



Testnet

ZomerWorkshops 2016

*“Specification by Example met Robots
Gedrag van robot specificeren”*



Agenda

- Introductie Specification by Example
- Introductie mBot
- Aan de slag
- Take aways

The header features a rounded rectangle on the left containing the name 'bartosz' in a bold, lowercase, purple sans-serif font. To the right of this rectangle is a horizontal strip with a blurred background of a person's face. Overlaid on this strip are several small, semi-transparent colored squares in shades of purple, blue, orange, grey, green, and light blue.

bartosz

Introductie

Specification by Example

Testing

Learning by experimenting, including study, questioning, modeling, observation, inference, etc.

Human checking

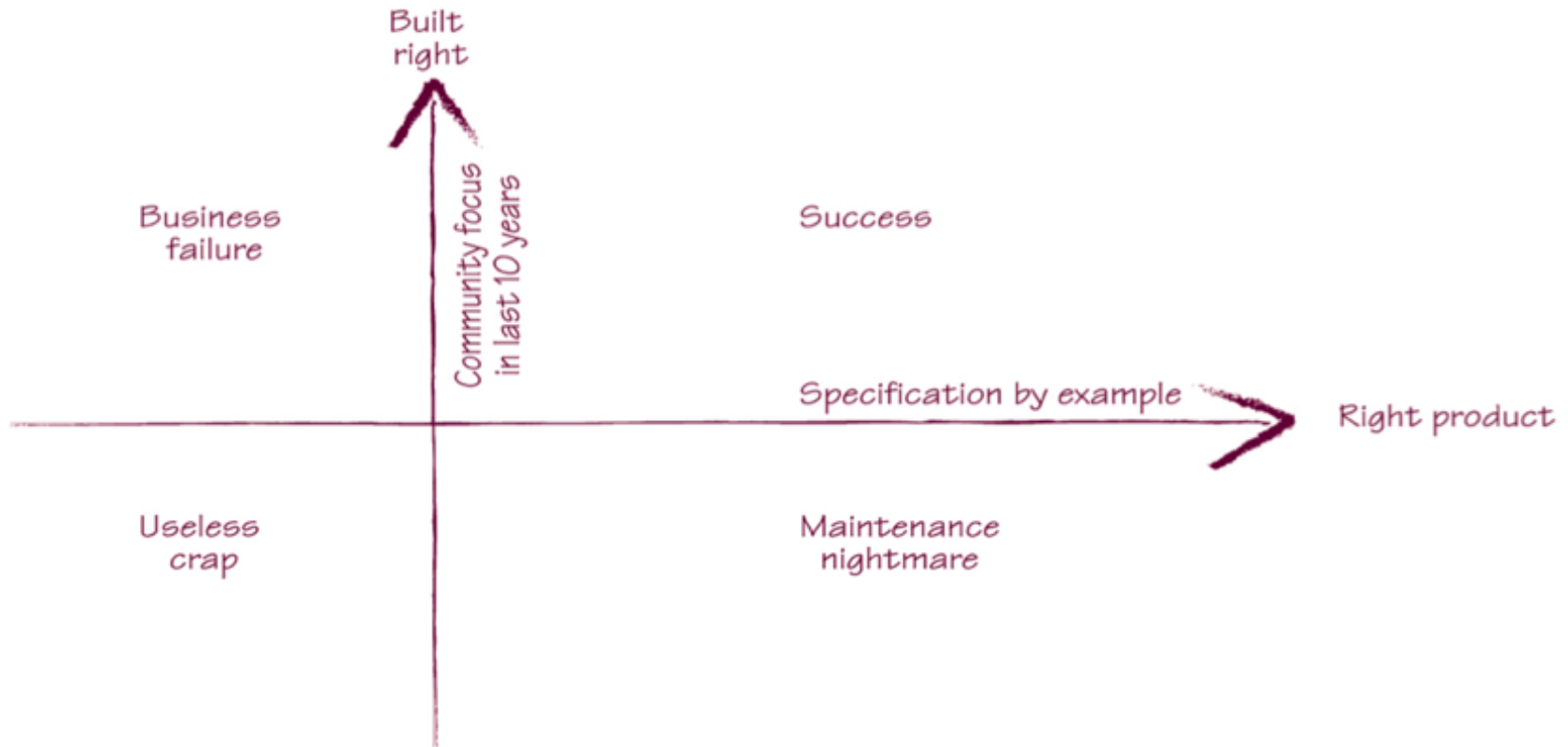
Machine checking

- Focus in the last 10-15 years:

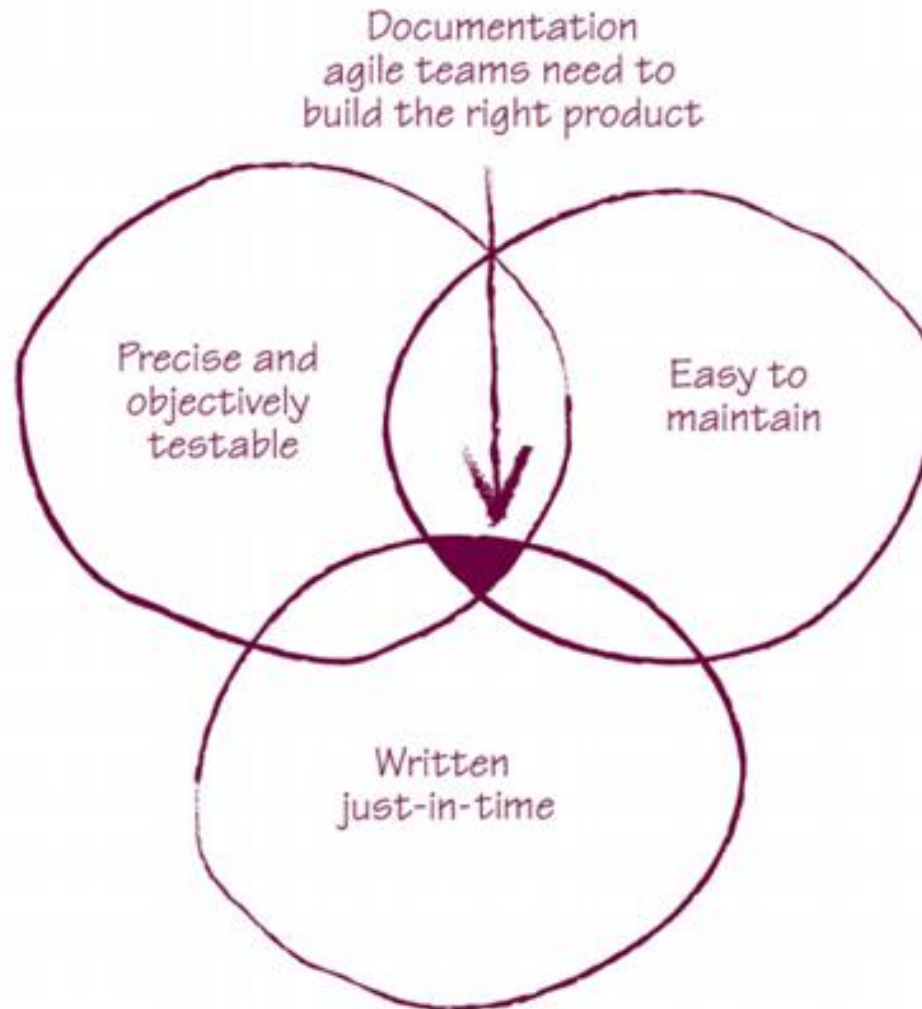
Building the product right

- In order to succeed AND survive:

Building the right product



- "To be succesfull at a regular basis we as software professionals need to rethink our specification and development strategies"- Some smart guy



- Specifications are important to anyone trying to understand the system
- Specifications have to be written in a language that everybody understands: **BUSINESS LANGUAGE**

