

# Technisch ontwerp Embedded fitness

---

**Project:** De vloer is lava  
**Opdrachtgever:** Embedded fitness  
**Projectnummer:** 1.0

**Auteur:** Sam Hendriks  
**Datum:** 25-5-2020  
**Versie:** 1.1

---

---

**Opdrachtgever**

***Paraaf gezien:***

Datum: \_\_\_\_\_

Plaats: \_\_\_\_\_

**Projectmanager**

***Paraaf gezien:***

Datum: \_\_\_\_\_

Plaats: \_\_\_\_\_

## Inhoudsopgave

|                                    |          |
|------------------------------------|----------|
| <b>Inleiding .....</b>             | <b>3</b> |
| <b>Technische Eisen .....</b>      | <b>3</b> |
| Hoe gaat het werken .....          | 3        |
| Voorbeeld.....                     | 3        |
| <b>Componenten en modules.....</b> | <b>4</b> |
| <b>Diagrammen .....</b>            | <b>5</b> |
| Klassen diagram.....               | 5        |
| Activiteiten diagram .....         | 6        |
| <b>Beslissingen .....</b>          | <b>7</b> |
| <b>Afspraken .....</b>             | <b>8</b> |
| <b>Grenzen .....</b>               | <b>8</b> |
| <b>Plan van Aanpak .....</b>       | <b>8</b> |
| Haalbaarheidsstappen .....         | 9        |
| <b>Test plan .....</b>             | <b>9</b> |

## Inleiding

Dit technisch ontwerp is voor het project “De vloer is lave”. In dit document gaan we kijken naar de technische eisen in het project, welke componenten in het project zitten, een aantal diagrammen, hoe wij beslissingen regelen, de afspraken wie wat doet, de genzen wat zit er niet in het ontwerp, een deel van het plan van aanpak hierin komt te staan wat de showstoppers zijn, en het testplan. Om dit document goed door te kunnen lezen is er een basis kennis nodig in C# en over diagrammen die gebruikt worden bij programmeren.

## Technische Eisen

Hierin komt te staan hoe een onderdeel moet gaan werken, op welke bepaalde snelheid het programma moet draaien, en welke programmeertaal er word gebruikt.

### Hoe gaat het werken

Om het spel te laten werken is er een aansturings programma gemaakt dat de commando's van het spel verwerkt en naar de Interactieve vloer stuurt. Dit word dan uitgevoerd door de vloer zelf. De vloer leest ten alle tijden de sensoren uit en levert deze aan het aansturings programma. Het programma geeft deze door aan het spel en dan kan hierop gereageert worden. Dit moet allemaal in miliseconden worden uitgevoerd zodat het spel niet blijft hangen want een gebruiker kan hier last van hebben. Het aansturings programma en het spel worden allebei gemaakt in visual studio, daarnaast word er de programmeertaal C# gebruikt voor het coderen. Het spel zelf krijgt een GUI doormiddel van het aansturings programma. In het aansturings programma kan ik zelf een GUI maken die dan vertoont word bij het opstarten van het spel.

### Voorbeeld

Om bijvoorbeeld de tegels rechtboven groen te maken worden er een aanstal stappen gemaakt.

Het begint met het aanmaken van een background, de background is een clone van de originele background. De originele background is zwart, zodat tegels die niet worden gebruikt ook zwart blijven. De belangrijkste functie om een tegel te kleuren is de DrawRectangle(). Hierin word de informatie van de tegel gezet die word aangepast, zoals de positie en kleur. Deze functie is van het aansturings programma zelf. Op het einde van een update word de draw functie uitgevoerd. Hierin word de background met alle aanpassingen gezet. Als dit is gedaan gaat het aansturings programma de data die verkregen is verwerken en dan doorgeven aan de interactieve vloer.

