

A dark blue vertical bar runs down the left side of the page. A blue arrow points to the right from this bar, containing the date.

4/11/2018

# Rapportage

Proftaak

Several thin, curved lines in dark blue and light grey originate from the bottom left and sweep upwards and to the right.

Gido ten Cate, Bram van Gils, Kieran Marriott, Thijs  
Spapens

P2T PROFTAAL GROEP EEN

## Versiebeheer

Versie	Datum	Taak	Persoon
1.0	10-04-2018	Basisopzet/layout Proftaakrapportage samenvoegen en controleren.	Gido ten Cate
1.1	11-04-2018	Implementatie fase	Gido ten Cate

## Inleiding

In deze rapportage wordt uitgelegd welke stappen zijn gezet om tot de ontwikkeling van Edubot te komen. Allereerst wordt uitgelegd waarom er is gekozen om Edubot te ontwikkelen.

Daarna wordt uitgelegd welke fasen zijn doorlopen om tot de uiteindelijke implementatie van het product EduBot te komen.

In de conceptfase werd de EduBot als idee op papier vastgelegd. Wat de EduBot precies inhoudt, doelgroep en rol in de markt. In de ontwerpfase werd vanuit de gemaakte concepten van de EduBot een ontwerp ontwikkeld. Tijdens de realisatiefase hebben wij de ontwerpen gerealiseerd van de EduBot. In de testfase werd de EduBot getest op wat hij daadwerkelijk kon en niet kon. Als laatste werd in de implementatiefase alle goede punten die de testfase hebben gehaald geïmplementeerd in het product EduBot.

Het document proftaakrapportage is opgesteld, voor het informeren van de projectleden en lezers over het project EduBot. Deze informatie zou de lezer in staat moeten stellen om tot goede conclusies te komen over het project. Dit document zou de lezer ervan ook in staat moeten stellen om eventueel mee te draaien in het project EduBot gedurende zijn loop. Het document is opgebouwd in verschillende fasen die in dezelfde chronologische volgorde staan aan de fasen waar in het project EduBot is verlopen.

## Inhoudsopgave

<b>Versiebeheer .....</b>	<b>1</b>
<b>Inleiding .....</b>	<b>2</b>
<b>Thema .....</b>	<b>5</b>
<b>Concept/Idee fase .....</b>	<b>6</b>
<b>Concept .....</b>	<b>6</b>
<b>Persona .....</b>	<b>6</b>
<b>OIS .....</b>	<b>7</b>
<b>OIT .....</b>	<b>8</b>
Welke sensoren en actuatoren worden standaard meegeleverd met de Lego EV3 en hoe werken deze? ..8	
Welke sensoren worden door externe partijen geleverd en hoe werken deze? .....	10
Welke toepassingen zijn door anderen gemaakt met de Lego EV3 kit? .....	11
<b>OIM .....</b>	<b>12</b>
<b>OIB .....</b>	<b>12</b>
Business Model Canvas & Business Requirements .....	12
Macrofactoren .....	12
Mesofactoren .....	13
Strategie .....	13
SWOT-analyse .....	13
Customer & Finance .....	14
Versiebeheer Conceptfase .....	15
<b>Ontwerp .....</b>	<b>16</b>
<b>Business .....</b>	<b>16</b>
Business Model Canvas .....	16
<b>Business Requirements .....</b>	<b>16</b>
<b>Media .....</b>	<b>17</b>
Presentation sketch .....	17
Toelichting .....	17
<b>Software Engineering .....</b>	<b>18</b>
Inleiding .....	18
Communicatie .....	18
Windows navigatie diagram .....	18
User Interface Sketches .....	19
Flowchart .....	20
Proof of Concept .....	23
MoSCoW .....	24
<b>Technology .....</b>	<b>25</b>
Sketches .....	25
Software .....	26
Versiebeheer Ontwerpfase .....	26
<b>Realiseren .....</b>	<b>27</b>
<b>OIS .....</b>	<b>27</b>
<b>OIT .....</b>	<b>28</b>
<b>OIM .....</b>	<b>29</b>

Scenario .....	29
Persona .....	30
User story.....	30
<b>OIB .....</b>	<b>31</b>
<b>Feedback .....</b>	<b>32</b>
Versiebeheer Realisatiefase .....	32
<b><i>Testen .....</i></b>	<b>33</b>
OIS .....	33
OIT .....	33
OIM.....	34
OIB.....	36
Feedback .....	37
Versiebeheer Testfase.....	37
<b><i>Implementatie.....</i></b>	<b>38</b>
OIS .....	38
OIT .....	38
OIM.....	38
OIB.....	39
Feedback .....	39
Versiebeheer Implementatiefase.....	39

## Thema

Het thema van ons project is educatie. Dit hebben we gekozen omdat ons eerste idee voor een soort Tamagotchi robot die kinderen kan helpen met leren doormiddel van games. We hadden dit gekozen omdat we allemaal vaak games spelen en bram ook nog eens een achtergrond heeft met game development.

Toch ligt de focus van die robot op het helpen van kinderen met het leren van rekenen, Nederlands en Engels. Ook is het de bedoeling dat ze leren verantwoordelijk te zijn door voor de robot te zorgen. Een ander thema van ons is digitalisatie.

Dit komt terug in het idee om deze robot te verkopen aan scholen om zo hele klassen te helpen met leren. Als basisschoolkinderen aan de gang moeten met echt robots vinden ze dat natuurlijk echt gaaf en cool en kan hun enthousiasme wekken. Dit is allemaal bij het digitaliseren van het onderwijs en wij willen dit proces versnellen met onze robot.

## Concept/Idee fase

### Concept

Ons project bestaat eigenlijk uit 2 onderdelen. Het eerste onderdeel is de EduBot. Dit is een EV3 robot die je met je zult mee kunnen nemen om overal spelletjes met hem te spelen. Je moet voor hem zorgen zodat hij gezond en gelukkig blijft. Zo zal hij moeten rusten, je zal hem moeten voeren en je moet met hem spelen. Op de EduBot zelf kun je testjes maken. De moeilijkheidsgraad zal afhangen van het level dat de EduBot op dat moment is. Je krijgt een reeks vragen waarbij je beloond wordt als je het antwoord goed hebt.

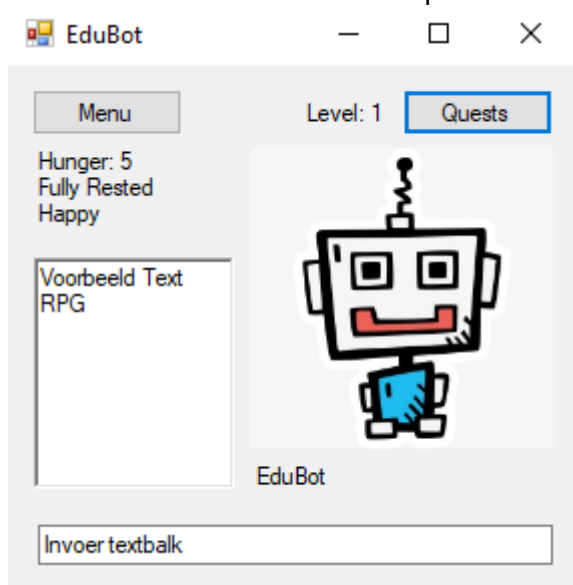
Als je de EduBot aan een pc aansluit wordt het echter nog interessanter. Je kunt zien hoe je voor de EduBot gezorgd hebt en hoe hij zich op dat moment voelt. Ook kun je verschillende quests met hem doen, zowel ingebouwde quests als quests die door docenten of ouders zelf zijn ingevoerd. Als de EduBot te moe is of te veel honger heeft dan zal hij niet mee werken om quests te doen, maar als je hem goed verzorgd dan lukt dat wel! Voor quests kun je ook beloningen krijgen en daar kun je vaak ook iets in de shop van kopen.

### Persona

Emily is net thuisgekomen van school. Ze zit net in groep 6 en is al bijna 9. In plaats van dat ze de rest van de dag op de bank hangt speelt ze met haar EduBot "Bart". Omdat hij honger heeft toont ze hem gekleurde balletjes waar hij met een vrolijk deuntje op reageert. Ook doet ze moeilijke rekensommen met hem, hij doet zo'n leuk dansje als je er een goed hebt. Zodra ze Bart aan haar macbook aansluit ziet ze dat ze een nieuwe quest ontvangen heeft. "Ruim je kamer op en ontvang 1 chocolade reep" heeft haar vader ingevuld. Ze besluit deze quest later te doen en eerst wat van de andere quests te doen. Zo moet ze samen met Bart een gevaarlijke bom onschadelijk maken door een serie van puzzels. Emily vindt met de EduBot spelen ontzettend leuk, en ze krijgt er ook nog af en toe een cadeautje van haar ouders mee!

## OIS

In Visual Studio zijn vele interface elementen aanwezig. Van buttons naar tekst-balkjes en pictureboxes. Deze komen allemaal goed van pas bij ons concept. Hieronder staat een basic concept van hoe onze UI eruit kan komen te zien.



Level label: Deze label laat zien welke level EduBot is op het gegeven moment.

Hunger label: Deze label laat zien hoeveel honger Edubot heeft.

Energie label: Deze label laat zien hoeveel energie Edubot heeft.

Mood label: Deze label laat zien hoe Edubot zich voelt.

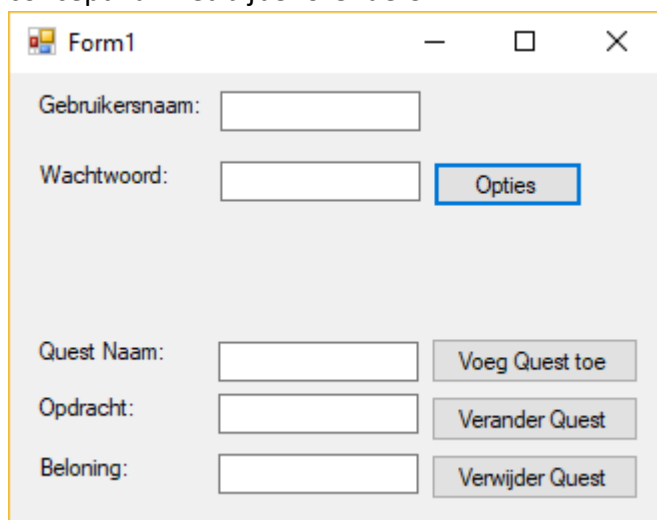
RPG Tekst Box: Deze textbox geeft informatie weer over de RPG

Invoer Textbalk: Deze textbalk wordt gebruikt als invoer voor de RPG, denk aan bewegen en aanvallen enz.

Quest knop: Door te klikken op deze knop kun je zien welke quests je nog moet voltooien.

Menu Knop: Als je op deze knop klikt krijg je een nieuw venster te zien met opties.

Door een gebruikersnaam en wachtwoord in te voeren komen opties op het scherm die ouders en/of leeraren kunnen gebruiken om zelf quests te maken. Hieronder staat een concept van het bijbehorende UI





## OIT

Welke sensoren en actuatoren worden standaard meegeleverd met de Lego EV3 en hoe werken deze?