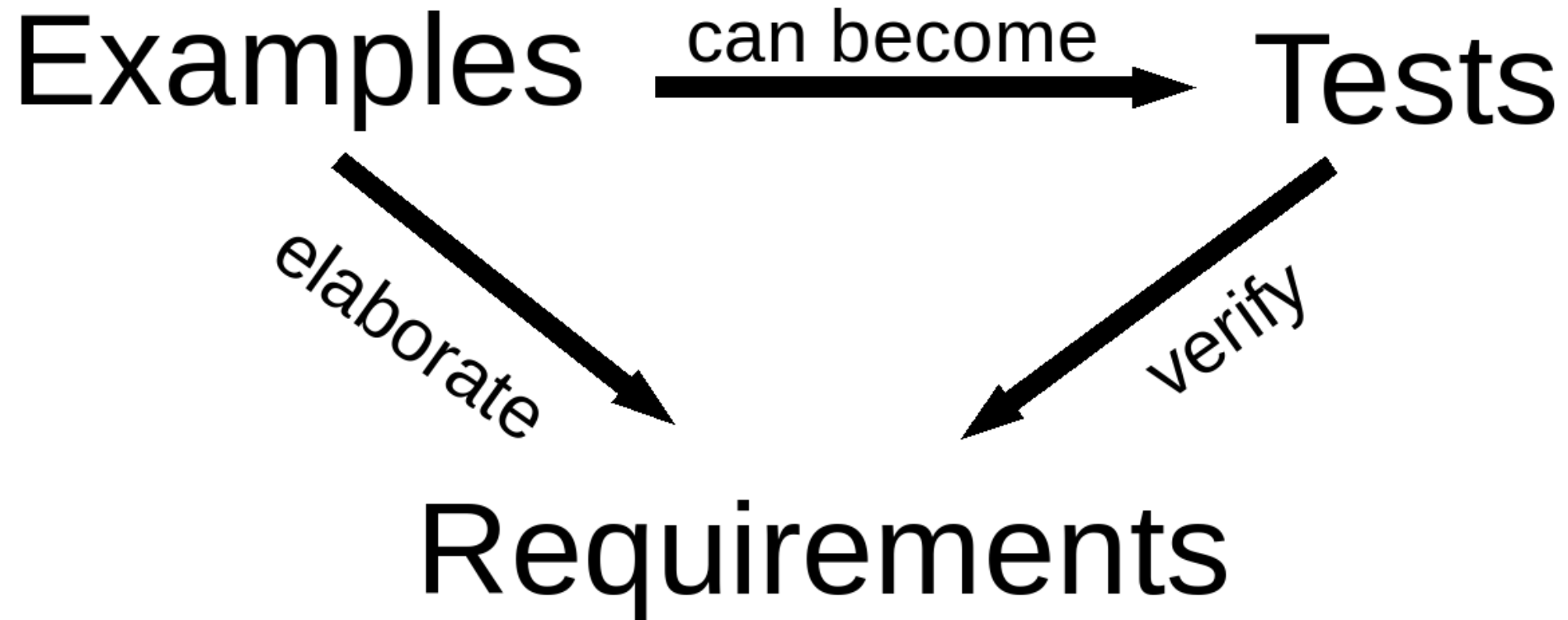
NO

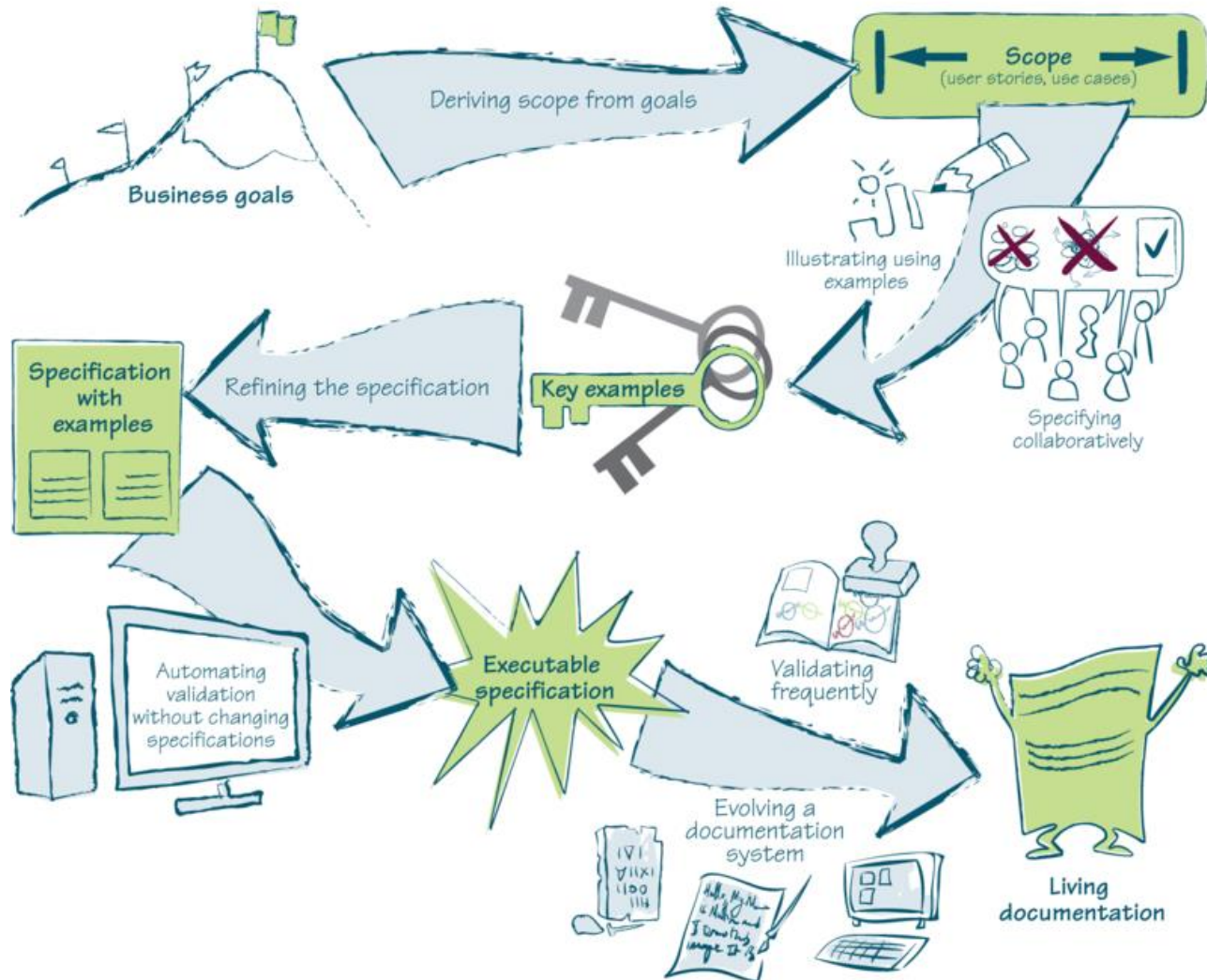
When **user** logs on with valid credentials, and empty **panel** is displayed.

YES

When **player** logs on with valid credentials, and empty **board game** is displayed.

- People often use examples as they explain complex topics
- Why don't add these examples to our specifications?