## Componenten en modules

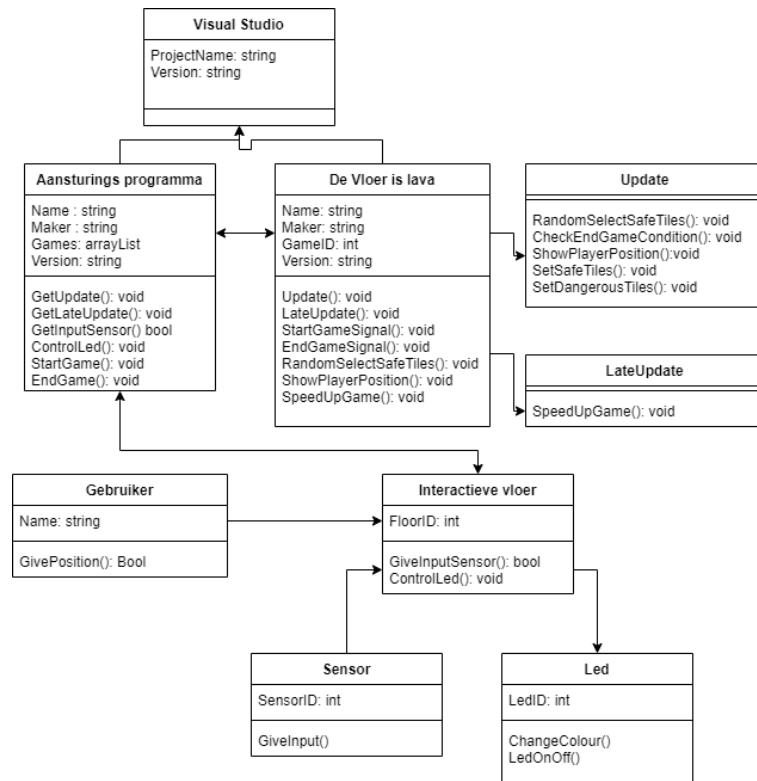
De volgende componenten zijn nodig om ons project functioneel te maken

- Interactieve vloer
  - Sensoren
  - ledjes
- Aansturings programma
  - Gemaakt in visual studio
  - Programmeer taal is C#
- “De vloer is Lava”
  - Gemaakt in visual studio
  - Programmeer taal is C#
  - Gemaakt in het aansturings programma
- Visual studio
  - .NET framework 4.8
  - C# & XAML capable IDE (Visual studio / VSCode recommended)

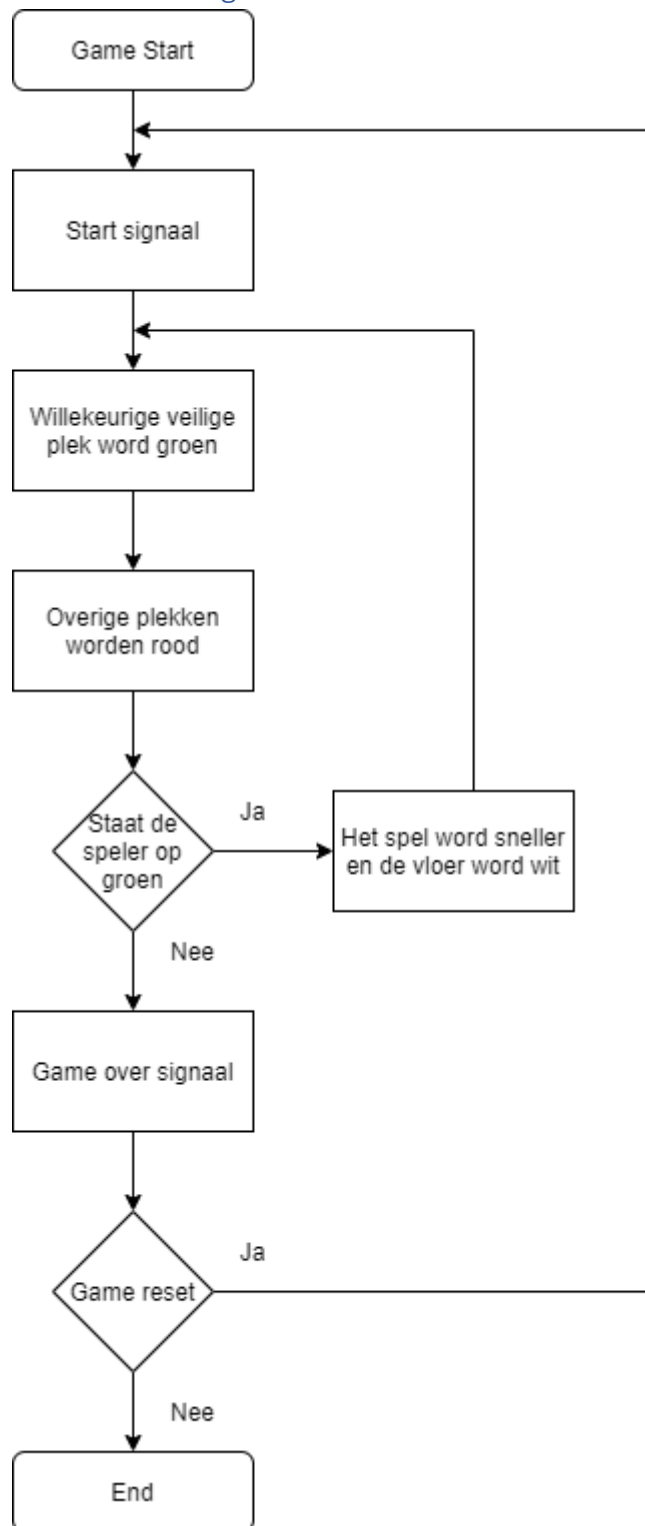
De interactieve vloer werkt samen met het aansturings programma, hierbij geeft het aansturings programma aan wat er moet gebeuren op de vloer zoals welke ledjes welke kleur moeten hebben. de interactieve vloer geeft ook aan wanneer iemand op een plaats staat met de sensoren die erin zitten. Dit geeft de vloer door aan het aansturings programma. Het aansturings programma werk ook samen met “De vloer is lava”. Hierbij geeft het aansturings programma door wat het van de vloer heeft gekregen, maar daarlangs kan “De vloer is lava” doorgeven aan het aansturings programma welke ledjes op welk punt aan moeten. Dit geeft het aansturings programma dan door aan de vloer zodat dit word uitgevoerd. Het aansturings programma en “De vloer is lava” worden allebei gemaakt in visual studio met de programmeer taal C#.