De volgende sensoren en actuatoren worden standaard meegeleverd met de Lego Mindstorms EV3 set:

### Sensoren

- 1 [Tastsensor](#)
- 1 [Kleursensor](#)
- 1 [Infraroodsensor](#)
- 1 [Infraroodbaken](#)

### Actuatoren

- 2 [Grote interactieve servomotoren](#)
- 1 [Middelgrote interactieve servomotor](#)

De sensoren en actuatoren zijn aan te sturen met de intelligente Lego Mindstorms steen die fungeert als het brein van de robot. Dit kan eventueel draadloos worden gedaan via Bluetooth en Wifi.

De documentatie over hoe precies de sensoren en actuatoren kunnen worden gestuurd worden is te vinden op: [ROBOTC gebruikershandleiding](#)

### Tastsensor

De analoge EV3-tastsensor is een eenvoudig maar uitzonderlijk nauwkeurig hulpmiddel om te detecteren wanneer de voorste knop wordt ingedrukt of losgelaten en om te registreren of er één of meerdere keren op de knop wordt gedrukt. Studenten kunnen start/stopsystemen bouwen, robots creëren die uit een doolhof geraken en de technologie ontdekken die wordt gebruikt in apparaten zoals digitale muziekinstrumenten, toetsenborden van computers en keukenapparaten.

- Opening voor kruis op de knop
- Auto-ID ingebouwd in de EV3-software

### Kleursensor

Het digitale EV3 Kleursensor kan acht verschillende kleuren lezen. Het functioneert ook als een licht sensor door lichtintensiteit te lezen. Studenten kunnen kleur sorterende en lijn-volgende robots bouwen, experimenteren met lichtweerkaatsing van verschillende kleuren, en ervaring opdoen met een technologie die veel is gebruikt in industrieën zoals recycling, landbouw en verpakking.

- Meet weerkaatst rood licht en omgevingslicht, van duisternis tot fel zonlicht
- In staat tot het detecteren van acht kleuren. Het kan onderscheid maken tussen: Zwart, wit, blauw, groen, geel, rood en bruin.
- Sample rate van 1 kHz
- Auto-ID ingebouwd in de EV3 software

### Infraroodsensor

Het digitale EV3 Infraroodsensor detecteert nabijheid tot de robot en leest signalen uitgestraald door het infraroodbaken. Studenten kunnen op afstand bedienbare robots maken die door hindernisbanen kan navigeren en leren hoe infrarood technologie wordt gebruikt in afstandsbedieningen voor televisies, bewakingssystemen en zelfs doelwit acquisitie apparatuur

- Nabijheid meten van ongeveer 50-70 cm.
- Werkende afstand van het baken tot twee meter.
- Ondersteund vier signaal kanalen.
- Ontvangt infrarood op afstand verzonden opdrachten.
- Auto-ID ingebouwd in de EV3 software

### **Infraroodbaken**

Dit is ontworpen voor gebruik met de EV3 infraroodsensor. Het baken stuurt een infrarood signaal die de sensor kan volgen. Het baken kan ook als afstandsbediening worden gebruikt voor de EV3 door middel van signalen die verzonden zijn naar de infraroodsensor.

- Vereist twee AAA-batterijen
- Four individual channels
- Includes a beacon button and toggle switch to activate/deactivate
- Green LED indicating if the beacon is active
- Auto power-down if the unit is not in action for one hour
- Working distance of up to two meters

### **Grote interactieve servomotoren**

De grote EV3-servomotor is een krachtige motor die gebruikmaakt van tacho-feedback voor een besturing die tot één graad nauwkeurig is. Door gebruik te maken van de ingebouwde rotatiesensor kan de intelligente motor worden uitgelijnd met andere motoren op de robot zodat hij in een rechte lijn kan rijden met een constante snelheid. De sensor kan ook worden gebruikt voor nauwkeurige metingen bij experimenten. Dankzij het ontwerp van de motorbehuizing is het erg makkelijk om meervoudige tandwielstellen te maken.

- Tacho-feedback tot op één graad nauwkeurig
- 160-170 tpm
- Draaikoppel van 20 N/cm
- Stilstandkoppel van 40 N/cm
- Auto-ID ingebouwd in de EV3-software

### **Middelgrote interactieve servomotoren**

De medium EV3-servomotor is uitermate geschikt voor toepassingen met lagere lasten en hogere snelheden en wanneer een snellere reactietijd en kleiner profiel vereist zijn voor het robotontwerp. De motor maakt gebruik van tacho-feedback voor een besturing tot op één graad nauwkeurig en heeft een ingebouwde rotatiesensor.

- Tacho-feedback tot op één graad nauwkeurig
- 240-250 tpm
- Draaikoppel van 8 N/cm
- Stilstandkoppel van 12 N/cm
- Auto-ID ingebouwd in de EV3-software

De volgende sensoren zijn ook beschikbaar als extra. Ze worden te koop aangeboden op [shop.lego.com](http://shop.lego.com)

- [Gyrosensor](#)
- [Ultrasonic sensor](#)
- Sensoren van derden

Welke sensoren worden door externe partijen geleverd en hoe werken deze?

Er worden verschillende sensoren geleverd door externe partijen. Een van de externe partijen die sensoren levert is [mindsensors](#). Enkele sensoren die Mindsensors levert voor Lego Mindstorms EV3 zijn:

- [Vision subsystem v5](#)
- [Gyro, MultiSensitivity Accelerometer and Compass](#)
- [IR Temperature Sensor](#)

### **Vision subsystem v5**

De Vision subsystem v5, kan worden gebruikt om objecten te volgen. Deze kunnen zowel lijnen, vlammen en zelfs gezichten zijn, maar ok anderen. Hij kan ook video's opnemen en foto's maken. Ook kan deze sensor aan een PC worden verbonden via een USB port.

- Verbindt met EV3/NXT op een sensor port
- Verbindt met computer door middel van USB interface
- Volgt lijnen of Gekleurde objecten (je kunt je eigen kleuren definiëren.)
- Volgt gezichten of ogen
- Neem Video's op of neem foto's op een microSD kaart
- Video/Photo opname mogelijk met autonoom g.
- Lens met ingebouwde infrarood bokking filter
- Biedt realtime tracking statistieken aan EV3/NXT
- Energieverbruik: 42 mA (max) at 4.7V
- Image & Video Resolutie: 320x240
- PC niet nodig voor autonoom functioneren
- Ondersteunde omgevingen: NXT-G, EV3G, RobotC, LeJOS, NXC, LabVIEW, LVEE
- Komt met een microSD kaart

## **Gyro, Multisensitivity Accelerometer en Compass**

Deze sensor heeft een gyro, accelerometer, compass en magnetisch veld sensor in een sensor gebouwd. Deze sensor kan dus meten welke richting de robot kijkt, hoeveel hij accelereert en welke kant hij draait en/of tilt.

- Tilt Sensor resolutie: 1 graad
- Compass resolutie: 1 graad
- Accelerometer Resolutie: 10 milli-g
- Tilt gecompenseerd magnetisch kompas
- Accelerometer gevoeligheden: 2G, 4G, 8G, 16G
- Gyro snelheid van verandering resolutie: 8.75 milli-graden/sec
- Energieverbruik: 2.1 mA op 4.7V

## **IR Temperature Sensor**

De IR Temperatuur sensor kan de kamertemperatuur meten zowel als het temperatuur van een object waar de sensor naartoe wijst.

- Contactloos infrarood temperatuur meten
- Resolutie: 0.1 graden
- Geeft metingen in Celsius en Fahrenheit
- EV3/NXT Compatible Connector
- Programming blokken zijn beschikbaar voor EV3/NXT

Welke toepassingen zijn door anderen gemaakt met de Lego EV3 kit?

### **MindCub3r**

Mindcub3r is een robot die de rubikscube oplost, deze gebruikt een lichtsensor om eerst te kijken waar elke kleur op de rubikscube zit. Daarna gebruikt hij een algoritme om te bepalen hoe hij het snelst de kubus kan oplossen.

Er is een versie gemaakt met de NXT genaamd CubeStormer2, deze heeft de wereldrecord gebroken voor het snelst een rubikscube te hebben opgelost. Hij gebruikten een smartphone met een geavanceerde processor om sneller en efficiënter te kunnen werken, maar maakte nog steeds gebruik van de mindstorm lichtsensoren en motors.

### **M&M sorteerder**

M&M sorteerder doet wat zijn naam zegt, hij sorteert M&M's op basis van kleur. Hiervoor wordt een lichtsensor gebruikt om te kunnen differentiëren tussen de verschillende kleuren M&Ms en daarna ze in de juiste bankjes te sorteren.

## OIM

Kinderen willen de laatste tijd steeds veel meer gamen. Edubot combineert een deel van dat aspect met educatie. Technologie is en wordt een groot deel van onze toekomst, en door Edubot kunnen kinderen speelsgewijs leren om met technologie om te gaan.

Waarom zouden ouders dit voor hun kinderen kopen?

Edubot kan ook ouders helpen hun kinderen klusjes te laten doen, door middel van zelf ingevoerde opdrachten voor Edubot. Ook leren hun ouders tegelijkertijd iets van computers als ze het zelf al niet wisten.

Waarom willen scholen dit product in hun leeromgeving hebben?

Speelsgewijs leren is belangrijk, zeker in de basisschool. Door middel van Edubot kunnen leeraren hun studenten open spelende manier meer leren. Ook kunnen ze Edubot huiswerk laten geven, maar op een zodanig manier dat het nog leuk én leerzaam blijft voor de leerlingen. Vaak wordt huiswerk te saai gevonden, dit kan ervoor zorgen dat leerlingen minder leren van hun huiswerk maken dan de bedoeling is.

## OIB

### Business Model Canvas & Business Requirements

Ons project valt onder het specialisatiegebied game-design. Wij vinden het een probleem dat veel kinderen leren heel erg saai vinden en willen hier een leuke en educatieve tamagotchi robot voor ontwikkelen.

De consument waarop we ons richting zijn kinderen tussen de 7 en 12 jaar dus onze doelgroep zal voornamelijk de ouders van de kinderen zijn. Ook zullen basisscholen onze educatieve robots kunnen gebruiken om leren te stimuleren.

Onze missie is dat wij leren voor kinderen een leuke ervaring maken in plaats van een last.

De visie die wij daar bij hebben is dat in de toekomst leren wordt geassocieerd met een leuke bezigheid die voor iedereen beschikbaar is.

Onze waardepropositie is dat kinderen slimmer worden.

### Macrofactoren

Voor ons bedrijf zijn niet alle macrofactoren erg van toepassing. Technologie is natuurlijk wel erg belangrijk omdat we met de tijd mee moeten gaan voor zowel software en hardware. Technologische ontwikkelingen zullen onze robot mooier en meer geavanceerd maken en onze game betere graphics en andere belovende opties geven. Demografisch kunnen we ons richten op wijken waar we merken dat er veel kinderen stoppen met school. In wijken waar heel weinig probleemkinderen zijn en uitval is op scholen zullen onze robots slechter verkopen. Economisch zullen wij waarschijnlijk een van de eerste zijn die zullen lijden, aangezien onze robot natuurlijk wel een luxeproduct is. Wij zijn niet afhankelijk van enige ecologische factoren omdat wij bezig zijn met software en hardware. sociaal-cultureel is niet iets waar we op kunnen focussen omdat onze robot gebruikt gaat worden door basisscholen dus het is niet zo dat er een verschil is tussen opleidingsniveau of andere verschillen bij de kinderen. Politiek kan daarentegen wel wat betekenen voor ons omdat een groot deel van onze verwachte verkopen aan basisscholen zijn. Mocht de overheid beslissen dat scholen minder geld krijgen of besluiten dat ons product niet in het curriculum past.

