

Plan van aanpak

Gemaakt door: Bas Smit, Steven Flendrig en Jarno Kluts

Begeleidende docent: De heer Raemaekers

Opdrachtgever: Persekutuan Silat Melayu Nederland

Contactpersoon: De heer Brandt

Datum: 30-9-2021

Locatie: ROER College Schöndeln



Inhoud:

Aanleiding tot de opdracht:	3
Formulatie van de opdracht:	3
Pakket van eisen:	3
Oriëntatie:	4
Inleiding bij de oriëntatie:	4
Restanten voorgaande projecten:	4
KP V4 2019 Bodyprotector 2.0:	4
Meesterproef Pencak Silat door Thijs Beurskens:	4
KP V5 2019 mede door Jip van Hoef:	5
Software:	5
Keuze IDE:	5
Programmeertalen	6
Encryptiemethoden	7
Hardware:	8
De originele Bodyprotector:	8
De bodyprotector 2.0:	9
Oriëntatie op REED-contacten:	9
Experts:	10
Inventarisatie van de benodigde inhoudelijke kennis:	11
Deelopdrachten:	11
Planning:	12
Bronnen:	13

Aanleiding tot de opdracht:

De huidige bodyprotector houdt geen punten bij. Hierdoor moet een jury bepalen wanneer een punt wel of niet telt. Dit is niet altijd even eerlijk, omdat de jury niet weet hoe hard een stoot/trap aankomt en omdat het soms te snel gaat om te zien waar iemand wordt geraakt. Daarom kregen wij de opdracht om een systeem te ontwikkelen dat precies bijhoudt hoe hard een stoot/trap aankomt en als het hard genoeg is punten erbij telt. Ook moet dit harnas weten of het een stoot of een trap is, omdat een trap meer punten waard is dan een stoot.

Formulatie van de opdracht:

De opdrachtgever wilt een complete bodyprotector dat het verschil tussen een trap en een stoot kan meten en deze via een beveiligde verbinding doorgeeft aan een puntenteller waar een jury de punten van kan aflezen.

Pakket van eisen:

Bij de opdracht horen een aantal eisen. Om deze voor de opdrachtgever, de begeleidende docent, de expert(s) en uiteraard onszelf overzichtelijk te houden zetten we ze hier even op een rij.

- De bodyprotector moet een trap en/of stoot waarnemen.
- Er moet een onderscheid gemaakt worden tussen een trap en een stoot.
 - Hiervoor kan een oplossing worden gevonden in hardware of software.
- De waarneming die door de bodyprotector moet worden verstuurd via een beveiligde verbinding naar een systeem dat de punten bijhoudt.
- De bodyprotector moet comfortabel zitten en de gebruiker niet hinderen.
- De bodyprotector moet veilig zijn.

Oriëntatie:

Inleiding bij de oriëntatie:

Tijdens dit project gaan we te werk met 2 groepen, net zoals in ons voorgaande project. Dit is handig aangezien er een duidelijke distinctie is te vormen tussen de 2 taken die gedaan moeten worden. Deze taken zijn: De hardware van het pak werkend krijgen en de software werkend krijgen. Hierdoor hebben we ervoor gekozen om Jarno en Steven toe te wijzen aan het gedeelte **hardware** terwijl Bas zal werken aan het gedeelte **software**. Vanaf hier zullen we dus ook weer bepaalde deelopdrachten beschrijven via een kleurcode. Dit zal het geval zijn wanneer de beschreven taak/opdracht niet van toepassing is tot het andere team.

Restanten voorgaande projecten:

KP V4 2019 Bodyprotector 2.0:

(Ontwerprapport)

Dit is het eerste jaar dat er aan het project gewerkt werd. Ze begonnen met de originele bodyprotector en hadden als doel om een werkend systeem te maken dat trappen en stoten kon registreren. Dit hebben ze gedaan door een stroom door de binnen van de bodyprotector te laten lopen aan 2 kanten van een schuimlaag, als deze in elkaar wordt gedrukt door een trap of stoot raken de draden elkaar en ontstaat er dus een gesloten stroomkring. Hierdoor kun je dus makkelijk een trap of stoot waarnemen door bijvoorbeeld een voltmeter op het pak aan te sluiten. Er is al bewezen dat dit werkt door meerdere testen en het is vrij goedkoop om te maken omdat je in principe alleen wat koperdraad en tape nodig hebt.

