Blikjes Automaat

produktdocument



Student: koen lenting

621852

Vak: Inleiding Software Engineering, D-B-INSE-O

Docent: Remko Welling

17-1-2019

**Documenthistorie**

|  |  |  |  |
| --- | --- | --- | --- |
| **Datum** | **Versie** | **Wie** | **Veranderingen** |
| 3-12-18 | 1 | ik | Productdocumentatie opbouw |
| 6-1-19 | 2 | Ik | Refill and ammount |
| 8-1-19 | 3 | Ik | Toevoegen order systeem |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

* **Samenvatting**

voor dit project moeten wij een eigen code schrijven. de code moet laten zien wat je allemaal in de vorige periode hebt geleerd. opgestelde eisen van de docent zijn:

gebruik van doxygen documentatie, een state machine in je code en dat die werkt

verdere eisen van het project zijn: het testen van de code, fouten beschrijven en oplossen en een productdocument.

* **Voorwoord**

ik vond het een zeer leuke opdracht, je kon echt laten zien dat je niet alleen maar commands kon onthouden maar ook echt toepassen. verder vond ik het wel een lastig project. vooral omdat ik het programeren mezelf heb aangeleerd moest ik vaak aanpassingen doen om de juiste stijl te creëren. ik had namelijk in mijn eigen onderzoek niks gehoord van states.

**Inhoudsopgave**

* **Inleiding**

ik begin met het vertellen wat de machine allemaal moet kunnen in de defenitiefase.

daarna vertel ik het doel van de code.

ik de funtionele eisen zal ik wat vertellen over wat het apparaat allemaal moet kunnen.

technische eisen zal meer vertellen over wat allemaal noodzakelijk is voor de code.

het ontwerp zal laten zien hoe het allemaal in elkaar zit.

en als laatste het eindresulaat die een cunclusie geeft.

* **Definitiefase**

De eisen van mijn drinkmachine zijn: Het bijvullen van de machine, blikjes automatisch verdelen, alle blikjes accepteren en niet een selectie, je moet de voorraad kunnen zijn per rij en per product soort, je moet er een blikje uit kunnen halen, het blikje betalen en de machine uitzetten

* **Kennismaking en doel van de drinkmachine**

Het apparaat geeft na een input van de gebruiker een blikje of laat de producthoeveelheid zien. Het apparaat kan ook blikjes ontvangen de naam is volledig vrij in te vullen door de gebruiker.

* **Functionele eisen**

De functionele eisen zijn als volgt:

* Een blikje geven na betaling
* Hoeveelheid bekijken
* Bijvullen
* Betalen
* stop zetten
* **Technische eisen**

De technische eisen zijn als volgt:

* De programmeertaal C moet worden gebruikt.
* De ontwikkelomgeving QtCreator moet worden toegepast.
* Hij moet sommige typfouten weigeren zoals te veel blikjes vullen of verkeerde locatie
* Hij moet de blikjes netjes verdelen over de hoeveelheid locaties die aangeven zijn
* **Schets user interface**

9 plekken in een 3x3 formfactor. onder elke plek de prijs. en daar onder het menu

* **Ontwerpen**

het ontwerpen van de drinkmachine begon heel simpel, met de 3 keuzes vullen hoeveelheid en bestellen. daarna ben ik de code gaan schrijven voor die 3. al snel merkte ik dat ik voor een blikje een hoop verschillende informatie nodig had een double voor de prijs een int voor de locatie een char voor de naam en dat moest allemaal goed bij elkaar blijven waardoor ik locatie.h ben gaan schrijven waar alle informatie per slot in een apparte struct staat

* **Architectuur**

<overzicht subsystemen in lijstvorm>

Input = keyboard display

Subsystem= blikje dispenser, geldchecker. Bijvuller,

verwijderen van legen locaties

* **State chart**



* **Realisate en testen**

Testen van alles.

Alles ging goed na elke aanpassing.

Het enige waar ik moeite mee had was het aanpassen naar de juiste stijl.

en de struct verplaatsen omdat regelmatig random nummer gaven nadat ik ze in meerde functies had gebruikt. verder waaren een aantal spelfouten de boosdoeners

* **Acceptatietesten**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test** | **Action/input** | **Expected result** | **Pass Fail** | **Actual result if test has failed** |
| 1. | main | Works  Start scherm doet het | yes |  |
| 2. | Order | Works  Geeft blikje en haalt er een uit het systeem | Yes |  |
| 3. | Start | Geef de start data | Yes |  |
| 4. | Display | Laat zien water in het apparaat zit | Yes |  |
| 5. | Struts verplaatsen | Alles staat in een losse .c file | No | Hij geeft random informatie |
| 6. | Structs verplaatsen | Struct in losse .c files | Yes |  |
|  |  |  |  |  |

Alles ging prima na aanpassingen

* **Eindresultaat en aanbevelingen**

hierin beschrijf ik het eindresultaat en de aanbevelingen en verbeteringen

* **Eindresultaat**

het programma werkt volledig en doet wat die moet doen.

* **Aanbevelingen**

het enige wat ik beter zou willen hebben in mijn code is mijn state machine