## Mesofactoren

Ons bedrijf wordt ook beïnvloed door een aantal mesofactoren. zo heb je bijvoorbeeld concurrentie met andere game developers die ook educatieve games maken. Mochten zij een populaire release hebben voelen wij dat financieel. als wij nieuwe games willen maken zijn we ook erg afhankelijk van de vermogensverschaffers om ons te financieren. Zonder sponsoren zou het allemaal niet mogelijk zijn. Sinds bij ons spel ook een fysieke robot hoort zijn we ook erg afhankelijk van de leveranciers om alles goed af te leveren. Dit houdt onder andere in dat de robot op tijd en heel afgeleverd wordt bij onze klanten en natuurlijk al de hardware die we nodig hebben om de robots te maken. Onder onze klanten vallen ook onder anderen ouders en scholen dus ook overheidsinstellingen vallen ook onder onze afnemers. Sterker nog we verwachten dat deze basisscholen onze grootste klanten zullen zijn aangezien ze in een keer gigantisch veel van die robots nodig hebben en kopen ze over een aantal jaar steeds meer robots in plaats van alles in een keer. Hoe video games in de media worden afgeschreven is ook erg belangrijk voor ons. zo kan het gebeuren dat video games erg negatief worden afgeschreven en zo onze verkoopcijfers dalen. Als laatste zijn belangen hartigheid organisaties ook van belang voor ons aangezien sommige van die organisaties ons product willen kopen voor kinderen in ontwikkelingslanden om hun ook een kans te geven om leuk te kunnen leren. Dit kan zelf zorgen voor beter opgeleide mensen en dat kan de economie van die landen stimuleren.

## Strategie

Onze strategie bestaat uit het vinden van kansen op ICT gebied en die zo goed mogelijk te gebruiken. Ons product is voor de digitalisatie van het onderwijs. Dit gaat nog niet in een snel tempo. Op basis hiervan maken wij hiervoor producten om dit proces te versnellen en daarvan te profiteren. We willen ook ervoor zorgen dat onze producten de beste kwaliteit hebben. Op deze manier bouwen we niet alleen een goede reputatie onder de consument maar hiermee krijgen we trouwe zakenpartners. Onder deze zakenpartners vallen zowel ouders als scholen. Zo blijven we relevant op de markt en hebben we een groep vaste klanten.

## SWOT-analyse

Wij hebben een SWOT-analyse gemaakt aangezien dat een erg handige manier is om overzicht te krijgen over onszelf en onze plek in de markt.

### *Strengths:*

1. We produceren goede kwaliteit producten.
2. Zo is onze game vele malen leuker als van de concurrent.
3. We zijn vrijwel een van de eerste die dit soort educatie robots maakt.

### *Weaknesses:*

1. Ons product is vrij prijzig en daarom ook minder interessant bij de armere consumenten.
2. Het is nog een vrij onzekere markt waar we op focusen.

#### *Opportunities:*

1. Er is sprake van een onverzadigde markt en wij springen daarop in.
2. De digitalisering van het onderwijs kan opeens erg snel gaan.
3. Als we een goed product kunnen produceren kunnen we de marktleider worden.

#### *Threads:*

1. Er zijn een aantal concurrenten die de leading role kunnen krijgen.
2. Het kan zijn dat de digitalisering van het onderwijs erg traag gaat verlopen.

### Customer & Finance

#### *Customer:*

Ondanks dat onze doelgroep basisschoolleerlingen is bestaan onze klanten uit de ouders en scholen van die leerlingen. Sinds de leerlingen niet het geld of het initiatief hebben om opeens een van onze robots te kopen moeten de scholen en ouders dit voor hen doen. Dit betekent dus dat we onze reclame aan moeten passen op zowel de kinderen als de ouders. Zo moeten we de kinderen enthousiast maken over de robot en de ouders moeten gaan geloven dat de robot hun kind echt kan helpen met leren en ze een stukje verantwoordelijkheid krijgen. Onze klanten rekenen erop dat onze robot die leerlingen kan helpen met studeren en dat moet dus ook duidelijk zijn in die reclames. De ouders die nu kinderen krijgen zijn zelf al die nieuwe technologie al gewend volgens ons aangezien ze er mee in contact komen op het werk en in de rest van het dagelijkse leven. Dit betekent voor ons dat de meeste ouders dus minder bang of aarzelend zijn om onze robot aan te schaffen. Dit is goed nieuws voor ons. Scholen daarentegen gaan misschien niet zo gemakkelijk overstag gezien ze met de overheid te maken hebben. Als scholen geen toestemming krijgen van de overheid om onze robots te kopen kan dit een harde klap voor ons zijn. Hiervoor moeten we misschien mee in gesprek gaan met eventuele partijen om te kijken of we dit kunnen veranderen.

#### *Finance:*

Om een start te maken met ons bedrijf hebben we een balans gemaakt om de financiën beter te kunnen overzien. Zo willen we een betere kijk hebben op onze financiële zaken zodat we beter beslissingen kunnen maken en dus ook efficiënter te werk kunnen gaan en daarmee dus sterker in de markt staan. We zijn van plan om een kantoorruimte te kopen als vestiging voor ons bedrijf. Vanuit daar kunnen we onze software ontwikkelen en marketing regelen. We kopen meteen een bedrijfswagen voor vervoer en bestellen meteen een flinke lading robots(150) om te starten. We willen dus voor het kantoor een hypotheek te nemen en de rest zelf te betalen. Voor de robots betalen we later zodat we eerst tijd hebben ze te verkopen.

Debet	Geld	Credit	Geld
Apparatuur	2.500	Eigen vermogen	5.000

Inventaris	30.000	Hypotheek	229.000
Kantoor	229.000	Crediteuren	30.000
Bedrijfswagen	1.500		
Kantoorartikelen	1.000		
Totaal:	264.000		264.000

Deze wagen willen we kopen:

<https://www.zoekjebedrijfswagen.nl/opel/combo/gesloten+bedrijfswagen/1-3-cdti/120865.detail>.

En dit is het kantoor dat we op het oog hebben:

<https://ventu.nl/Kantoor/Tilburg/Aphroditestraat/71/e7253fcd-2585-49c9-bd38-edff44fcd591>.

#### Versiebeheer Conceptfase

Versie	Datum	Taak	Persoon
1.0	26-02-2018	Basisopzet/layout	Gido ten Cate
1.1	15-03-2018	Basisconceptdocument	Kieran Marriot
1.2	26-02-2018	Onderzoeksverslag	Gido ten Cate
1.3	28-02-2018	Onderzoeksverslag	Gido ten Cate
1.4	10-04-2018	Technology Software Technology Sketches	Thijs Spapens Kieran Marriott



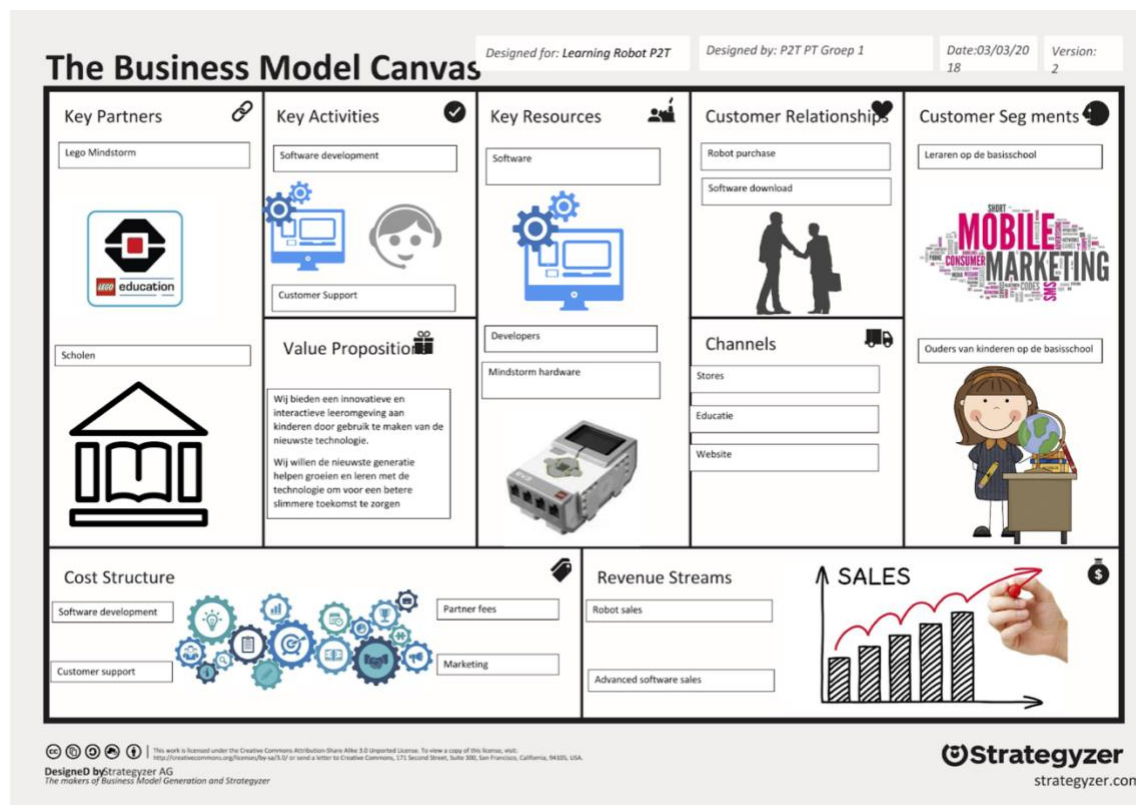
## Ontwerp

### Business

Voor Business in de ontwerpfase hebben wij verder gewerkt aan ons Business Model Canvas om op een simpele maar krachtige manier te laten zien hoe elk deel van ons businessmodel tot elkaar zich verhouden maar ook op welke manier.

### Business Model Canvas

Hieronder staat het Business Model Canvas van de EduBot waar in een opzicht gemakkelijk elk aspect van ons businessmodel wordt weergegeven. Hier is heel duidelijk weergegeven wie onze doelgroep is, onze partners zijn, wat onze kanalen zijn maar ook waar onze inkomsten vandaan zouden moeten komen.



### Business Requirements

Voor de Business Requirements hebben wij onze doelgroep beter beschreven en gesegmenteerd. Naast de doelgroep beter hebben te omschreven hebben wij ook de financiën van het product EduBot beschreven. Welke kosten er zijn en de verschillende manieren om opbrengst te generen. Hoe wij dit hebben gedaan is allemaal omschreven in de Business Requirements. Naast de kosten beschreven te hebben is de kostenstructuur ook verduidelijkt in onze Business Requirements.

Als laatste hebben wij de investeringen van de Edu-Bot zijn beschreven in de balans, investeringsbegroting en financieringsbegroting.

## Media

Voor Media in de ontwerpfase hebben wij gewerkt aan de ontwerpschetsen voor de visuele presentatie op het p1 event waar wij ons product aan de wereld verkondigen.

