

|  |  |
| --- | --- |
| Naam: | Studentnummer: |
| Michel Rummens | 500778934 |
| Timo Kee | 500780933 |
| Gilbert Matabadal | 500781920 |

Project Fasten Your Seatbelts 2017/2018

Ontwerpen fotokiosk

Team: IT\_103, Groep 2  
Versie: Versie #1  
Datum: 18-01-2018

**Documentatie programmeur**

Inhoud

[1. Inleiding 3](#_Toc504489183)

[2. Software 4](#_Toc504489184)

[2.1 De frames 5](#_Toc504489185)

[2.2 6](#_Toc504489186)

[3. Hardware 6](#_Toc504489187)

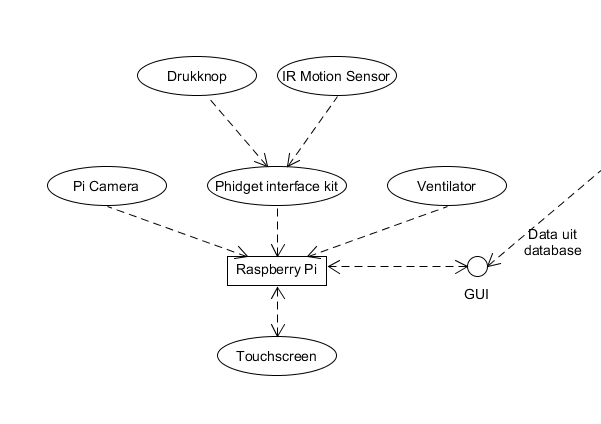
[3.1 Touchscreen 6](#_Toc504489188)

[3.2 Phidget interface kit 1018 7](#_Toc504489189)

[3.3 Ventilator 8](#_Toc504489190)

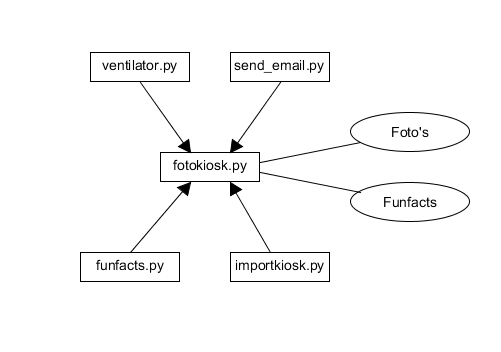
# Inleiding

Het is belangrijk voor een programmeur dat hij/zij weet hoe een product in elkaar zit. In dit document zal dan ook de fotokiosk, zowel software als hardware, uiteengezet worden. Uit dit document moet duidelijk worden hoe het product werkt en elke programmeur zou het moeten kunnen begrijpen. De software bestaat uit de GUI en de hardware bestaat uit alles om de GUI heen, denk aan de pi en de andere sensoren/actuatoren. Hieronder is een algemene schets van het product dat uiteengezet wordt.