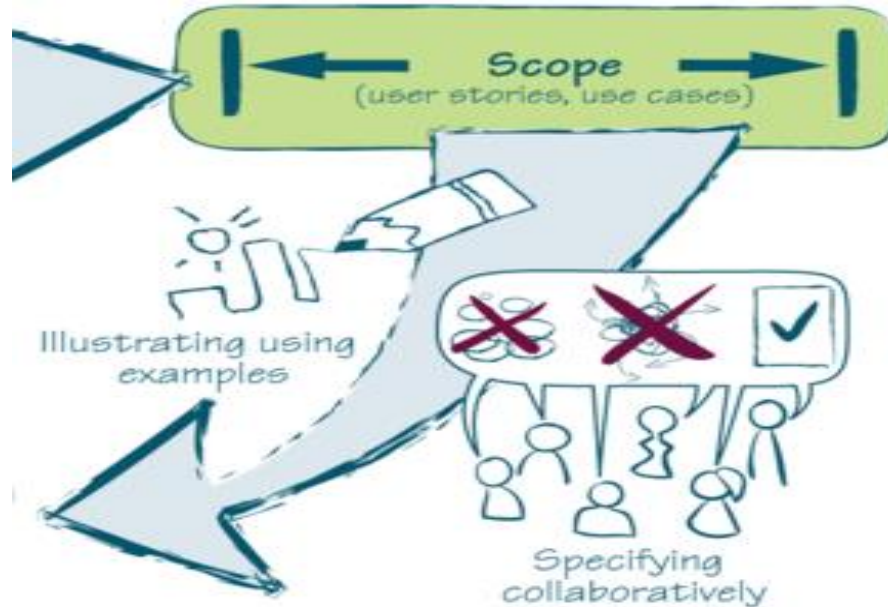






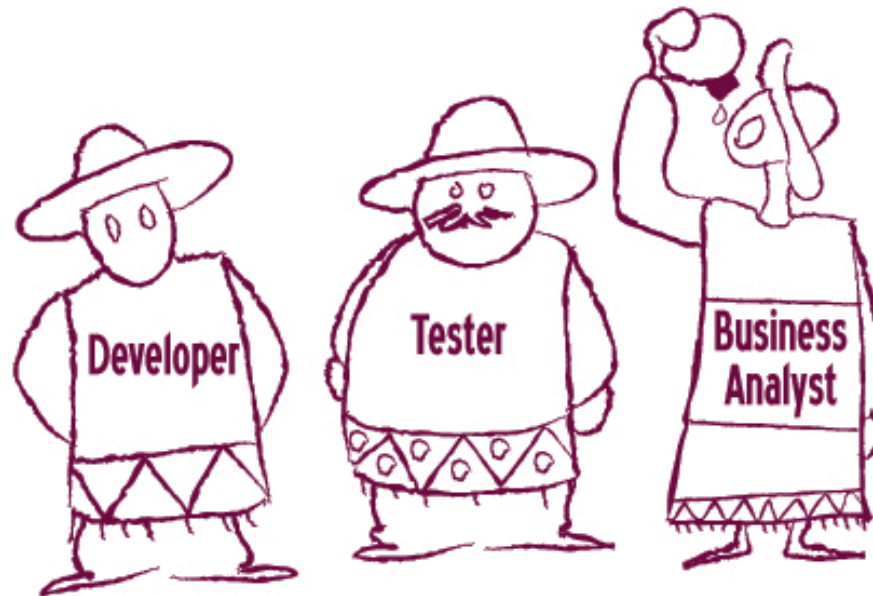
- ‘Why’ and ‘how’
 - Understanding why something is needed by whom is crucial to evaluating a suggested solution
 - Understand where value is coming from
 - Understand what output the customer expects
 - Asking about usefulness and alternatives
- Make sure teams deliver complete features

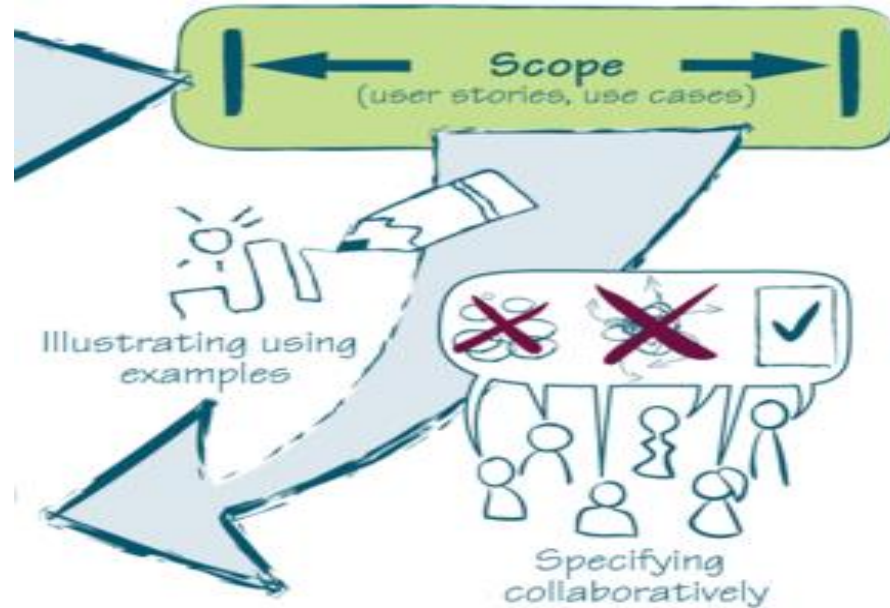
- Building shared understanding of what needs to be done



- Helps to produce specifications that are easy to understand
- Helps to write tests that are easy to maintain

Three amigos





- Examples from specification through development to testing
- Shared understanding
- Realistic, precise, complete, easy to understand
- Helps spotting inconsistencies and functional gaps