### Presentation sketch



### Toelichting

Op het P1 event willen wij dat onze tafel er spectaculair en uitnodigend uit ziet. Doormiddel van een groot uitnodigend bord met behulpzame maar ook lokkende tekst zoals:

- Controls
- Businessmodel Canvas
- User Story
- Logo

Met een extern toetsenbord en muis kunnen de spelers het spel spelen. Er zal een ander scherm ook aan een laptop verbonden zijn zodat de bezoekers het project in zijn geheel kunnen beleven. Ook zullen er minstens twee mensen bij de tafel staan om toelichting te geven, vragen te beantwoorden en problemen op te lossen als die zich voordoen.

## Software Engineering

### Inleiding

Het doel van ons product is het brengen van een educatieve game voor kinderen die moeite hebben met bepaalde vakken. De Mindstorm robot die hierbij hoort kunnen kinderen mee naar huis nemen om voor te zorgen en educatieve spelletjes mee te spelen. De robot kan communiceren met een desktopapplicatie waarin kinderen ook een spel kunnen spelen. Aan de hand van hoe goed ze de robot hebben verzorgd krijgen de kinderen munten waarmee ze leuke in-game dingen kunnen kopen. En aan de hand van hoe goed ze de educatieve spelletjes hebben gespeeld wordt het in-game character weer sterker en slimmer. Wanneer de kinderen op hun computer het spel aan het spelen zijn maakt de robot positieve en negatieve geluiden op aan de hand van hoe goed de kinderen het spel spelen.

### Communicatie

Tussen de Robot en de desktopapplicatie moeten de volgende dingen gecommuniceerd worden:

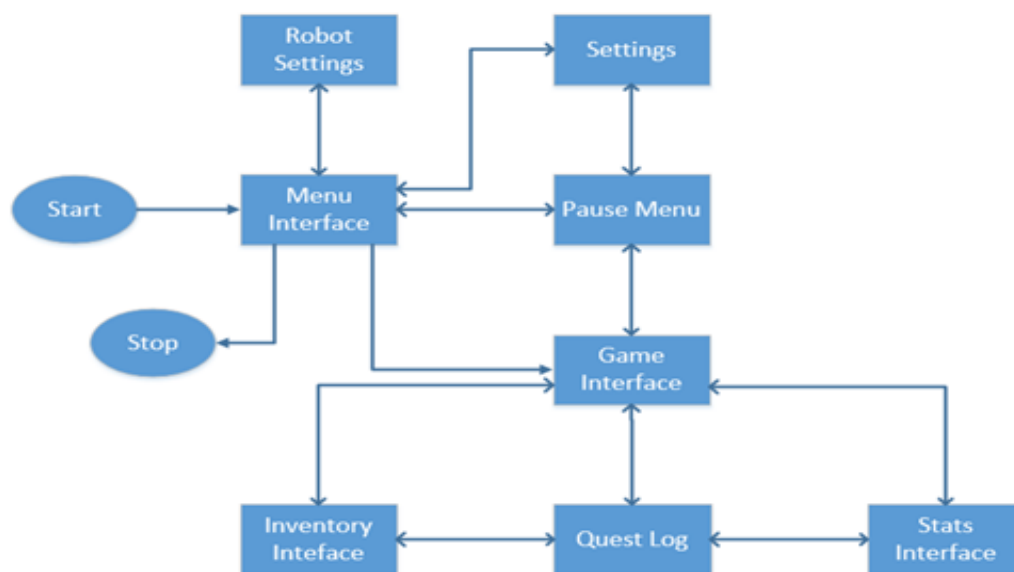
De robot moet kunnen doorgeven hoeveel educatieve minigames het kind heeft behaald op basis waarvan de robot XP kan verdienen en zo Level Up kan gaan.

De robot moet door kunnen geven hoe goed de speler voor hem gezorgd heeft waarop de speler wordt beloond met extra punten/currency.

De robot moet tijdens het spelen positieve en negatieve geluidjes maken op basis van hoe goed de speler het spel speelt.

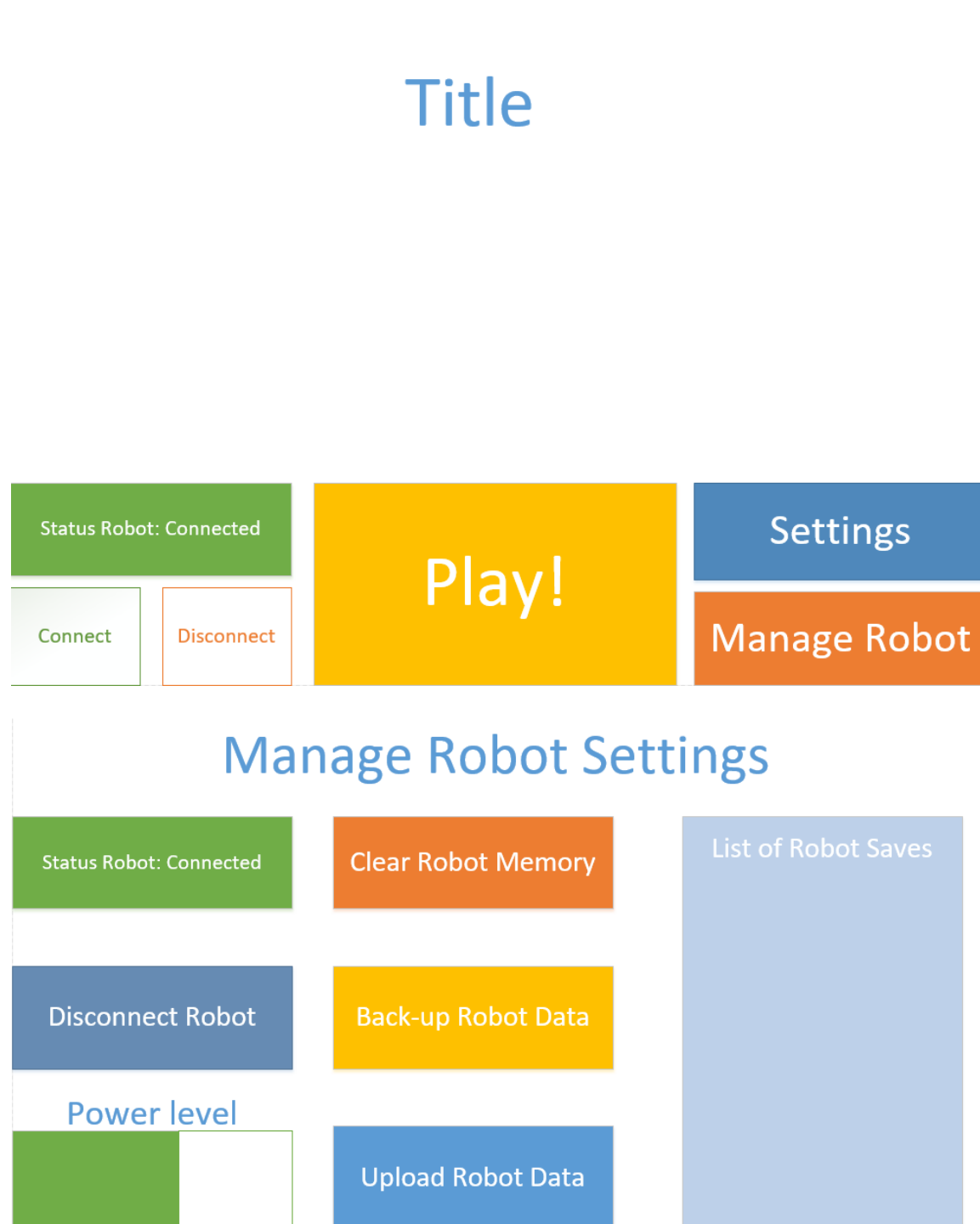
### Windows navigatie diagram

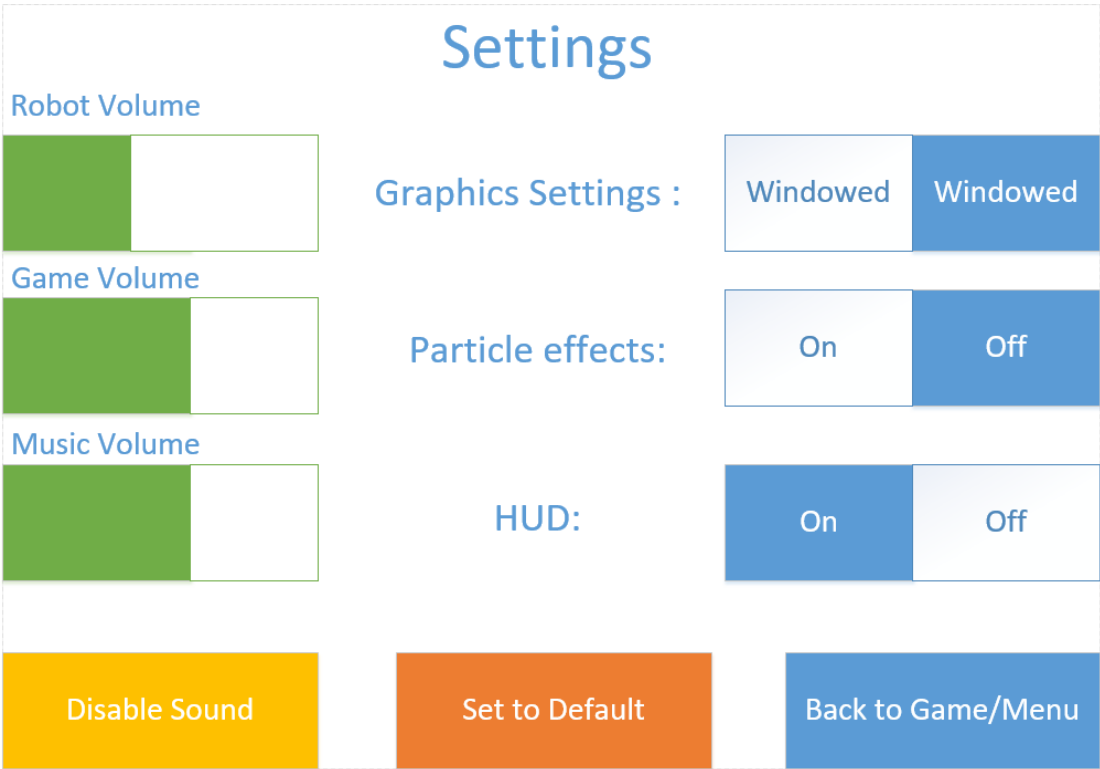
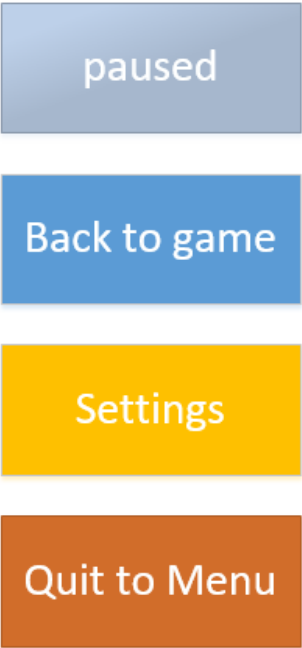
## Windows Navigation Diagram