# Diagrammen

## Klassen diagram

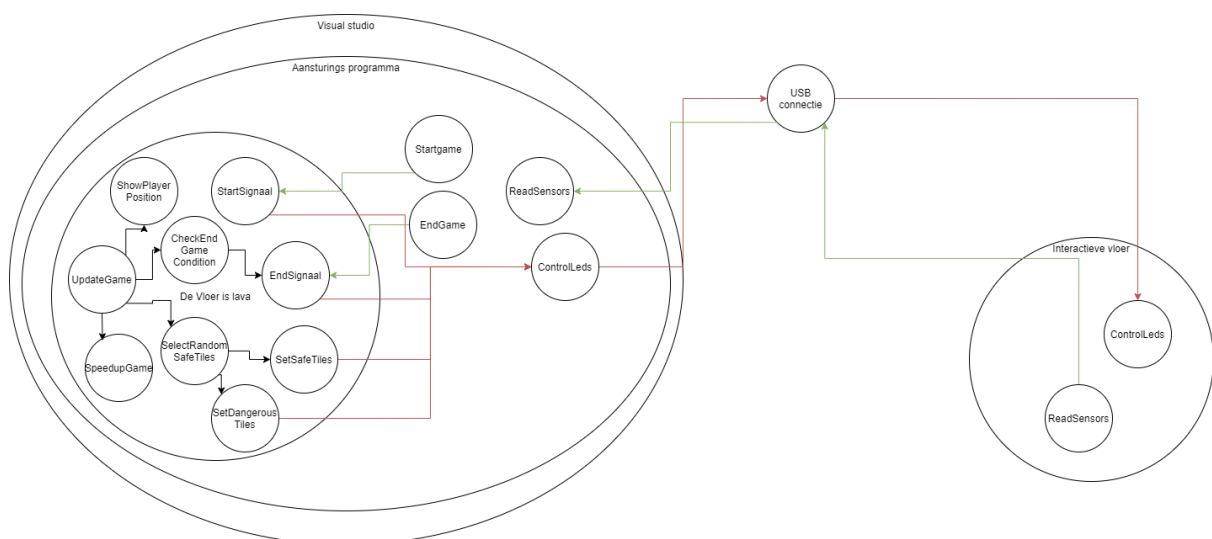


## Activiteiten diagram



## Relationeel datamodel

In het relationeel datamodel gaan we kijken naar de communicatie tussen onderdelen in het project. Voor de communicatie van “de vloer is lava” naar de interactieve vloer moet “de vloer is lava” eerst communiceren met het aansturings programma. “de vloer is lava” is deel van het aansturings programma hierdoor is de communicatie simpel om uit te voeren. Om met elkaar te communiceren worden bepaalde methodes aangeroepen om functies uit te voeren. “de vloer is lava” stuurt naar het aansturings programma wat er moet gebeuren (welke ledjes welke kleur worden). Zodra het aansturings programma dit heeft ontvangen vertaalt dit het in een code waarmee de interactieve vloer kan werken. De code die het aansturings programma heeft gemaakt wordt via een usb connectie verstuurd naar de interactieve vloer. De interactieve vloer verwerkt deze code en voert deze uit. De interactieve vloer leest zelf ook sensoren uit, deze lezen uit of er ergens iets op de vloer staat. De sensoren geven een signaal als een sensor high of low is, dus deze geven niet constante data aan het aansturings programma. Wanneer er een sensor high of low wordt, gaat er eenzelfde soort stringcode (die het aansturings programma stuurt naar de interactieve vloer) via de usb connectie naar het aansturings programma. Bij het aansturings programma wordt dit verwerkt en doorgegeven aan “de vloer is lava”. Het spel verwerkt de code en reageert hierop met gameEnd of hij speelt verder.