Meesterproef Pencak Silat door Thijs Beurskens:

(Ontwerprapport)

De meesterproef van Thijs Beurskens is een vervolg op het project: KP V4 Bodyprotector 2.0 vandaar de naam Bodyprotector 2.1. Thijs Beurskens heeft op het eerste concept uitgebreid en gebruik gemaakt heeft van REED-contacten. Dit zijn kleine buisjes met twee metalen plaatjes aan de binnenkant die elkaar alleen raken als een magneet in de buurt komt. Hiervoor zijn handschoenen ontwikkeld met magneten erin die de REED-contacten 'activeren.' Door gebruik te maken van deze sensoren is het probleem van het onderscheiden van de trap en stoot zonder veel complicaties opgelost.

Enkele nadelen die het gebruik van REED-contacten met zich meebrengen:

- De gebruiker zou mogelijk zijn of haar eigen REED-contacten kunnen activeren als de handschoenen tegen het pak aankomen.
- Het gebruik van handschoenen bij een sport waar oorspronkelijk geen handschoenen gebruikt worden kan problemen opleveren.
- Het ontwikkelen van een handschoen met magneten kost extra tijd.

KP V5 2019 mede door Jip van Hoef:

[\(Ontwerprapport\)](#)

Tijdens dit project is vooral aandacht besteed aan het ontwikkelen van de software. Ze hebben gewerkt met een microcontroller met een wifi module. Deze kon signalen doorsturen naar een raspberry pi waaraan een MySQL server gekoppeld was via een localhost connectie. Deze signalen werden daarna naar de jury gestuurd om te beoordelen of het een trap of een stoot is. Op deze manier kan er erg moeilijk gefraudeerd worden doordat de jury alleen een punt kan 'opschrijven' als er gescoord is. De enige complicatie met dit systeem is dat alle juryleden een fout kunnen maken en bijvoorbeeld een trap invoeren wanneer er een stoot heeft plaatsgevonden. Het is alleen onwaarschijnlijk dat dit daadwerkelijk gebeurt behalve als alle juryleden omgekocht zijn.

Software:

Keuze IDE:

De software zal gecodeerd moeten worden in een IDE (Integrated development environment). Hieronder volgen enkele keuzes:

- [Visual Studio Code](#)
- [Atom](#)
- [Pycharm](#)

Elk van deze IDE hebben zo hun eigen voor en nadelen. Zo is visual studio een standaard in het bedrijfsleven en kunnen er veel verschillende talen in worden geprogrammeerd. Ook heeft Bas hier al eerdere ervaring mee.

Atom is simpel opgebouwd maar werkt hand in hand met GitHub, wat dus een fijne toevoeging kan zijn aangezien het dus allemaal direct op de cloud wordt opgeslagen. Ook geeft Atom te ruimte voor developers om zelf aanpassingen te maken aan de source code wat dus ook een handige toevoeging kan zijn (open-source).

Pycharm is de IDE voor het coderen in python. Echter wordt er naast python alleen nog Java support aangeboden. Dus deze keuze zal alleen belangrijk worden als er gecodeerd moet gaan worden in python. Echter kan dit ook gedaan worden in de andere 2 IDE's.

De keuze gaat voorlopig naar visual studio code doordat het vorige team hier ook mee gewerkt heeft en het bedrijfsstandaard is waardoor hierin oefenen een goede oefening is.

Programmeertalen

De talen die gebruikt zijn door de vorige groepjes zijn:

- [C#](#)
- [Python](#)
- [HTML](#)
- [CSS](#)
- [Javascript](#)
- [MySQL](#)

[C#](#) is gebruikt om de applicatie te maken om de scores te onderscheiden tussen een trap of een stoot. Dit is geprogrammeerd met behulp van Windows Forms (Een applicatie gemaakt door microsoft om makkelijk applicaties vorm te geven).

[Python](#) is gebruikt om de raspberry te programmeren en configureren.

[HTML](#), [CSS](#) en [Javascript](#) zijn gebruikt om de webserver te programmeren en vorm te geven. Hierin is HTML een taal om een website op te stellen en data te communiceren. CSS is om de website vorm te geven en Javascript is ok voor vormgeving en gecompliceerde scripts.

[MySQL](#) is gebruikt om de data van het pak op te slaan en vervolgens door te communiceren naar de webserver.