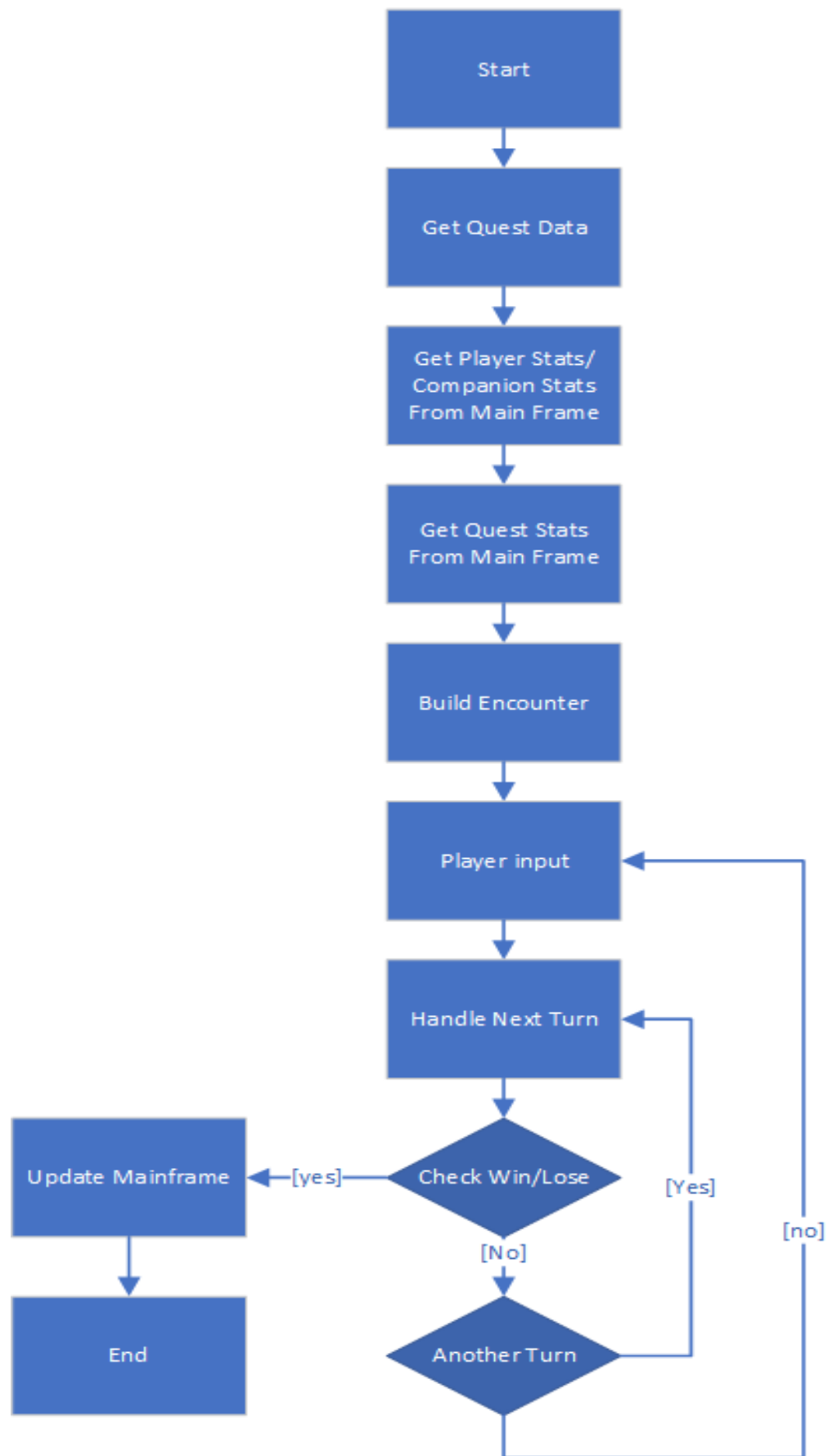
In het hierboven weergegeven Windows navigation diagram staat beschreven welke Interfaces er nodig zijn voor de desktopapplicatie en hoe zij met elkaar verbonden zijn