- Get basic examples direct from customers
- Use real life data:
 - Clarifies understanding of the domain
 - Understanding of realistic usage
 - Testing with real data is eating your own dogfood
 - Helps users/customers to validate the requirement



A good specification, with examples, is effectively an acceptance test for the described functionality

- Descriptive title and goal:
 - A set of examples won't do
 - A description of the business rule won't do
 - Both comprise 80% of the content. Working them out together clarifies title, goal and examples
 - Validate the specification: "Show and keep quiet"

Feature: Some terse yet descriptive text of what is desired
Textual description of the business value of this feature
Business rules that govern the scope of the feature
Additional information that makes it easier to understand

Scenario: Some determinable business situation

Given some precondition

When some action by the actor

Then some testable outcome is achieved

Gherkin syntax:

- You can combine several Given and When statements by using **And**
- You can use **But** to add more conditions

```
Scenario: Multiple Givens
```

```
  Given one thing
```

```
    And another thing
```

```
    And yet another thing
```

```
  When I open my eyes
```

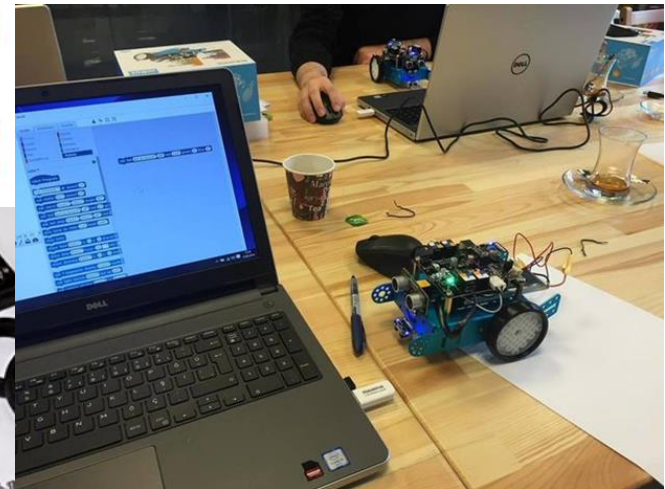
```
  Then I see something
```

```
    But I don't see something else
```

bartosz

Introductie mBot





“De mBot is een speciaal voor het onderwijs ontwikkelde robot. Met de mBot kunnen leerlingen stap voor stap de beginselen leren van programmeren en robotica. Door het programmeren te koppelen aan robotica zien de leerlingen niet enkel op het scherm wat hun programma doet maar gebeurt er echt iets met hun robot.”



Arduino is een opensource-computerplatform dat is opgebouwd rond de ATmega168-microcontroller van Atmel en het softwareontwikkelplatform Processing. Dit platform is bedoeld voor hobbyisten, artiesten, kunstenaars en iedereen die geïnteresseerd is in het maken en ontwerpen van slimme en creatieve objecten die kunnen reageren op hun omgeving.

Met Arduino is het mogelijk apparaten en objecten te creëren die reageren op hun omgeving door middel van digitale en analoge inputsignalen. Op basis van deze input kan een Arduinoschakeling autonome actie initiëren door het afgeven van digitale en analoge outputsignalen. Input kan bijvoorbeeld worden gegenereerd door schakelaars, lichtsensoren, bewegingsmeters, afstandsmeters, temperatuursensoren, of op basis van commando's afkomstig van internet, een radiomodule of een ander apparaat met een seriële interface. Outputsignalen kunnen bijvoorbeeld motoren, lampjes, pompjes en beeldschermen aansturen, maar ook input genereren voor een andere Arduinomodule.



```
Arduino - 0011 Alpha
File Edit Sketch Tools Help
Blink
/*
 * Blink
 *
 * The basic Arduino example. Turns on an LED on for one second,
 * then off for one second, and so on... We use pin 13 because,
 * depending on your Arduino board, it has either a built-in LED
 * or a built-in resistor so that you need only an LED.
 *
 * http://www.arduino.cc/en/Tutorial/Blink
 */

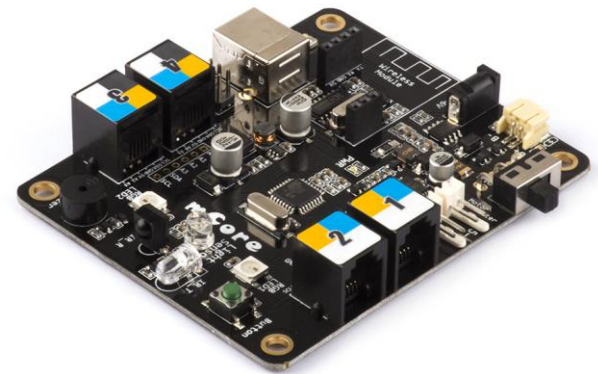
int ledPin = 13;           // LED connected to digital pin 13

void setup()               // run once, when the sketch starts
{
  pinMode(ledPin, OUTPUT); // sets the digital pin as output
}

void loop()                // run over and over again
{
  digitalWrite(ledPin, HIGH); // sets the LED on
  delay(1000);                // waits for a second
  digitalWrite(ledPin, LOW);  // sets the LED off
  delay(1000);                // waits for a second
}

Done compiling.

Binary sketch size: 1098 bytes (of a 14336 byte maximum)
22
```