# Software

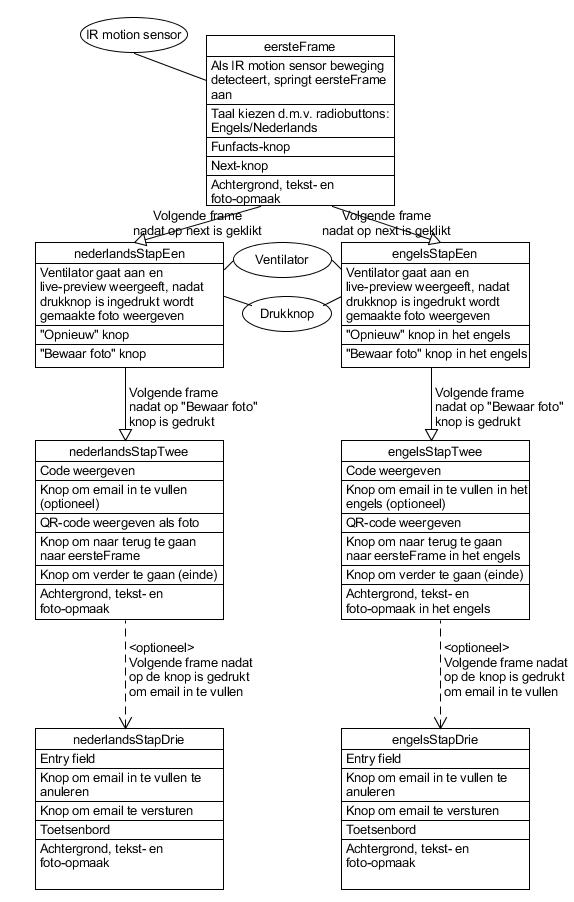
De software is geschreven in Python met Tkinter voor de GUI. Verder wordt er ook gecommuniceerd met de database o.a. de foto te versturen. De software bestaat uit meerdere bestanden. Het hoofdbestand heet fotokiosk. Alle andere bestanden zijn uitbreidingen op dit het hoofdbestand die meestal een class bevatten. Hieronder is een diagram te zien waarin dit duidelijk wordt. Het merendeel van de code is geschreven in het hoofdbestand. Ook zijn er twee mappen. In de map funfacts zijn de audiobestanden van de funfacts-knop opgeslagen. In de map foto’s zijn de foto’s die voor de GUI wordt gebruikt opgeslagen. Hiervoor is gekozen, zodat het overzichtelijk blijft.



## De frames

Bij elke functie en code zijn ook comments te vinden, dus de code wordt in het bestand zelf ook uitgelegd.

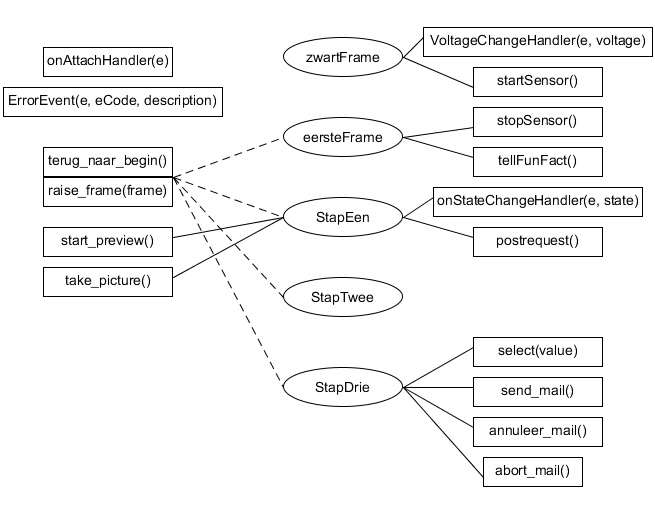
De code wordt uiteengezet d.m.v. diagrammen. De eerste diagram geeft de frames van de GUI weer.



## fotokiosk.py

Hieronder is een diagram te zien van de frames en alle functies die hieraan verbonden zijn.

Alle resterende code in dit bestand is niet ingewikkeld. Zo worden variabelen aangemaakt en waarden toegewezen, instellingen geconfigureerd om de GUI fullscreen e.d. te maken, de frames worden aangemaakt met de bijbehorende widgets (en toetsenbord). Ook worden bepaalde channels (van de Phidget interface kit) uit of aangezet bij een bepaalde actie. Alle functies zijn hieronder in een diagram gezet. Ook is weergeven op welke frame de functie werkt. De functies worden hieronder toegelicht.