#### User Interface Sketches

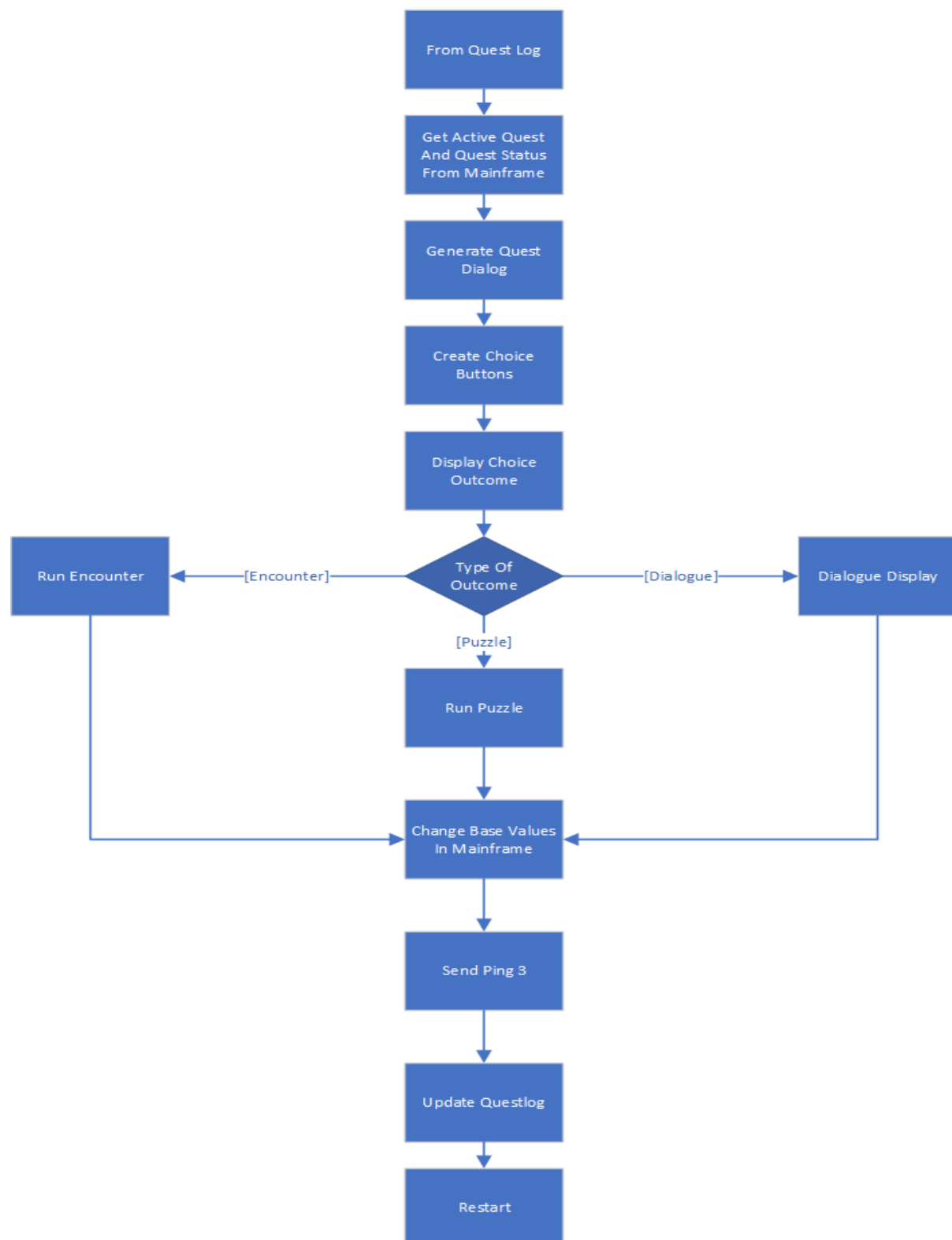




Flowchart  
Game interface

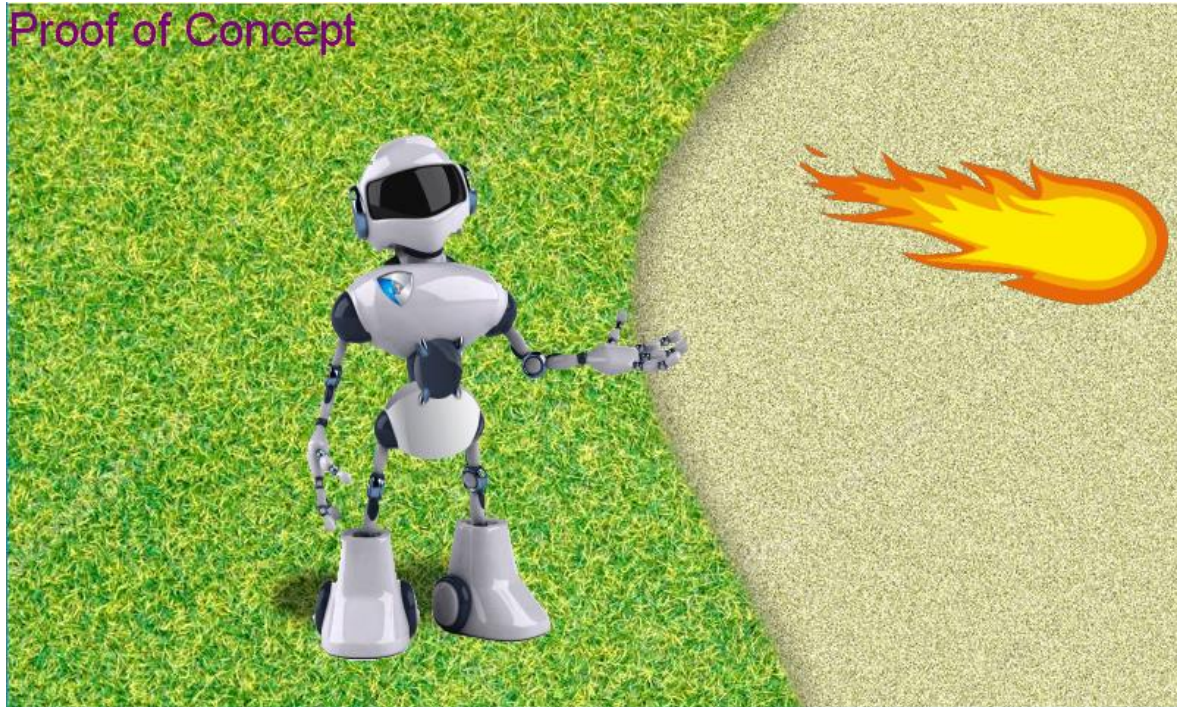


## Quest interface

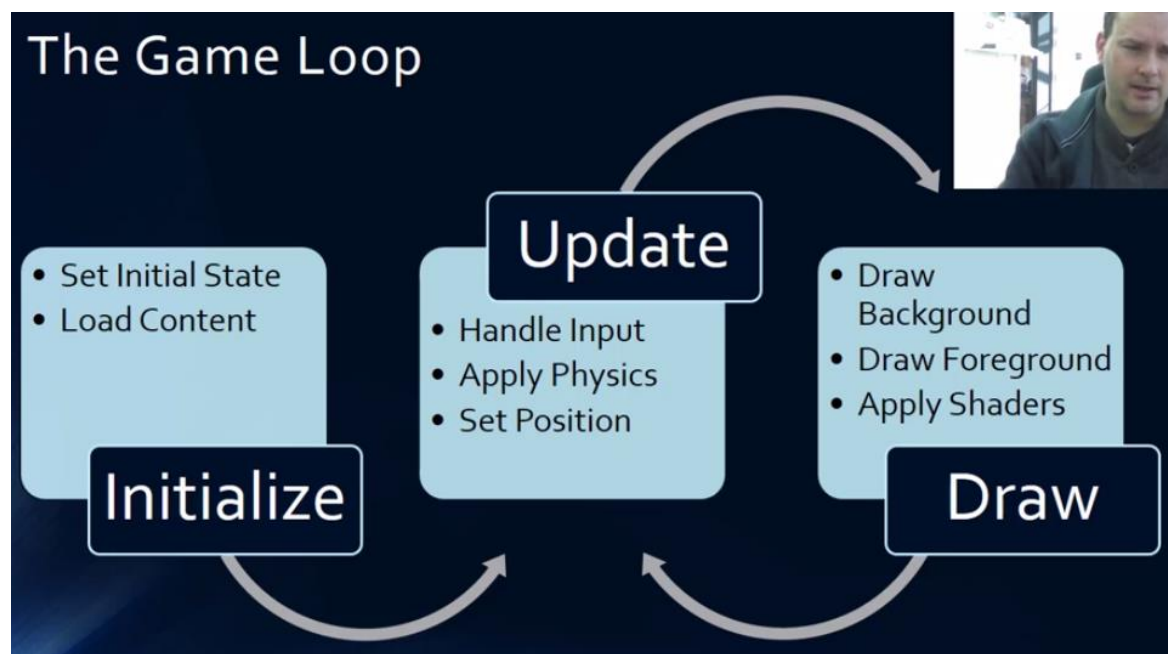




## Proof of Concept



Voor de Game Interface hebben we een framework gebruikt genaamd monogame. Om te bewijzen dat wij in staat zijn om deze framework te gebruiken hebben we een Proof of concept gemaakt waarbij een bestuurbare robot vuurballen afschiet. Het framework biedt namelijk een ingebouwde game loop en zal heel veel helpen als we bewegende characters willen maken. De componenten van de gameloop staan hieronder beschreven.





## MoSCoW

Om de komende weken beter in te delen en idee en te structureren hebben we een MoSCoW model gebouwd:

### *Musts*

- Communiceren met de Edu-bot
- Menu Interface
- Stat Interface
- Quest log
- Stats opslaan en beheren
- Inventory Interface
- Game interface
- Min 5 Quests

### *Shoulds*

- Educational Minigames
- Shop Interface
- Sprites in the game interface
- Arenas (Background)
- Sound effect
- Sprite effects
- History Implementation
- Min 10 Quests

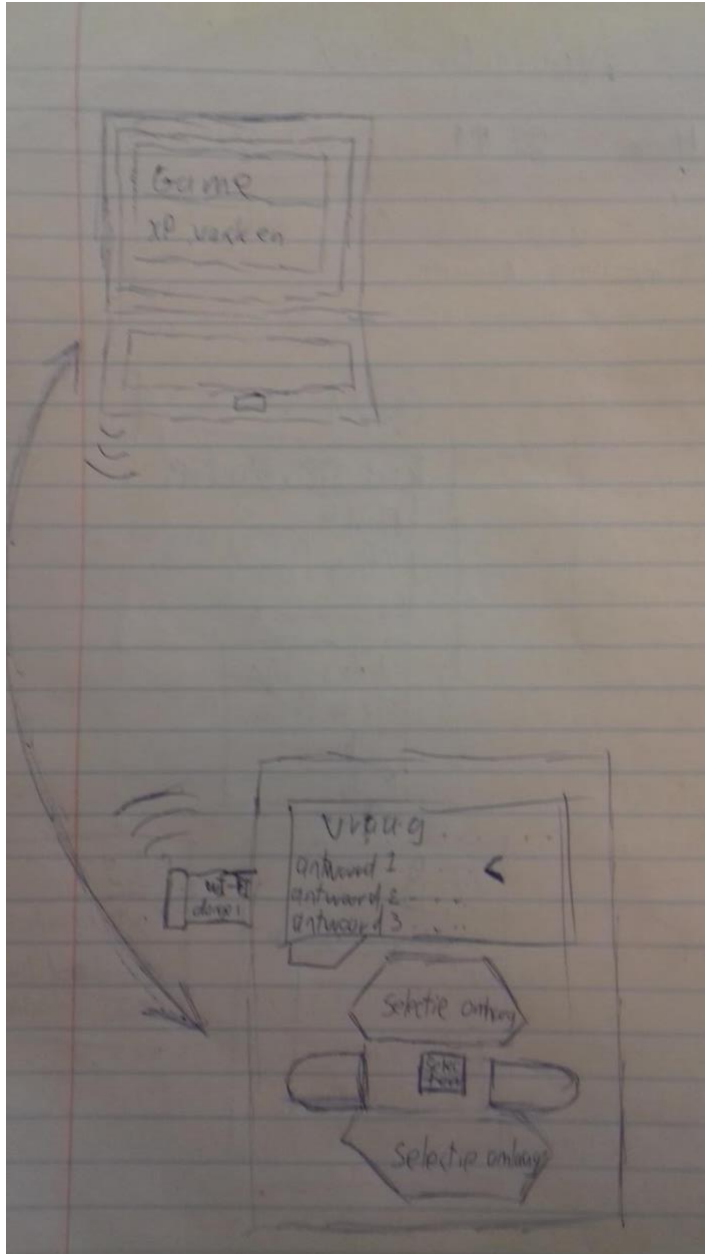
### *Could's*

- Character control
- Character combat
- Character Effects
- Open World
- Decorations
- NPC's
- Enemy AI
- Companion System
- Companion AI
- Bosses
- Progressing storyline
- Separate Maps
- Room Maps
- Fishing
- Min 15 Quests
- Wonts
- Multiplayer
- Min 20 Quests

## Technology

Voor Technology voor de ontwerpfase hebben wij gewerkt aan het storyboard en het Proof of Concept. In het Storyboard hebben wij ons bouwplan beschreven van hoe wij onze EV3 robot omtoveren in de EduBot. In het Proof of Concept hebben wij beschreven of ons bouwplan en ideeën daadwerkelijk haalbaar zijn.

## Sketches



## Software

De Edu-Bot zelf maakt alleen maar gebruik van een Wi-Fi dongle om contact te zoeken met een PC via een hotspot van de PC zelf. Het is de bedoeling dat de connectie via de PC gaat en op het moment dat de connectie gaande is de robot zijn gegevens doorstuurt via Wi-Fi. Omdat de speler toetjes heeft gemaakt op de robot heeft de speler meer ervaring gekregen in de vakken die hij geoefend heeft. De robot geeft dat dan door aan de PC applicatie zodat de game up to date is met de robot. De robot zelf heeft geen sensoren nodig om te functioneren.

Via het schermpje kan de speler interactief te werk gaan met de robot. De speler kan kiezen welke toets hij wil doen en de robot begint met de toets. De robot kan zelf voor rekenen zelf sommen genereren en voor een taaltoets kiest hij woorden uit een lijst. Dit maakt de toetsen volledig willekeurig en zo is het niet voorspelbaar wat de antwoorden zullen gaan worden. Op deze manier kan de speler niet een vast rijtje woorden of sommen onthouden en moeten ze blijven nadenken voor de antwoorden. Daarnaast kan de robot ook nog geluidjes afspelen als iemand een toets haalt en ze ontvangen kisten die unlocks bevatten voor in de game. Door die betere spullen kan de speler beter en gemakkelijker de quests halen. De Edu-Bot zorgt er zo voor dat de leerlingen gestimuleerd worden om te leren zowel door unlocks zoals geluidjes maar ook doordat ze dan sterker worden in de game.

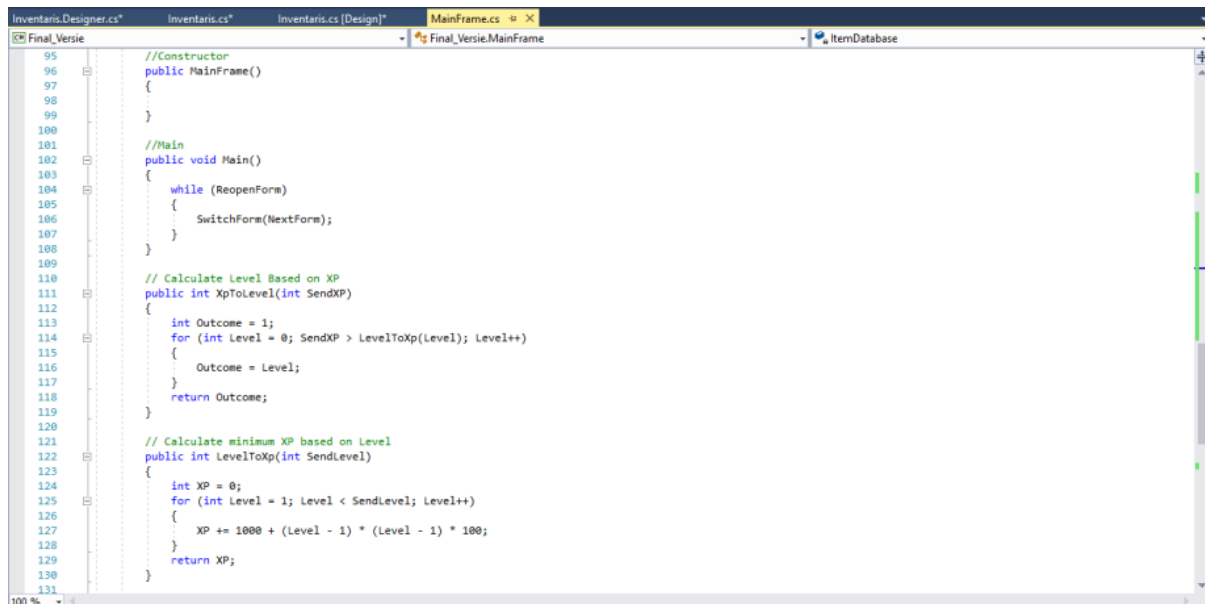
Het is de bedoeling dat de leerlingen de Edu-bot mee naar huis kunnen nemen en er dan mee gaan spelen. Als ze dan er toetsen mee gaan oefenen worden ze vanzelf beter in die vakken en krijgen ze betere items in het spel. Als ze genoeg geoefend hebben sluiten ze hun Edu-bot aan op een Wi-Fi hotspot en kunnen ze lekker verder met hun spelletje. De koppeling met de PC is nodig om de extra ervaring door te kunnen geven van de toetsen die de speler heeft gedaan. Het is ook nodig om het aantal kisten door te geven die ze kunnen open maken. Daarnaast geeft de PC applicatie de nieuwe geüpdatet levelniveaus door. Zo kan de robot weer verder tellen als de speler meer gaat oefenen.