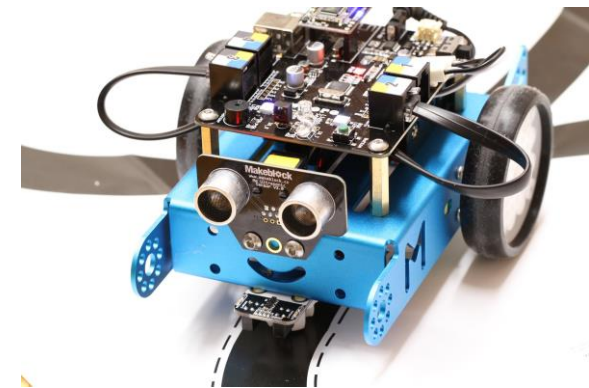


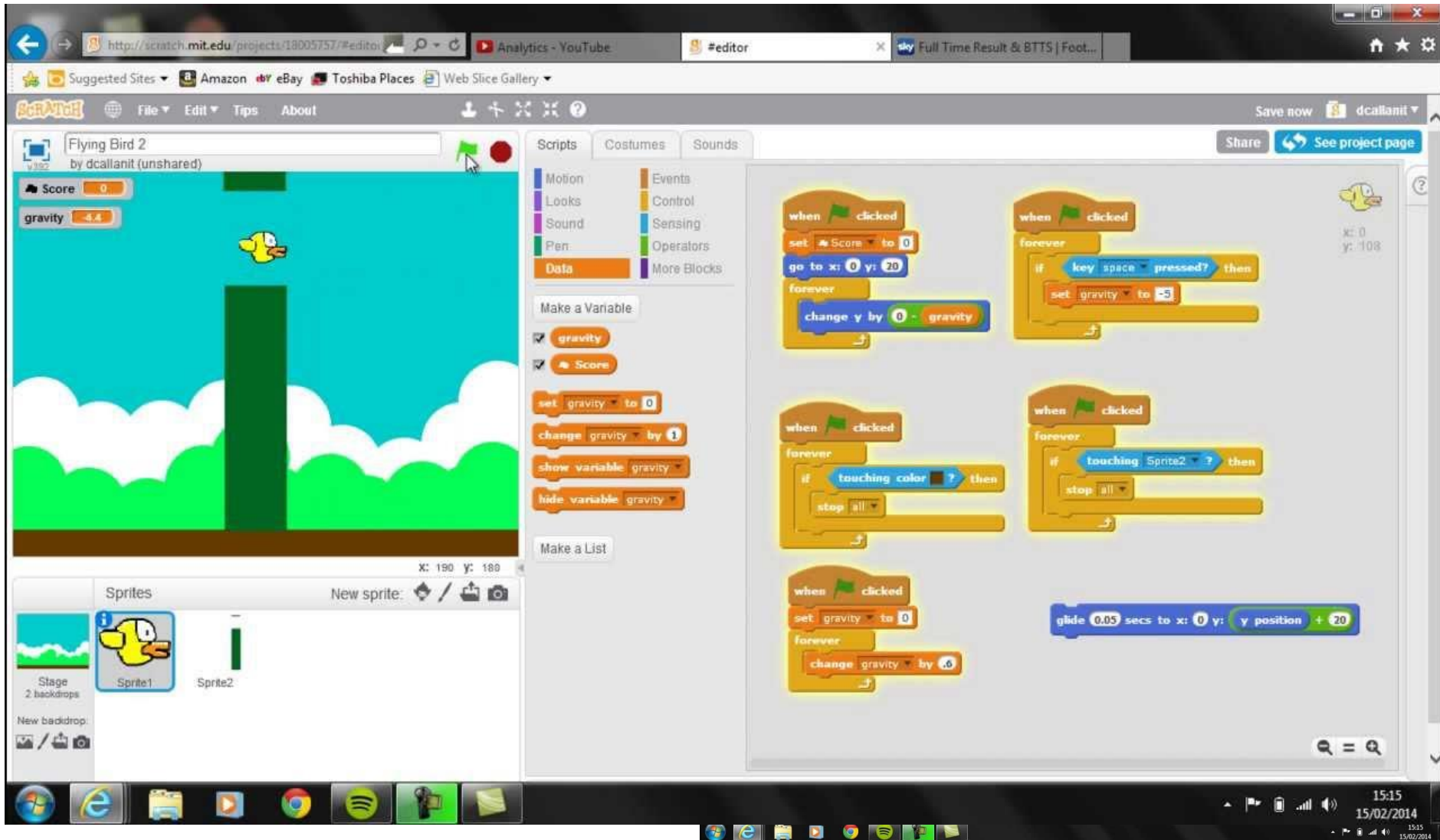
...die reageren op hun omgeving door middel van digitale en analoge inputsignalen...