onAttachHandler(e) print of een phidget is aangesloten. ErrorEvent print dat een phidget niet is aangesloten. Deze functies zijn niet aan een frame verbonden. Terug\_naar\_begin() en raise\_frame() zijn aan alle frames verbonden. Terug\_naar\_begin() is verbonden aan een knop die de gui laat terugkeren naar eersteFrame. Daarnaast worden alle gegevens verwijderd van de gebruiker. Deze functie is op elke frame actief. Raise\_frame(frame) zorgt ervoor dat een andere frame op de voorgrond treedt. Deze functie wordt in principe ook op elke frame gebruikt. Start\_preview() zet de camera aan en die output wordt op het scherm weergeven. Ook gaat de ventilator aan en speelt er een geluidsbestand af. Take\_picture maakt de frame StapEen in de taal die de gebruiker had gekozen bij eersteFrame en maakt de foto. Deze functie wordt aangeroepen zodra er op de drukknop is gedrukt en zet de ventilator dan weer uit. VoltageChangeHandler roept de functie stopSensor() aan wanneer de motion sensor beweging detecteert. startSensor wordt gebruikt om het zwartFrame aan te roepen en zet de sensor weer aan (de “slaapstand”). stopSensor() roept eersteFrame aan en zet de sensor uit. tellFunFact() vertelt een leuk feitje. Wanneer alle 6 de feitjes zijn afgespeeld, gaat de teller weer terug naar 1. onStateChangeHandler(e, state) bekijkt de status van de digitale input. Als deze 1 is, wordt de foto genomen. Postrequest() stuurt de foto door naar de database en maakt de QR-code aan. Ook wordt de StapTwee frame aangemaakt in de gekozen taal en er speelt een audiobestand af. Select(value) is een functie die het keyboard op StapDrie werkend maakt. Send\_mail() verstuurt de mail wanneer er een e-mailadres is ingevuld in StapDrie. Annuleer\_mail() annuleert de frame om de mail in te drukken en is verbonden aan de annuleer-knop in StapDrie. Abort\_mail() doet hetzelfde, maar dan bij de Engelse frame.

## funfacts.py

In dit bestand is de class Funfact te vinden. Deze speelt een ‘fun fact’ (audiobestand met leuk feitje) af wanneer er op de funfact-knop wordt gedrukt die op eersteFrame is weergeven. In totaal zijn er 6 ‘’funfacts’’.

## importkiosk.py

In dit bestand worden alle benodigde libraries voor fotokiosk.py geïmporteerd. In fotokiosk.py wordt importkiosk.py geïmporteerd en uiteindelijk zijn alle libraries geïmporteerd. Dit is gedaan zodat het hoofdbestand overzichtelijker is.

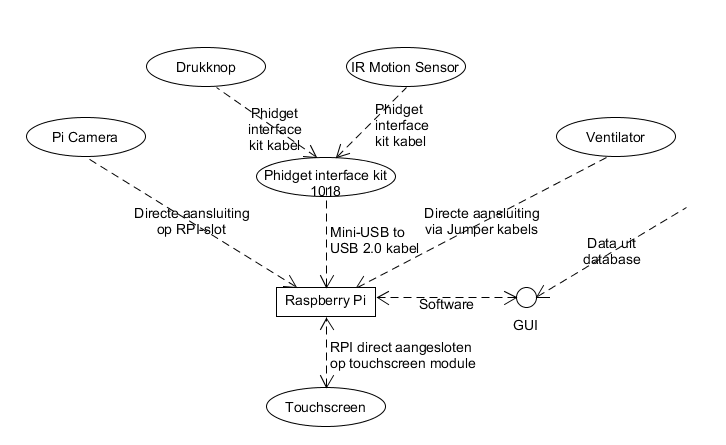
## send\_email.py

In dit bestand is de class Sendmail te vinden. Deze class verstuurd een mail vanuit onze e-mailadres naar het mailadres van de gebruiker (die wordt ingevuld in StapDrie) met als inhoud de code, foto en eventuele stappen die ondernomen kunnen worden als de foto’s niet goed aangekomen zijn.

## ventilator.py

In dit bestand is de code voor de ventilator geschreven. Dit zijn maar een paar regels code. Ook zijn er 2 functies gemaakt die de ventilator uit of aan kunnen zetten.