Encryptiemethoden

De uiteindelijk manier van encryptie dat gebruikt is door het vorige groepje is door de belangrijke data te hashen. Dit is een uitstekende vorm van databeveiliging echter is het niet volledig waterdicht aangezien er wel een mogelijkheid van kraken is. Echter is niks volledig veilig in een gedigitaliseerde omgeving.

Een mogelijke manier om een hash te kraken is door het gebruiken van tools zoals:

- [Hashcat](#)
- [John the Ripper](#)

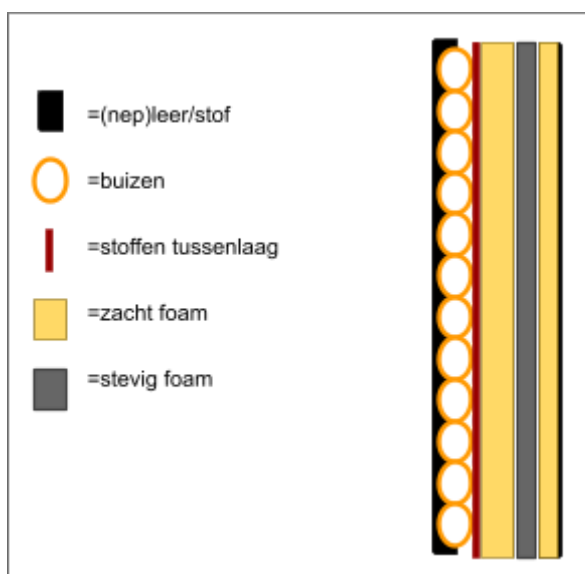
Deze tools kunnen door middel van brute force (het herhaald proberen van alle mogelijke oplossingen) een hash kraken of door het gebruik te maken van een wordlist zoals bijvoorbeeld the [ROCKYOU.txt](#) wordlist in [MD5](#) vorm en hierin resultaten vergelijkt met de gevonden hash. Na een match gevonden te hebben kan er door gebruik van simpele decryptie tools de hashes gedecrypt worden.

Hardware:

De originele Bodyprotector:

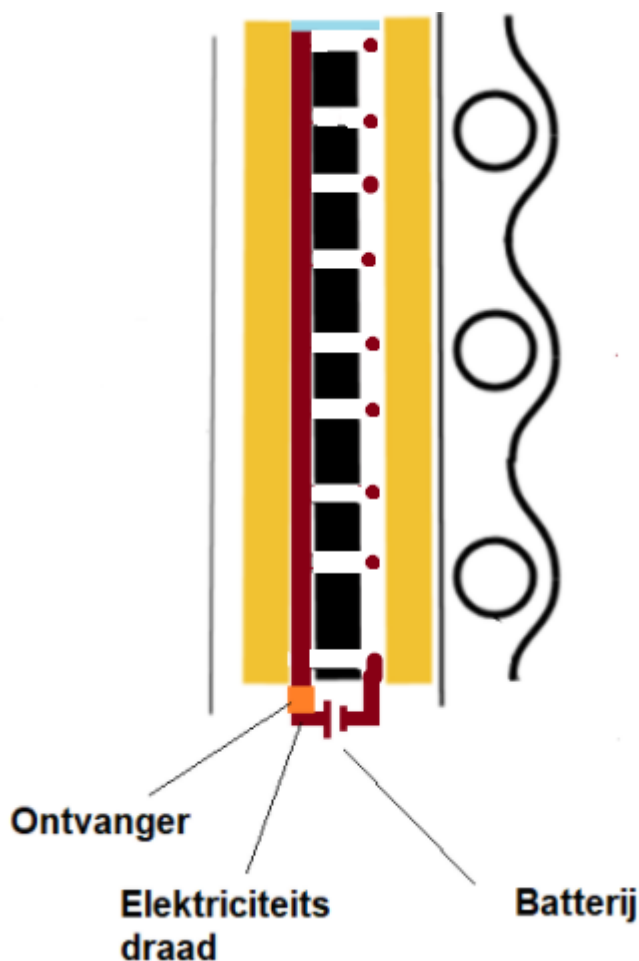
De eerste bodyprotector had als enige functie om de torso van de drager te beschermen tegen trappen en stoten tijdens een gevecht/sparsessie. Hier kwam dus nog geen elektriciteit aan te pas een was nog vrij simpel. Hieronder staan alle materialen/onderdelen en diens functie.