- Knop
- IR Sensor
- Afstandsensor (Ultrasonische module)
- Licht/Donker sensor
- Lijnvolg sensor

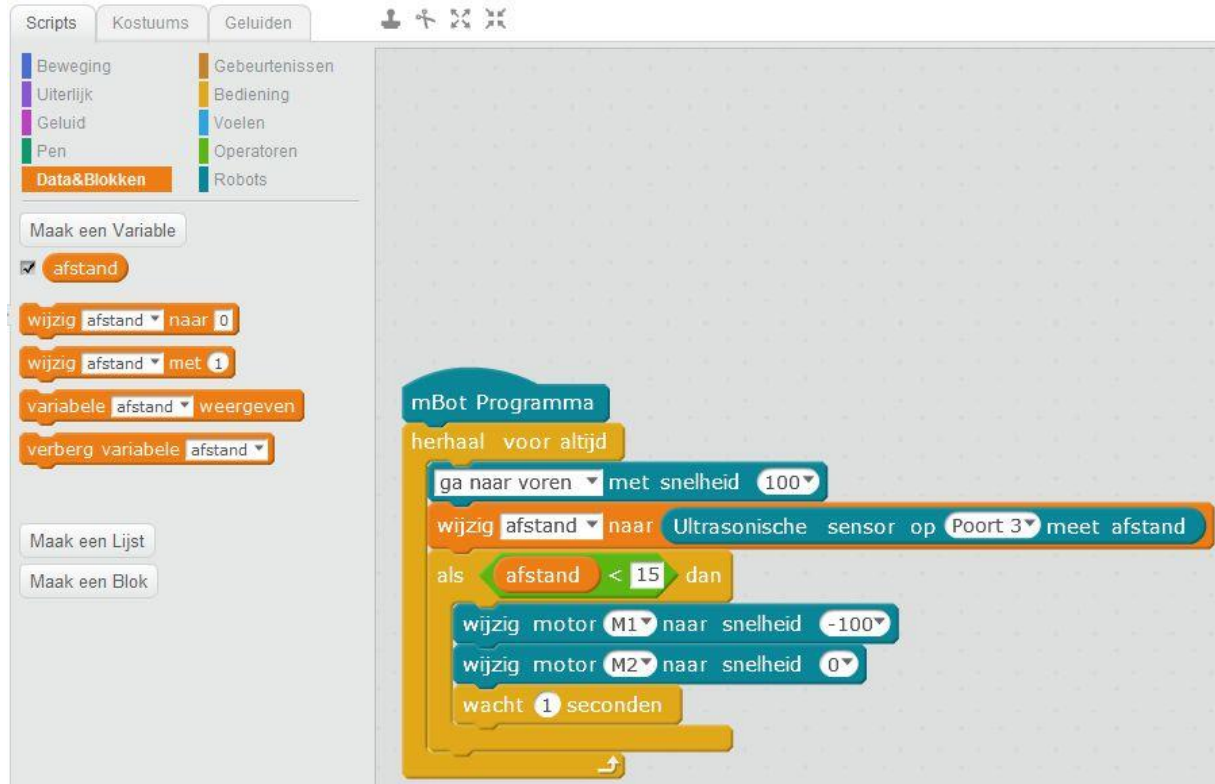
Outputsignalen kunnen bijvoorbeeld motoren, lampjes, pompjes en beeldschermen aansturen...

- Twee motoren
- Twee RGB ledlampjes
- Geluid





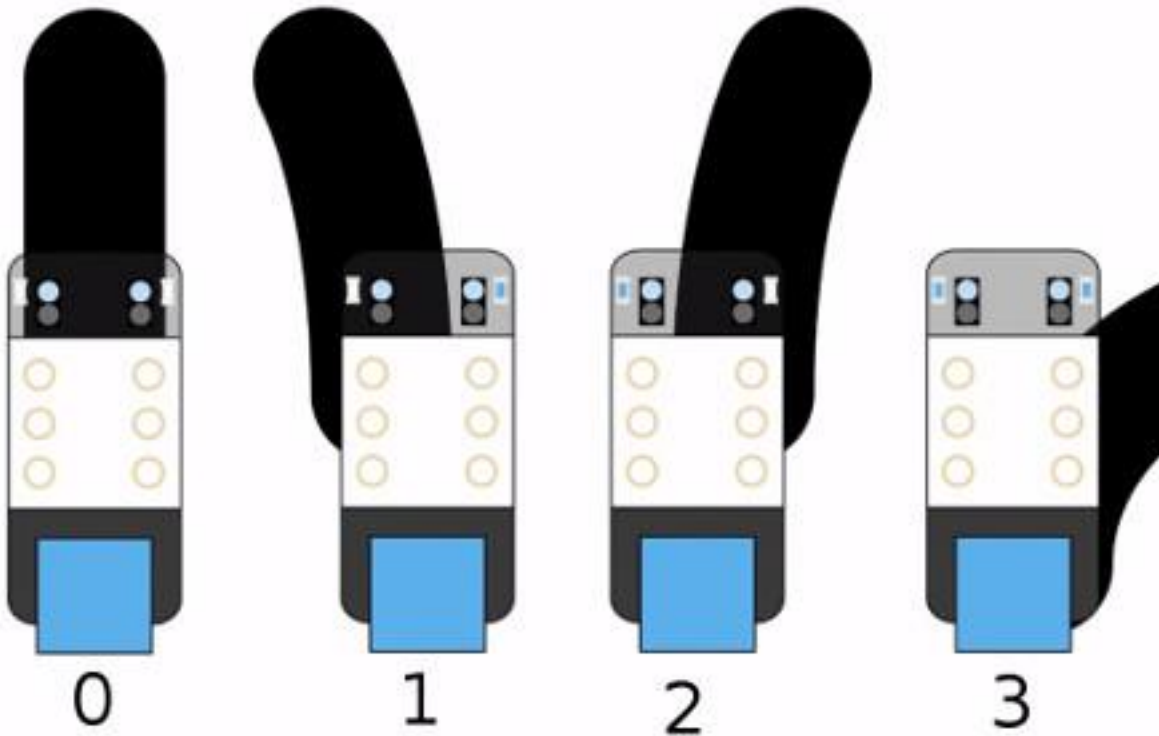
mBlock is een uitbreiding van Scratch. Er zijn blokken in toegevoegd waarmee de mBot aangestuurd kan worden. Scratch code wordt vertaald naar Arduino code. Via de mBlock EDI wordt deze code naar de mBot gestuurd.