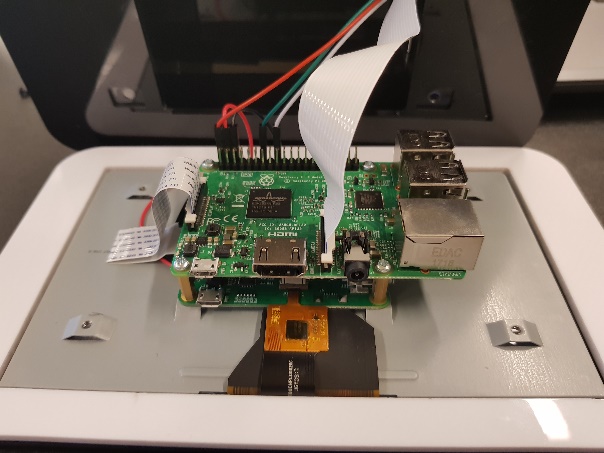
# 3. Hardware

In dit hoofdstuk wordt duidelijk hoe de hardware van de fotokiosk in elkaar zit. In de inleiding was te het product globaal geschetst. Hieronder is een uitgebreider diagram te zien van het product. Daarna worden er foto’s weergeven en daarna wordt het product en zijn onderdelen nog nader beschreven.

## 3.1 Touchscreen

Hiernaast zijn twee foto’s te zien van de buitenkant van de touchscreen. Op de eerste foto steekt de Pi Camera en de jumper kabels voor de ventilator uit. De touchscreen heeft een resolutie van 800x480 en heeft een diameter van 7 inch.

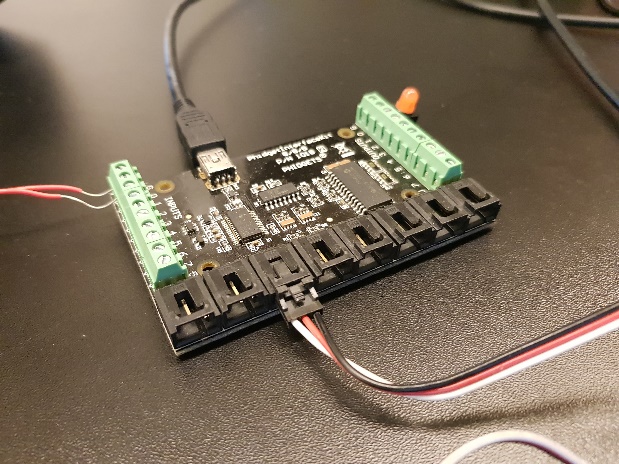
Op de tweede foto is de achterkant van de touchscreen te zien. Hier is duidelijk te zien dat de RPI in deze touchscreen is gebouwd. D.m.v. schroeven is de touchscreen los te schroeven en kan de RPI eruit gehaald worden. Weer is de Pi Camera te zien. Aan de zijkant zijn ook een paar aansluitingen te vinden. De aansluitingen voor de voeding, HDMI, USB en Ethernet zijn hier te vinden.

