Pmst1T

Realisatie Document

Never Board

Contents

[Software 2](#_Toc517160472)

[Transities 2](#_Toc517160473)

[UI Componenten 2](#_Toc517160474)

[Schaakspel 3](#_Toc517160475)

[Media 4](#_Toc517160476)

[FUN 5](#_Toc517160477)

# Technology

|  |  |  |
| --- | --- | --- |
| **Gewenst gedrag** | **Werkelijk gedrag** | **Oplossing** |
| De positie van de steppenmotoren moet in Visual studio kunnen worden aangepast | In Visual Studio kun je de stappenmotoren een bepaalde afstand laten draaien of laten doorrijden zonder te stoppen op een bepaalde snelheid | Er is hiervoor geen oplossing nodig |
| Er moeten verschillende motoren tegelijkertijd met 1 Arduino bestuurd kunnen worden | Als je verschillende motorshields met daaraan verbonden stappenmotoren koppelt aan de Arduino bewegen alle aangesloten motoren synchroon, het is op dit moment niet mogelijk om de motoren verschillende commando’s te geven | Door female-to-male jumper kabels aan te sluiten op de bovenkant van bepaalde pins op de eerste motorshield en die te verbinden aan de onder kant van de andere 2 motorshields lukt het om met 12 digitale pins, waarvan 6 PWM pins zijn en 6 reguliere, om 3 verschillende stappenmotoren te besturen. |
| Het moet mogelijk zijn om de stappenmotoren een exacte positie te geven | De stappen van de motoren worden niet bijgehouden | In de code gezet dat de totale stappen van een motor onthouden moeten worden, alsook het beoogde doel in positie, vanwaaruit de Arduino berekent welke kant de motor op moet draaien om dat doel te bereiken |
| De Arduino moet verschillende motoren aan kunnen sturen omdat het bord minimaal 3 motoren nodig heeft | Alle L298P motorshields gebruiks de pins 4,6,7,5 ( 2 PWM, 2 niet-PWM ) om de motoren aan te sturen. Het is op dit moment niet mogelijk om verschillende motoren verschillende instructies te geven omdat ze allemaal dezelfde pins gebruiken | Door male-to-female jumper wires te verbinden tussen de bovenkant van de eerste motorshield en de onderkant van de tweede en derde motorshields is het gelukt om signalen uit andere digitale pins naar de pins 4,6,7,5 te sturen van de desbetreffende motorshield. Zo ziet dat er ongeveer uit in de code:  Stepper X(4, 6, 7, 5);  Stepper Y(8, 9, 12, 11);  Stepper Z(13, 10, 2, 3);  (8 en 13 staan op 4, 9 en 10 op 6, etc.)  Hierdoor kunnen alle 3 de motoren individueel bestuurd worden |
| Project Hades moet via wifi bestuurt kunnen worden | Dan moet er wel eerst een wifi module zijn | Er is een ESP8266 geprogrammeerd en aangesloten op de RX en TX pins van de Arduino, de ESP8266 is geprogrammeerd om te verbinden met wifi-netwerken en via TCP data te kunnen ontvangen en versturen. Alle data die hij ontvangt via TCP stuurt hij door naar de Arduino als seriële communicatie. Op deze manier kan Project Hades via wifi bestuurt worden. |
| De motoren moeten genoeg elektrisch vermogen krijgen om alle 3 snel te kunnen werken | Meerdere batterijen zijn ontploft | Er is netadapter voor de stroomvoeding van de motoren gebruikt, het grote voordeel hiervan is dat het :  1: niet ontploft  2: constante stroom levert, batterijen gaan leeg, maar een netadapter zit in een stopcontact  3: van voltage veranderd kan worden |

# Software

|  |  |  |
| --- | --- | --- |
| **Gewenste uitkomst** | **Ondervonden problemen** | **Aanpak** |
| De knoppen op het form moeten geleidelijk draaien en geen signalen van hapering laten zien. | Windows forms kon de snelheid van de animaties niet bijhouden. | Om te voorkomen dat het form 50 afbeeldingen per knop per seconden moest laden hebben we de afbeeldingen van de knop late “pre-loaden” hierdoor duurt het langer om van form naar form te gaan maar verdwijnen wel alle signalen van lag |
| Een knop kan alleen maar aan geklikt worden op de knop zelf | Een knop kan ook aangeklikt worden van buiten de cirkel maar niet al te ver. | Omdat de knop de waarde pakt van de totale omvang van de knop moesten de afbeeldingen beter gekropt worden. Toen dit gedaan was liepen de animaties soepeler en was het click event op gelost. |
| Een transitie op een form verloopt soepel. | Als je op een knop klikt in de openingstransitie bevriest het form | Omdat het form tegelijkertijd wil openen en sluiten lijkt het alsof de pagina bevriest. Maar door een boolean statement te makken dat checkt of de pagina in transitie is verdwijnt het probleem. |
| De kleur van de transitie is hetzelfde als de kleur van de gedrukte knop. | Bij sommige knoppen is de transitie kleur zwart | Dit probleem doet zich ook voor bij slecht gekropte knoppen. Als het form namelijk een transparant vakje vindt dan pakt het een zwarte kleur. Door de knopen beter te kroppen is het probleem verholpen. |
| Een Speelstuk mag niet over ander speelstukken heen lopen | Een toren kan over speelstukken heen lopen die niet bij zijn team horen. | Het probleem van de toren lag in het feit dat de for loop die kijkt naar afwijkingen de verkeerde kant op telde. |
| Als de teams verwisseld zijn loopt het spel op dezelfde manier | De kleuren van de speelstukken worden niet omgedraaid. | De kleuren van de speelstukken werden niet omgedraaid omdat er niet gecontroleerd werd op de kleur bij het tekenen van de speelstukken. |

# Media

# Usability Testing

Wat is mijn doel?er achter te komen of er nog verbeter punten zitten in de lay-out van onze interface

### Wat is mijn doelgroep?

Iedereen die beschikbaar is.

Welke testmethode ga je gebruiken?ik geef het programma aan mijn testpersoon en dan vraag ik om naar een spel te navigeren. Als het ze lukt is het een succes anders vraag ik wat er onduidelijk is en probeer ik ze verder te helpen.

### Hoe ga ik de resultaten analyseren?

De feedback die ik krijg schrijf ik op en de plekken waar het fout gaat noteer ik ook.

## Resultaat

Het is er positief verlopen iedereen die ik heb gevraagd om door onze interface heen te navigeren is het gelukt om het juiste spel te vinden. Zelfs mijn opa en oma hadden er geen problemen mee en mijn opa is half dement. Wel heb ik als feedback ontvangen dat de tekst op de knoppen iets te snel rondraaien, helaas is hier momenteel geen oplossing voor want dit kunnen we namelijk niet vertragen in Visual studio.

# FUN

|  |  |  |
| --- | --- | --- |
| Test | Methode | Uitkomst |
| Test query’s | Probeer alle query’s uit om te zien of ze werken |  |
| Test database | Laat een andere student proberen op een bepaalde manier de database te breken |  |