The screenshot shows the mBlock software interface, which is a Scratch-based environment. The left sidebar contains various block categories: Beweging, Uiterlijk, Geluid, Pen, Data&Blokken (highlighted), Gebeurtenissen, Bediening, Voelen, Operatoren, and Robots. The main workspace displays a script for an mBot:

```
mBot Programma
herhaal voor altijd
  ga naar voren met snelheid 100
  wijzig afstand naar Ultrasonische sensor op Poort 3 meet afstand
  als afstand < 15 dan
    wijzig motor M1 naar snelheid -100
    wijzig motor M2 naar snelheid 0
    wacht 1 seconden
```





- 0 - op de lijn
- 1 - rechts van de lijn
- 2 - links van de lijn
- 3 - niet op de lijn

```

mBot Programma
herhaal voor altijd
  wijzig lijn naar Lijnvolg module op Poort 2
  als lijn = 0 dan
    ga naar voren met snelheid 175
  anders
    als lijn = 1 dan
      wijzig motor M1 naar snelheid 50
      wijzig motor M2 naar snelheid 175
    anders
      als lijn = 2 dan
        wijzig motor M1 naar snelheid 175
        wijzig motor M2 naar snelheid 50
      anders
        ga naar achteren met snelheid 175
  
```

Wat gaan we doen?

Als Robot

*wil ik zo snel mogelijk het parcours afleggen
Zodat ik in de paddock kan gaan rusten*



Wat gaan we doen?

- Vijf groepen: Voorbeelden van gedrag beschrijven*
- Iedereen: Bespreken van de voorbeelden*
- Vijf groepen: Werkend maken van Scratch scenario's*
- Vijf groepen: Wedstrijd: Gehele parcours (scenario's samenvoegen)*



Voorbeelden van gedrag beschrijven

- 1. Knip de baan op in stukken*
- 2. Gebruik post-it blaadjes om voorbeelden te verzinnen*
- 3. Werk de voorbeelden uit in Gegeven...Als...Dan...*



Wat gaan we doen?

Bespreken van de voorbeelden



Functionaliteit: mBot volg de lijn

Scenario: mBot volgt een rechte lijn

Gegeven dat de mBot de lijn volgt

Als de mBot op de lijn blijft

Dan gaat de mBot rechtdoor

Scenario: mBot wijkt af aan de rechterkant van de lijn

Gegeven dat de mBot de lijn volgt

Als de mBot aan de rechterkant van de lijn afwijkt

Dan stuurt de mBot bij naar links

Scenario: mBot wijkt af aan de linkerkant van de lijn

Gegeven dat de mBot de lijn volgt

Als de mBot aan de linkerkant van de lijn afwijkt

Dan stuurt de mBot bij naar rechts

Functionaliteit: mBot neemt bocht naar rechts

Scenario: mBot neemt een bocht naar rechts

Gegeven dat de mBot de lijn volgt

Als de mBot de lijn helemaal niet meer ziet

En er geen muur op een afstand van minder dan 20
cm staat

Dan neemt de mBot een bocht naar rechts

Functionaliteit: mBot neemt bocht naar links

Scenario: mBot neemt een bocht naar links

Gegeven dat de mBot de lijn volgt

Als de mBot de lijn helemaal niet meer ziet

En er wel een muur op een afstand van minder dan 20
cm staat

Dan neemt de mBot een bocht naar links

Functionaliteit: mBot bereikt paddock

Scenario: mBot bereikt de paddock en zet zijn motoren uit

Gegeven dat de mBot de lijn volgt

Als het donker wordt

Dan stopt de mBot met bewegen

Wat gaan we doen?

*Werkend maken van Scratch
scenario's*



Ga aan de slag met de mBot en Scratch op de PC.

Maak de vier stukken code werkend (in map 'Oefen scenarios')

Wij hebben al wat voorbereid maar... het bevat fouten.

Je kan de uitgeprinte 'mini' parcours stukken gebruiken.

Wat gaan we doen?

*Wedstrijd:
Gehele parcours*



Maak van de 4 stukken correcte code een werkend geheel waarmee het gehele parcours gedaan kan worden.

Je krijgt de uitwerkingen van de vier verschillende stukken op papier



- Bereik een gedeeld beeld over gedrag, niet over de implementatie
- Laat implementatie niet terugkomen in je scenario's
- BDD & SbE gaat niet primair over testen
- Tools als Cucumber en Fitnesse gaan niet over testautomatisering

bartosz

Vragen?







Parcours op
papier



Scratch Code