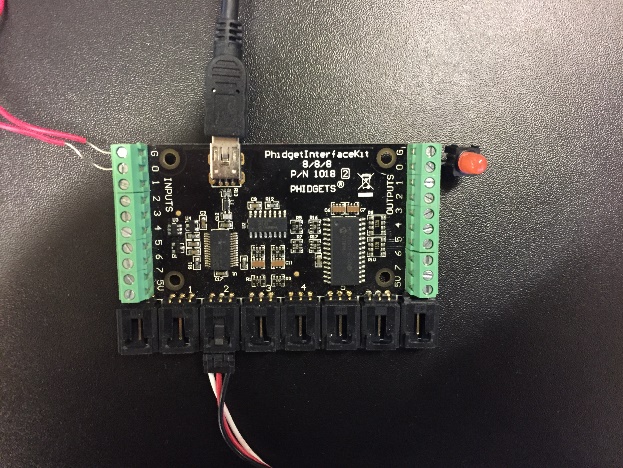


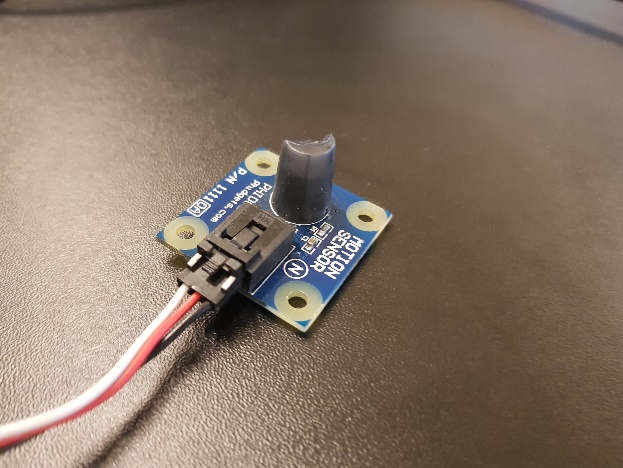
Hiernaast is de touchscreen te zien, maar nu is hij opengeschroefd. Hier is te zien hoe de RPI is geschroefd op de touchscreen module. Ook is te zien hoe de Pi Camera en jumper kabels zijn aangesloten op de Pi. Ook zijn de schroefgaten te zien om de touchscreen open te schroeven.

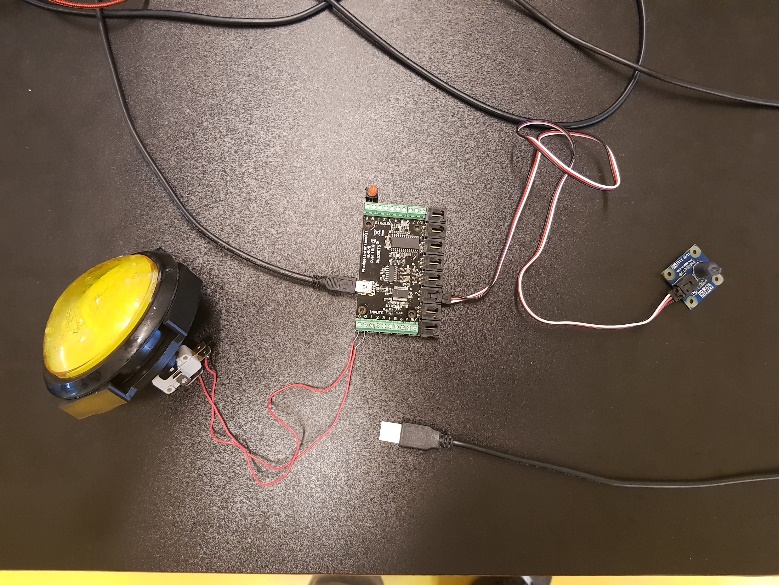
Daaronder is nog een foto te zien waarin de touchscreen vanuit een andere hoek te zien is. Op deze foto is duidelijk te zien hoe de Pi Camera uit de behuizing steekt.

## 3.2 Phidget interface kit 1018



Hiernaast is de Phidget interface kit te zien. De bovenste kabel is een Mini USB-kabel. Deze kabel is aangesloten op de Pi. Links zijn de Input poorten en rechts output poorten. Op de input poorten zijn twee kabels te zien. Dit zijn de kabels van de drukknop. De drukknop wordt gebruikt om de foto te maken. Rechts is een ledje te zien, maar die wordt verder niet gebruikt, was enkel om te testen. Aan de onderkant van de kit zijn de aansluitingen voor Phidgets te vinden. Hierop is de IR Motion Sensor aangesloten. De Motion Sensor traceert beweging en roept de GUI uit de slaapstand. Op de tweede foto is weer de kit te zien, maar dan van een bovenaanzicht. Op de volgende pagina zijn foto’s te zien van de drukknop, de IR Motion Sensor de kit met zijn aangesloten onderdeel in het geheel.





## 3.3 Ventilator

Hiernaast is de ventilator te zien die ook is aangesloten op de Raspberry Pi. Uit de ventilator steken jumper kabels. Deze kabels worden direct aangesloten op de Pi, wat te zien is op de foto van de opengeschroefde touchscreen. Overigens zijn er drie kabels, omdat de invoerkabel is gesplitst. In de ventilator zit natuurlijk een motortje. Deze motor is aangesloten op een relay. De relay zorgt ervoor dat de ventilator pas aangaat

wanneer hij wordt

opgeroepen. De motor en

relay is te zien op de andere

foto.