- Versturen bericht
  - Byte 1&2 : Tegelnummer (16 bits = 65536 tegels)
  - Byte 3-66 : 64 bytes voor 64 leds in een tegel [XXBBRRGG] kleuren
- Ontvangen bericht
  - Byte 1 : Tegelgroep aanduiden (0 = tegel 0-31, 1 = tegel 32-63 etc)
  - Byte 2-65 : 32 byte paren waarbij elke paar de 16 capsense bytes (4\*4 sensors) terug zend naar de pc. Dus dan wordt er een batch van 32 tegels meegegeven

## Beslissingen

Indien er een beslissing genomen moet worden over een onderdeel of een keuze dan wordt dit bij de projectleider neergelegd. De projectleider zal dan met onder andere de projectleden en de klant overleggen wat het beste is om te doen. Uit dit overleg zal de teamleider beslissen wat er gaat gebeuren.

## Afspraken

Het project “de vloer is lava” bestaat alleen uit het spel hiervoor is de teamleider verantwoordelijk. De code en de documentatie die hierbij hoort moeten voor 29 mei af zijn zodat deze beoordeeld kunnen worden. Hiervoor word elk document wanneer deze klaar is bekeken door de klant totdat zij met het document akkoord gaan wat word aangegeven door een handtekening. De materialen die nodig zijn voor het project zijn visual studio, het aansturings programma, en de interactieve vloer. Visual studio word ingesteld door de teamleider. Het aansturings programma word gemaakt en geleverd door Rik van Maaren. Als laatst word de interactieve vloer opgezet door Embedded fitness zelf.

## Grenzen

In dit project zijn vooral de technische zaken vastgelegd hierdoor zijn er een aantal zaken niet in dit document vastgelegd. Voor de functionelen delen van het project hebben wij een document het functioneel ontwerp. Hierin staat alles wat met de functie van het project en de modules ervan te maken heeft. In dit document staat ook geen planning dit staat vast in het projectplan.

## Plan van Aanpak

Hier gaan we kijken naar de showstopper van het project. Dit zijn problemen die ervoor kunnen zorgen dat het project niet werkt of het hele project kunnen stil leggen. Onder de showstoppers vallen:

- Er is nog geen werkend aansturings programma
- Communicatie tussen spel en aansturings programma werkt niet
- Het aansturings programma kan niet communiceren met de interactieve vloer

Om dit te voorkomen testen wij deze specifieke showstoppers als eerst zodat we niet midden in het project vast komen te zitten. Hiernaast voeren wij een haalbaarheidstest uit zo weten wij of we het project kunnen afmaken voor onze einddatum. Nadat we dit hebben gedaan kunnen we verder met de rest van het project en komen we niet snel voor onverwachte problemen te staan.



## Haalbaarheidsstappen

Om de haalbaarheid te testen worden er een paar onderzoeken uitgevoerd om te kijken of het project haalbaar is. Dit gebeurt in stappen van een kleine test naar een groter doeleinde hiermee kunnen we kijken of het project haalbaar is. Hieronder staat een voorbeeld van hoe zo iets voor ons eruit kan zien.

| Haalbaarheidsstappen |   |
|----------------------|---|
| 1.                   | Spel opzetten in het aansturings programma  |
| 2.                   | Simpelen commando's sturen naar het aansturings programma om een ledje aan te zetten en een sensor uit te lezen |
| 3.                   | Een simpele test doen waarbij het spel via het aansturings programma de vloer bestuurd                          |

Met deze testen zouden wij kunnen zien of wij dit deel van het project kunnen halen in de aangeven tijd. Dit zou op zijn minst een schatting kunnen opleveren waardoor wij verder kunnen gaan met het project. De stappen zoals te zien is zijn kleine makkelijke stappen naar iets moeilijker.

## Test plan

Om ons project te testen gebruiken wij een testlijst de testlijst staat in een excel bestand. In dit bestand leggen wij vast wanneer we aan het testen zijn en wat we als uitkomst verwachten. Na het testen schrijven we ook het resultaat op en of het gewenste resultaat behaald is. Bij elk nieuw onderdeel aan code dat geschreven wordt, word heel de testlijst uitgevoerd. Naast dat de programmeurs gaan testen vragen wij ook aan mensen die niks weten van het project om een keer te testen. De personen die getest hebben vullen dan een testraport in die wij kunnen inkijken. Alle delen die getest moeten worden zijn van te voren al in de testlijst gezet. Als een van deze delen getest wordt schrijft de persoon die dat deel test zijn naam erbij zodat alle projectleden weten wie wat getest heeft. Zo zal er geen verwarring komen of iets nog getest moet worden en voor meer informatie weet je dan bij wie je moet zijn.