Lagen voorkant naar achterkant	Beschrijving	Materiaal:	Functie:	Eigenschappen:
Laag 1	Omhulsel 1	(Nep)leer en stof	Houdt het pak in elkaar	Waterbestendig, geleid geen stroom, stevig.
Laag 2	Rijen buisjes	Plastic	Zorgen voor meer stevigheid, het opvangen van de stoot, kan ook voor geluid zorgen zodat de trappen en stoten te horen zijn.	Waterbestendig, geleid geen stroom, hol
Laag 3	Scheiding buisjes - foam	Stof	Houdt het foam bij elkaar.	Geleid geen stroom
Laag 4	Schuimrubber 1	Zacht schuimrubber	Vangt de klap of stoot op	Geleid geen stroom, flexibel
Laag 5	Foam	Hard foam	Vangt de klap of stoot op en verspreidt de kracht over een groter oppervlak.	Geleid geen stroom, zeer stevig
Laag 6	Schuimrubber 2	Zacht schuimrubber	Vangt de klap of stoot op	Geleid geen stroom, flexibel
Laag 7	Omhulsel 2	Stof	Houdt het pak in elkaar	Geleid geen stroom



De bodyprotector 2.0:

Dit was de eerste vernieuwde versie van de bodyprotector die we gaan gebruiken. Dit project is gedaan door Koen, Thijs, Jarno en Steven. Jarno en Steven hebben dus ook al aan dit project gewerkt en zijn dus ook onze experts voor dit project. De bodyprotector 2.0 werkt door koperdraden kruislings over elkaar te leggen met een schuimlaag met gaten op de kruispunten ertussen (zie onderstaande afbeelding)



Oriëntatie op REED-contacten:

Een reedcontact of reedschakelaar is een schakelaar die bedient wordt met behulp van magneten. Als de contacten dus beïnvloed worden door een sterk genoeg magnetisch veld geven ze een signaal af. Als er dus kleine magneten in een vechtsport handschoen worden verwerkt geven ze tijdens contact een signaal af waardoor we kunnen zien dat er een stoot wordt gegeven. Als dit wordt gecombineerd met de bodyprotector 2.0 kun je dus trappen en stoten waarnemen.

Experts:

ICT en Arduino expert: Dhr. Baars Contact: <ul style="list-style-type: none">- via school -> ICT afdeling
ICT, coderen en arduino/raspberry pi expert: Jip van Hoef <ul style="list-style-type: none">- Student Computer Science aan de TU Eindhoven Contact: <ul style="list-style-type: none">- Mobiel: 0613775747
Privacy, ICT en Security expert: Tamara Brandt Contact: <ul style="list-style-type: none">- mogelijk via Dhr. Brandt

Inventarisatie van de benodigde inhoudelijke kennis:

Om de opdracht op een goede manier uit te voeren hebben we kennis (van experts) nodig. Omdat we tijdens dit project aan de slag gaan met 'hardware' en 'software' hebben we op verschillende gebieden expertise nodig.

- Benodigde kennis bij **hardware**:
 - Kennis over elektronica
 - Apparatuur in het pak kunnen installeren (vb. Raspberry Pi, REED-contacten, WiFi module)
 - Rekening kunnen houden met de veiligheid
 - Efficiënt design
 - Comfortabel design
 - De juiste productietechniek(en) gebruiken
 - Technische tekening maken
- Benodigde kennis bij **software**:
 - Coderen
 - Kunnen werken met een Raspberry Pi
 - Een webserver creeën
 - Data encryptie
 - De software ook kunnen laten samenwerken met de hardware

Deelopdrachten:

1. Het verder visualiseren van ons concept en de benodigde voorbereidingen treffen voor het realiseren van ons concept.
 - 1.1: Het maken van een flowchart dat de processen die de software uitvoert weergeeft.
 - 1.2: Het leren beheersen van (codetaal) zodat de uiteindelijk hiermee de software gemaakt kan worden.
 - 1.3: Kijken welke onderdelen in het pak gaan en hierbij een materialenlijst maken. We leggen ook uit waarom we voor welke onderdelen kiezen. We letten ook op welke elektronica het beste bij ons ontwerp en gekozen computer past.
 - 1.4: Bekijken welke productietechnieken het handigst zijn voor ons project en zo bekijken hoe we het pak het beste kunnen realiseren. Hierbij schrijven we een kort verslag van onze vondsten.
 - 1.5: Bedenken hoe we het fysieke deel van het pak in elkaar gaan zetten en hierbij rekening houden met de mechanica en de arduino. Hierbij zullen we ook een aantal schetsen. Hierbij moeten we dus ook kijken hoe we de soft en hardware in het pak gaan combineren.
2. Het realiseren van het ontwerp met behulp van kennis opgedaan van experts, van ons ontwerp en de mock-up een product maken die voldoet aan de eisen. Aan het einde van deze deelopdracht hebben een werkend

eindproduct dat voldoet aan de eisen van de opdrachtgever.

- 2.1: Het coderen van de software dat alle processen van de flowchart regelt.
- 2.2: Een technische tekening maken van het pak die goed weergeeft hoe de bodyprotector v2.2 in elkaar moeten zetten, hierbij lettend op alle materialen en onderdelen van het pak met behulp van een materialenlijst en hierbij lettend op het goed verwerken van alle elektronika. We moeten bij deze stap ook alle materialen die we nog niet hebben bestellen.
- 2.3: Deze stap gaan we met behulp van onze voorbereiding (1.3, 1.4 en 1.5) en de technische tekening het apparaat met het mechanisme in elkaar zetten.
- 2.4: Het samenvoegen van de arduino en het apparaat waarbij we ook alle bedrading aanleggen zodat het product klaar is om als bodyprotector gebruikt te worden.
- 2.5: Het pak uittesten waarbij we kijken of de bodyprotector voldoet aan de eisen van de opdrachtgever en van alle observaties en eventuele verbeteringen een verslag hierbij maken.
- 2.6: Als het niet aan de eisen van de opdrachtgever voldoet kijken we of de fout bij software, hardware of beide lag en bij 2.1 of 2.3 opnieuw beginnen.

3. Dit is dan onze eindpresentatie we kijken terug project en reflecteren op wat we gedaan en geleerd hebben.

Planning:

[Hier](#) de link naar de planning in google spreadsheet

[Hier](#) de link naar het logboek

[Hier](#) de link naar de trello

- de trello is onze 'dynamische' planning, deze passen we aan en voegen we deeltaken aan toe in verloop van het project.

Bronnen:

Pencak Silat vereniging

<http://www.silatmelayu.nl/>

Hellman key exchange algoritme

https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange

AES encryption algorithm

https://en.wikipedia.org/wiki/Advanced_Encryption_Standard

Asymmetric key encryption

https://nl.wikipedia.org/wiki/Asymmetrische_cryptografie

Hacknplan

<https://hacknplan.com>

HTTP Request

https://nl.wikipedia.org/wiki/Hypertext_Transfer_Protocol

Windows.Forms

https://en.wikipedia.org/wiki/Windows_Forms

C#

<https://nl.wikipedia.org/wiki/C%E2%99%AF>

Visual Studio

https://nl.wikipedia.org/wiki/Microsoft_Visual_Studio

MQTT

<https://www.ct.nl/achtergrond/iot-protocol-mqtt-betrouwbaar-data/>

Accelerometer

<https://en.wikipedia.org/wiki/Accelerometer>

RFID

https://nl.wikipedia.org/wiki/Radio-frequency_identification

HTTP

<https://www.dfrobot.com/blog-1036.html>

Basic python

https://www.learnpython.org/en/String_Formatting

Python:

[https://en.wikipedia.org/wiki/Python_\(programming_language\)](https://en.wikipedia.org/wiki/Python_(programming_language))

Hashcat:

<https://hashcat.net/hashcat/>

John the Ripper:

<https://tools.kali.org/password-attacks/john>

ROCKYOU.txt:

<https://www.kaggle.com/wjburns/common-password-list-rockyoutxt>

MD5:

<https://nl.wikipedia.org/wiki/MD5>

JavaScript:

<https://nl.wikipedia.org/wiki/JavaScript>

CSS:

https://nl.wikipedia.org/wiki/Cascading_Style_Sheets

HTML:

https://nl.wikipedia.org/wiki/HyperText_Markup_Language

MySQL:

<https://nl.wikipedia.org/wiki/MySQL>

REED-contacten:

<https://nl.wikipedia.org/wiki/Reed-contact>