje maken van producten die je uit de vending machine wilt halen. Deze producten kun je dan meteen afrekenen op dezelfde website. De vending machine heeft een qr code scanner waar de klant heen kan gaan wanneer hij de bestelling heeft afgerond en een speciale qr code genereerd is.   
de vending machine bestaat uit 2 lossen applicatie’s 1 voor de webkant en 1 voor de arduino kant. De programeer talen die we voor de webkant gebruiken zijn html, sql, css, javascript, php en phpmyadmin 4.8.3. de progameer taal voor de arduino kant is C. We gebruiken voor de webkant Sublime Text 3 en voor de arduino kant gebruiken we de arduino IDE.

# Beslissingen

We moesten meerdere beslissingen maken voor de werking van ons product

* Kinect of webcam. We hebben voor de kinect gekozen omdat deze een vele hogere resolutie heeft en een snellere manier van datahanding. Het voordeel ook van de kinect is dat er al heel veel programmas voor gemaakt zijn die wij dus makkelijk kunnen implemteren.
* Website voor telefoon of voor pc. We hebben de keus gemaakt om de website voor de telefoon te maken zodat mensen makkelijk de bestellingen on the go kunnen maken. En dan ook gemakkelijk de qr code gebruiken.
* Traditionele barcode of qr code. We hebben voor een qr code gekozen omdat in een qr code veel meer data verwerkt kan worden. Daarnaast ook dat een qr code genereren redelijk makkelijk is.

# Afspraken

We hebben het project in 2 delen verdeeld het Arduino gedeelte en de web kant. De webkant wordt gedaan door Maarten en de arduino kant wordt gedaan door Max. Verder wordt de documentatie samen gedaan. Het hardware gedeelte van de kast wordt gedaan door Max. Zo is de verdeling van het project ongeveer hetzelfde.  
we hebben ook gekozen voor een versie beheer we gebruiken daar github voor, om het ons makkelijk te maken gebruiken we daar Git kraken voor.

# Plan van Aanpak

De vending machine wordt bestuurd vanuit een webiste maar de werkelijke hardware (de vending machine zelf) wordt bestuurd door een Arduino. Omdat er meer outputs nodig zijn dat de arduino uno aankan is er gekozen om een arduino MEGA2560 te gebruiken. De arduino comminuceert via een PC die via een website informatie doorstuurd. Er bestaan dus verschillende punten in de totale applicatie die als Showstopper geindentificeerd kunnen worden. In het volgende hoofdstuk staan de stappen die ondernomen moeten worden om er zeker van te zijn dat de totale applicatie een kans van slagen heeft.

de stappen worden in het hoofdstuk Ontwerponderdelen verwerkt.

# Ontwerponderdelen

## Web server aanmaken

We maken een webserver in de arduino. Deze wordt gemaakt via een Ethernet library van arduino. Deze stuurt en onvangt jsons strings van en naar de website.

## Website

De website wordt gehost op de vps van Max. Deze heeft wordt gebruikt door de users door naar myvending.amxdev.nl de vending pc zelf heeft de subdomein naam vendingpc.amxdev.nl

# Ontwerponderdelen datahandling

## Basisopbouw dataverzending

Website > QR code generator > QR code reader > Arduino

## Stap 1 Website

De webiste is de plaats waar de gebruiker een bestelling kan maken. Hier kan hij items toevoegen in het winkelmandje die hij dan ook meteen kan betalen.  
het betalen gaat via een credit systeem dit credit systeem kan je opwaarderen via paypal.

## Stap 2: QR code generator

Om de prodcuten op te halen bij het apparaat creeert het apparaat een QR code. Deze QR code wordt gegenereerd door middel van een functie die werkt via een library. Deze functie geef je een string aan die hij daarna verandert naar een Qr code.

## Stap 3: QR code reader

Om de QR code te lezen wordt er een kinect gebruikt. Deze kinect

## Stap 4: Splits byte-array in RGB

*Werking*:  
Splits byteArray met RGB colorData in 3 aparte arrays (voor elke kleur 1), status van de kleur in het byte  
  
opbouw byte: xxxx x###, x heeft geen waarde of betekenis, # is de kleurstatus

Voorbeeld

* voor segment 1, led 0, kleur rood
* waarde ASCII is 49

*Declaratie*:  
in Arduino voor de kolommen, bij een Common Anode LED type wordt de kathode ‘ge-sync-ed’, ofwel de kathode wordt aan de GND gelegd.

int doSyncLedRowRed[8];  
int doSyncLedRowGreen[8];  
int doSyncLedRowBlue[8];  
int ledValueInAscii;

*Handling*:  
onderstaande zijn vast geprogrammeerde waarden, in werkelijkheid moet er een for loop worden gebruikt.

Voorbeeld voor colorData 3

ledValueInAscii = (int)m\_data[3] – 48;   
//result for colorData 0 is: 1

doSyncLedRowRed[0] = ledValueInAscii AND 0x01;  
doSyncLedRowGreen[0] = ledValueInAscii AND 0x02;  
doSyncLedRowBlue[0] = ledValueInAscii AND 0x04;  
//result for LED 0 is: red is on, green is off, blue is off

Voorbeeld voor colorData 6

ledValueInAscii = (string)m\_data[6] – 48;   
//result for colorData 6 is: 7

doSyncLedRowRed[4] = ledValueInAscii AND 0x01;  
doSyncLedRowGreen[4] = ledValueInAscii AND 0x02;  
doSyncLedRowBlue[4] = ledValueInAscii AND 0x04;  
//result for LED 0 is: red is on, green is on, blue is on

## Stap 5: patronen

Rode tekst: kernwoorden en ideeën

Multi dimensionaal array

Element 1: RGB waarden

Element 2: patroonstap

Voorbeeld: doSyncLedRowRed[regelWaardenIndex][patroonNummerIndex]

Implementatie:

* Arduino [][]
* C# [,]

Afhandeling in Arduino

* In de loop loopt een patroonTicker, bij elke keer dat de loop doorlopen wordt gaat er 1 af, tot bij de ingestelde waarde, dan weer op de default
* Met delays zou het hele systeem vertragen

Afhandeling in C#

* Het tweede gestuurde karakter is het patroon nummer

# Over dit document

## Afkortingen

|  |  |
| --- | --- |
| Afkorting | Omschrijving |
| WBS | WorkbreakDownStucture |
|  |  |

## Referenties

|  |  |
| --- | --- |
| Referentie | Omschrijving |
|  | Voorbeelden: ‘bovenliggende’ documenten (ProjectPlan, Functioneel Ontwerp, etc.) |
|  |  |

## Definities

|  |  |
| --- | --- |
| Definitie | Omschrijving |
| Showstopper | Technisch onderdeel van een ontwerp dat er voor kan zorgen dat de totale haalbaar heid van het project in gevaar komt. Het is belangrijk dat de eerst de showstoppers geïdentificeerd worden en getest worden.  Voorbeeld: als een apparaat bestuurd moet worden vanuit een PC maar de communicatie werkt niet tussen PC en apparaat, zal het dus nooit vanuit een PC bestuurd kunnen worden. Oplossing: eerst dit testen, dan pas naar de volgende stap. |
|  |  |

## Gebruikte materialen

|  |  |
| --- | --- |
| Onderdeel | Omschrijving |
|  | Voorbeelden: hardware (elektronica, arduino, PC-type), software (IDE, DLL’s) |
|  |  |

# Bijlagen

## Ascii tabel



## Arduino MEGA 2560 pinout