## Versiebeheer Ontwerpfase

Versie	Datum	Taak	Persoon
1.0	26-02-2018	Basisopzet/layout	Gido ten Cate
1.1	15-03-2018	Basisontwerpdokument	Gido ten Cate
1.2	19-03-2018	Software uitbreiden omgezet naar word document	Bram van Gils
1.3	10-04-2018	Media Sketches ingevuld	Bram van Gils
1.4	10-04-2018	Technology Software Technology Sketches	Thijs Spapens Kieran Marriott

# Realiseren

## OIS



```
95 //Constructor
96 public MainForm()
97 {
98 }
99
100
101 //Main
102 public void Main()
103 {
104     while (ReopenForm)
105     {
106         SwitchForm(NextForm);
107     }
108 }
109
110 // Calculate Level Based on XP
111 public int XpToLevel(int SendXP)
112 {
113     int Outcome = 1;
114     for (int Level = 0; SendXP > LevelToXp(Level); Level++)
115     {
116         Outcome = Level;
117     }
118     return Outcome;
119 }
120
121 // Calculate minimum XP based on Level
122 public int LevelToXp(int SendLevel)
123 {
124     int XP = 0;
125     for (int Level = 1; Level < SendLevel; Level++)
126     {
127         XP += 1000 + (Level - 1) * (Level - 1) * 100;
128     }
129     return XP;
130 }
131
```

### Lucas Derksen (Schriftelijk response)

De methode XpToLevel wordt gebruikt om uit te rekenen hoeveel levels erbij moeten komen op basis van een hoeveelheid xp dat wordt meegegeven. De methode LevelToXp wordt in de methode XpToLevel opgeroepen om te checken hoeveel Xp er nodig is om een specifieke level omhoog te gaan (volgens mij heeft iedere level een andere waarde nodig aan xp om lvl up te gaan)

Er wordt al heel goed gebruik gemaakt van comments om de code te verduidelijken alleen wellicht mag dit ietsjes uitgebreider om voor andere makkelijker te maken wat er precies gebeurt in de methode met name de LevelToXp methode.

### Jort van Waes (Schriftelijk response)

Deze twee methods, XpToLevel() en LevelToXp() rekenen levels en xp naar elkaar om. XpToLevel neemt als input Xp en als output Level, en LeveltoXp andersom. Ook neemt XpToLevel de outcome van LevelToXp mee in zijn eigen berekening, waardoor de hoeveelheid xp die je per level hebt exponentieel groeit. Zolang je LevelToXp vóór XpToLevel uitvoert zou het qua code goed moeten komen ( als je het andersom uitvoert voor het eerst crasht het denk ik ).

Als tip, ik weet niet hoe de rest van de code eruitziet natuurlijk, maar als elk level exponentieel langer duurt om te behalen kan ik me voorstellen dat je na een tijdje tot praktisch oneindig bezig zou zijn met levels behalen. Je zou ook de xp rewards moeten scalen of steeds moeilijkere opdrachten moeten geven welke ook hogere xp rewards returnen, maar ik weet natuurlijk niet of dit al zo is. De code zelf ziet er werkend uit.

## Resultaat

Zowel Lucas als Jort begrepen dat het gaat om het omrekenen van Xp naar Level en andersom. Beide hebben het code bergrepen, hieruit kunnen we concluderen dat onze code ook duidelijk en verstaanbaar is.

Als tip heeft Lucas aangegeven dat de comments misschien uitgebreider kunnen voor de mensen die het misschien minder zullen snappen.

Jort heeft aangegeven dat, vanwege hoe de XptoLevel methode werkt, je op hogere levels heel veel moet doen om maar één level omhoog te gaan. Echter het maximale level is 10 dus hier zullen we weinig last van krijgen. Jort heeft ook aangegeven dat hij niet weet hoe de code achter het xp zelf werkt, vandaar deze tip.

## OIT

EduBot datastroom	Inhoud verzuurde bericht. Rood is karakters, Blauw is integers.	Betekenis verzuurde bericht.
Ontvangen PC->EduBot	"1"	De C# applicatie vraagt hiermee "Edubot, ben je gereed?"
Versturen EduBot->PC	"2AxpEngBxpNedCxpRekDlbCElbRFlbEGlL"	Edubot zegt met de 2 Ja ik ben er, ben jij er?" En verzuurde dan in deze string 7 integers. Het verdiende xp van Engels, Nederlands en Rekenen en het aantal Common Rare Epic en Legendary lootboxes.
Ontvangen PC->EduBot	"3AxpEngBxpNedCxpRek"	Met de 3 zegt de game "Ja, ik heb jou gegevens ontvangen. Je mag die gegevens nu wissen" en stuurt vervolgens het totale xp van Engels Nederlands en Rekenen
Versturen EduBot->PC	"4"	Met de 4 zegt Edubot "Ik heb mijn gegevens nu gewist en wacht op verdere instructies"
Ontvangen EduBot->PC	"5"	De 5 zegt nu "Ik ben nu klaar met quests,

		je mag weer quizjes geven.”
--	--	-----------------------------

De code die gemaakt is voor de EduBot is gesplitst over meerdere bestanden om de code overzichtelijker te maken voor de programmeur. Door de scheiding over verschillende bestanden zou dit tijd besparen wanneer iets aangepast zou moeten worden aangezien de code is verspreid over meerdere bestanden i.p.v. een gigantische file. Dit zou ook de compile time kunnen verkorten als je een slimme compiler hebt die alleen de aangepaste files recompiled.

- Main.c
- GUI.c
- Subjects.c
- EV3Mailbox.c

Naam	Beschrijving
Main.c	Hier in staat de main loop waaruit de rest van de code wordt aangeroepen. Hier staan tevens de FormatForPC en ParseString functie in de data van de PC omzet of zelf omzet naar een bruikbaar formaat.
GUI.c	Hierin staat alle code die te maken heeft met de GUI. Hieronder valt het navigeren in of naar ander menus.
Subjects.c	Hierin staat de data voor de vragen en antwoorden. Dit bestand wordt geinclude in de GUI.c zodat de code toegang heeft tot de data.
EV3Mailbox.c	Hierin staat alle code die de communicatie tussen de EduBot en de PC werkend maakt.

## OIM

### Scenario

EduBot zorgt ervoor dat kinderen kunnen leren on the go. Door gamification en gepatenteerde leermethodes van Malmberg maakt de EduBot het leertraject effectief en leuk voor uw kinderen. Het verschil met andere educatieve robots is dat wij de nieuwste en beste leermethodes kunnen leveren aan kinderen.

## Persona

Foto:	
Naam:	Rick
Leeftijd:	7
Educatie	Basisschool (Groep 5)
Doel:	Spelenderwijs door gamificatie kennis eigen maken.

## User story

### Stappen

1. Rick pakt de Edubot op.
2. Rick selecteert het vak wat hij wil leren.
3. Rick leert.
4. Rick heeft het level voltooid en krijgt zijn beloning.
5. Rick verbind zijn EduBot met de pc-applicatie en zijn beloning wordt naar de pc-applicatie gestuurd.
6. Rick speelt de game met zijn verdiende beloningen verkregen van de EduBot.

# EDUBOT



1. Rick pakt de Edubot op.



2. Rick selecteert het vak wat hij wil leren.



3. Rick leert.



4. Rick heeft het level voltooid en krijgt zijn beloning.

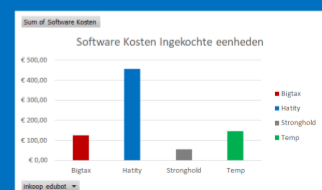
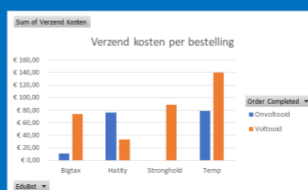
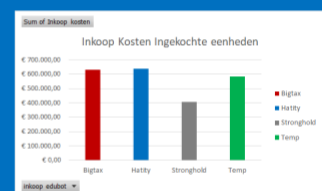
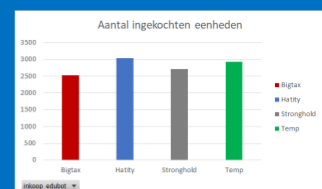
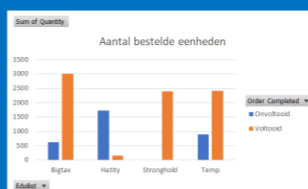
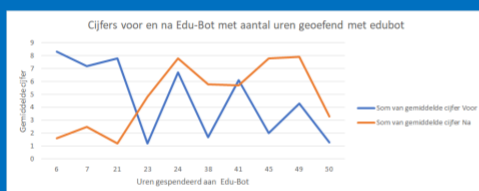


5. Rick verbind zijn EduBot met de pc-applicatie en zijn beloning wordt naar de pc-applicatie gestuurd.



6. Rick speelt de game met zijn verdiende beloningen verkregen van de EduBot.

## OIB





## Feedback

Ronald Marcelis – Technology

Storyboard kan meer worden omschreven in een use case vorm.

Ronald Marcelis – Technology – 09-04-2018

Erasedisplay en bouw dan het hele scherm weer op. Dit is veel makkelijker dan bepaalde plaatsen te updaten telkens.

Ronald Marcelis – Technology -09-04-2018

Geen multidimensionale arrays gebruiken maar een dimensionale arrays gebruiken waaruit je volgens een bepaalde formule bepaald welk item je er uit haalt.

## Versiebeheer Realisatiefase

Versie	Datum	Taak	Persoon
1.0	09-04-2018	Basisopzet/Layout	Bram van Gils
1.1	11-04-2018	Document en uitwerking gemaakt en in het document samengevoegd	Kieran Marriot Thijs Spapens

## Testen

### OIS

Test	Metode	Uitkomst
het programma kan met de robot verbinden	We runnen de robot en de applicatie en connecten de robot op een hotspot. we gaan de robot proberen te connecten	de robot heeft af en toe problemen met de hotspot vinden maar als hij hem vindt kan hij moeiteloos verbinden
het programma kan data ontvangen van de robot	We kijken in Visual Studio en in ROBOTC met breakpoint en we kijken of de data opgevraagd kan worden	Het programma kan data ontvangen van de robot
het programma kan data naar de robot sturen	We doen hetzelfde als bij het ontvangen alleen kijken we nu op de robot of hij data ontvangt	De robot kan de data van de applicatie ontvangen
het programma kan de variabelen, vanuit de robot gestuurd, gebruiken	We connecten de robot met het programma en sturen variabelen door via de robot om te zien of het programma reageert op de data	de applicatie kan op de doorgestuurde variabelen reageren door andere variabelen terug te sturen
het programma navigeert goed	We runnen de applicatie en klikken op alle knoppen die naar andere forms gaan	De navigatie werkt prima met als enige problemen dat alt+f4 en het kruisje niet werken, om af te sluiten moet je eerst naar het hoofdmenu gaan.
Je kan een spel spelen met het programma	we kijken of we via het questlog form een quest kunnen gaan spelen	het questlog werkt nog niet en er is nog geen quest toegevoegd
het spel functioneert goed	we kijken of alle knoppen en opties doen wat ze horen te doen	er is nog geen quest te doen in het spel dus dit kunnen we nog niet testen

### OIT

Test	Methode	Uitkomst
kan de robot data doorsturen	We houden visual studio en ROBOTC naast elkaar met breakpoints en runnen de	

	applicatie om te zien of de data doorgestuurd kan worden naar C#	
Kan de robot data ontvangen en beslissen welke actie hij moet ondernemen	We doen hetzelfde met de data versturen alleen kijken we nu of de robot de data ontvangt en of de de robot iets met die data doet	Ja we hebben op de robot kunnen zien dat de data van de PC is doorgestuurd en hij kan ook dingen doen met de ontvangen data in de code
Werkt alle navigatie in de robot	We runnen de robot en we navigeren naar alles om te checken of het fatsoenlijk werkt	er zijn problemen met het openen van toetsen, het maken van toetsen en de instellingen
Kan de robot toetsen afleggen en op verschillende niveau's	We runnen de robot en proberen om een toets te maken op verschillende niveau's	De robot kan geen toetsen afleggen want hij skipt heel de toets en komt weer terecht op het niveau menu. hij kan wel meerdere niveau's doen
Kan de robot willekeurige toets vragen stellen	we doen dezelfde toets een paar keer en kijken of de vragen willekeurig zijn	de robot skipt over de toetsen heen en we kunnen daarom nog geen test doen

OIM

STATUS ROBOT: VERBONDEN

VERBINDEN

LOS KOPPELEN

UPLOAD DATA

WIS DATA

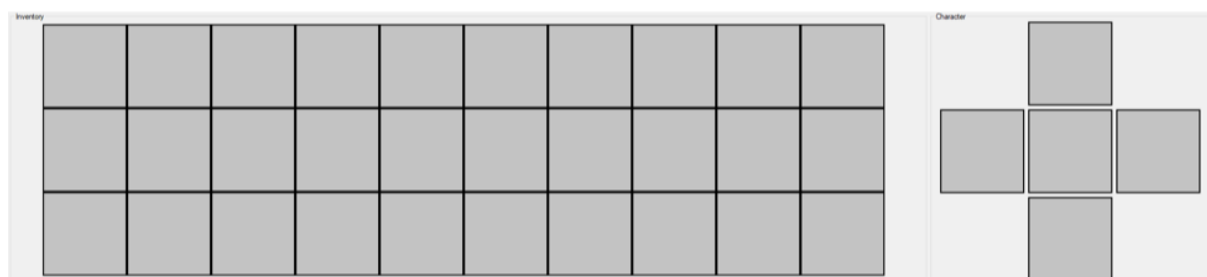
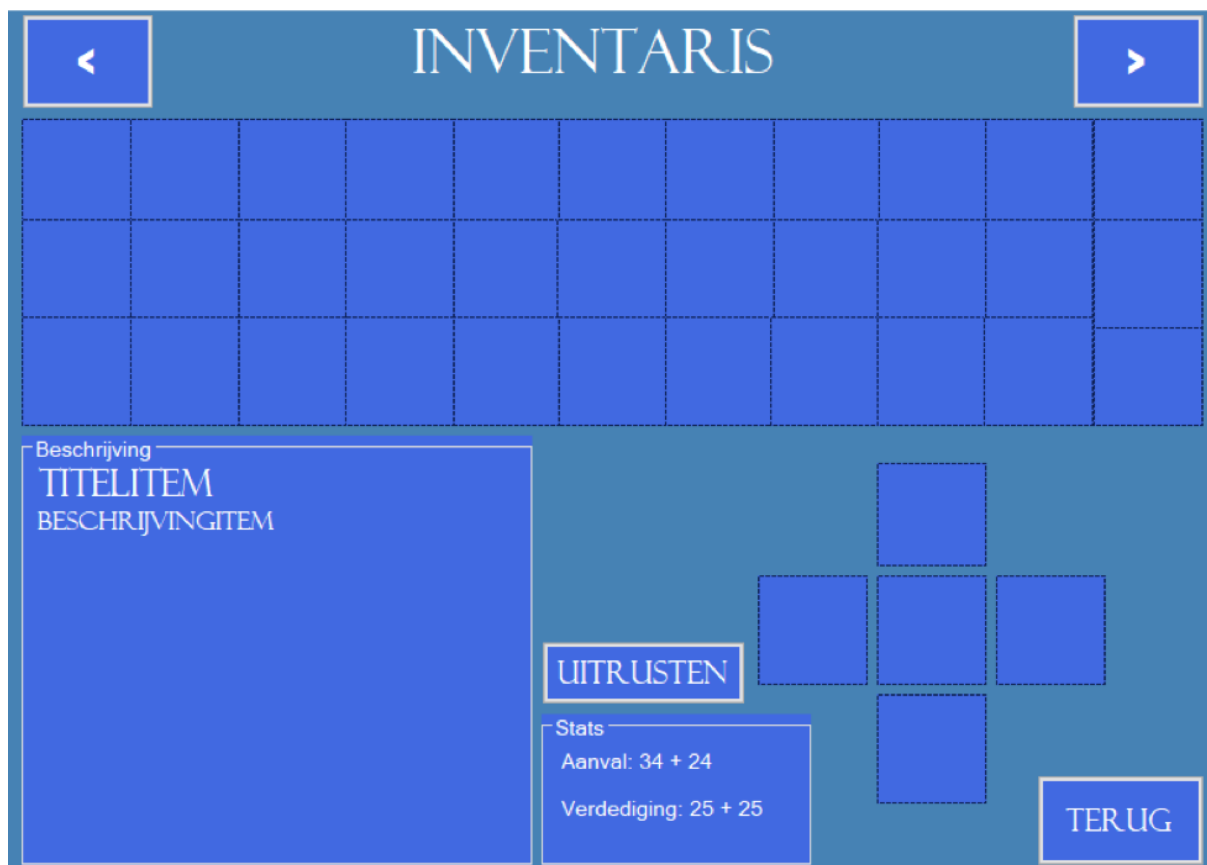
BACK-UP DATA

LISTBOX ROBOT INSTELLINGEN

MENU



Om te beginnen hebben we een grote aanpassing gemaakt aan de gezondheids, energie en Manabar. Dit kwam omdat we geen progressbar konden vinden die verticaal ging. Hierdoor hebben we maar gekozen om een horizontale bar te doen en ook elke bar een aparte kleur te geven.



Het tweede wat we drastisch hebben moeten veranderen is de inventaris. Dit is omdat het nieuwe design veel gemakkelijker te coderen was en ook simpeler is voor kinderen. Omdat dit design zo eenvoudig is snappen de kinderen dit waarschijnlijk veel beter als het eerdere design.

STATUS ROBOT: VERBONDEN

VERBINDEN LOS KOPPELEN

UPLOAD DATA

WIS DATA

BACK-UP DATA

LISTBOX ROBOTINSTELLINGEN

MENU

IP ADDRES VAN EV3

LOS KOPPELEN VERBINDEN

receivedMessagesListBox

UPLOAD DATA NAAR DE EV3

LEES INBOX UPLOAD

TERUG

WIS DATA

We hebben ook dit menuscherm flink moeten aanpakken omdat het eerdere design niet alles had om fatsoenlijk verbinding te kunnen maken met de robot. Door het nieuwe design kunnen we nu wel goed met de robot verbinden en is dat probleem ook weer opgelost. Daarnaast is het niet echt veel moeilijker geworden om te snappen dan het oude design.

#### OIB

Onze doelen voor ons project waren dat kinderen leren leuker zouden vinden en daardoor ook slimmer zouden worden. Om dit aan te tonen hebben we in ons management dashboard een grafiek gezet voor resultaten voor en na de introductie Edu-Bot samen met de uren gespendeerd met het oefenen met de Edu-Bot. Hierin kan je zien dat gemiddeld klassen die meer tijd stoppen met oefenen met de Edu-Bot hoger scoren dan de klassen die minder tijd stoppen in oefenen. De klas die juist minder had geoefend heeft gemiddeld veel lager gescoord op de toets. Dit geeft dus aan dat de kinderen die meer oefenen ook beter worden en dus ook hoger scoren. Hierdoor kunnen we concluderen dat ons product ook echt kan helpen om schoolleerlingen hogere cijfers te halen.

## Feedback

Peter Dingemans - 11-4-2018

Voor media en business kan je geen tests doen dus hoeven die niet in dit document.

Kees Velthuijs - 11-4-2018

Het is de bedoeling dat je in je test rapportage gaat uitleggen dat je je doel in je business requirements kan aantonen in je management dashboard. Als je doelt op kinderen slimmer maken, vertel dan dat dit aan te geven is via je management dashboard.

## Versiebeheer Testfase

Versie	Datum	Taak	Persoon
1.0	10-04-2018	Basisopzet/layout	Bram van Gils
1.1	10-04-2018	Basisvragen toegevoegd	Thijs Spapens
1.2	11-04-2018	Testrapportage	Thijs Spapens

## Implementatie

Het doel van de implementatiefase is het verwezenlijken, testen, en documenteren van onze gemaakte producten voor het P1 event. Om een goede indruk te maken op het P1 event implementeren wij volgende punten voor het P1 event. De reden hiervoor is omdat dit zo ons een betere indruk geeft bij voor de beoordeling maar ook op het publiek dat onze stand bezoekt.

- Houd de stand netjes
- Geen medewerkers op bezoekersstoelen
- Geen halfvolle glazen, kopjes, schotels en plastic bekers laten slingeren
- Geen rondslingerende bagage (koffertjes, tassen)
- Geen persoonlijke eigendommen in het zicht

### OIS

#### Prototype

Voor het OIS tijdens het P1 event laten wij een prototype zien aan bezoekers. Dit prototype geeft de bezoeker een inzicht over hoe het prototype interact met zijn omgeving.

#### Flowchart

De flowchart later wij zien tijdens het P1 event om de bezoekers een beter beeld te geven van de EduBot en zijn activiteiten. In de flowchart kunnen de bezoekers in een makkelijk overzicht zien hoe de flow van de programma loopt.

### OIT

#### Prototype

Het OIT prototype voor het P1 event laten wij zien zodat de bezoeker een systeem heeft waarmee hij kan interacteren. Dit geeft de bezoeker een glimps van hoe een echte EduBot eruit zou kunnen zien.

### OIM

Aan de hand van de ontwerpschetsen implementeren wij de producten van project EduBot. Deze schetsen geven een beeld over hoe

#### Presentatie

Met de presentatie willen wij het publiek informeren belangrijke informatie laten zien over de EduBot zodat het publiek een beeld krijgt waarom ze de EduBot eventueel zouden willen aanschaffen en hoe hij werkt.

#### User story

Met de user story's willen wij het publiek laten zien hoe bepaalde features zouden werken van de EduBot. Deze features zouden niet per se al werkend hoeven te zijn in het prototype. Dit zouden wij eventueel opvangen door gebruik te maken van de Wizard of Oz techniek (Wikipedia, sd).

## OIB

### Business Model Canvas

Door het Business Model Canvas te laten zien aan het publiek willen wij laten zien hoe de EduBot zich onderscheidt van andere soortgelijke producten maar ook voor die het bedoeld is en ook via welke kanalen de EduBot geld zou verdienen.

## Feedback

Peter Dingemans – 11-04-2018

Het implementatie document moet beschrijven hoe de gemaakte producten geïmplementeerd/gemanaged zijn. Dit betekent dat voor Technology wij het zouden kunnen hebben over hoe wij onze code hebben verspreid en geïncludeerd doormiddel van de #include preprocessor directive.

Peter Dingemans – 11-04-2018

Voor de implementatiefase hoef je niet veel in de documentatie van de proftaak te zetten. Je moet bij elk vak vertellen op het P1 event doen wij x voor vak y. Dit wordt eventueel onderbouwd door argumenten.

### Versiebeheer Implementatiefase

Versie	Datum	Taak	Persoon
1.0	11-04-2018	Implementatiefase	Gido ten Cate