

leren. durven. doen.



# SOFTWARE TESTEN – OP BASIS VAN RISICO

*Risk Management – Risk Based Software Testing*  
2 werkuur

Hoofdtitel: Software Testen

Subtitel: Risk Management – Risk Based Software Testing

Auteur / docent: Vincent Van De Walle

Versie: Gebruikt tijdens opleiding

© Syntra West

*De docent/auteur en Syntra West behouden zich het recht voor de inhoud, het beeldmateriaal en alle verdere informatie in deze PowerPoint zonder voorafgaand bericht aan te passen of te verbeteren.*

*Deze PowerPoint is met zorg samengesteld, waarbij alles werd in het werk gesteld om de juistheid van de inhoud te verzekeren, niettemin wordt deze PowerPoint ter beschikking gesteld zonder enige garantie.*

*De docent/auteur en Syntra West kunnen niet verantwoordelijk noch aansprakelijk gesteld worden voor eventuele materiële schade of andere vergissingen die voortvloeien uit het gebruik van de informatie uit deze PowerPoint of voor eventuele fouten of vergissingen in de PowerPoint. Dit cursusmateriaal is auteursrechtelijk beschermd.*

**Alles** uit deze uitgave mag worden verveelvoudigd en/of openbaar gemaakt door middel van druk, fotokopie, elektronisch of op welke andere wijze ook, of in een retrieval systeem worden opgeslagen, zonder voorafgaandelijke schriftelijke toestemming van de docent/auteur.

**Niets** uit deze uitgave mag worden verveelvoudigd en/of openbaar gemaakt door middel van druk, fotokopie, elektronisch of op welke andere wijze ook, of in een retrieval systeem worden opgeslagen, zonder voorafgaandelijke schriftelijke toestemming van de Syntra West.

leren. durven. doen.



# VINCENT VAN DE WALLE

*Vincent@CopyPaste.be*

*VincentVanDeWalle@digi4u.be*

+32 (0) 475 / 97.63.80

# Als je weet dat je niet alles kunt testen ...

- **RISICO INCALCULEREN.**
  - Voorbeeld van een online webshop.
  - Stel je kunt alles doen, behalve betalen.
- **MET ANDERE WOORDEN.**
  - We gaan moeten bepalen wat we eerst doen.
- **AGILE.**
  - De dingen goed doen.
- **LEAN.**
  - De goede dingen doen.

# **Wat heeft een fout in productie tot gevolg?**

- NEGATIEVE IMPACT.**

- Op inkomsten.
  - Op perceptie naar het bedrijf toe.
    - Lastig om hier een waarde op te kleven.
    - Voorbeeld: Toyota roept 500.000 wagens terug naar de garage.
  - Ontevreden klant.
  - Slechte gebruikers ervaring.

- EEN KLANT VERLIEZEN IS ALTIJD ERG.**

- ER IS NOOIT EEN POSITIEVE IMPACT.**

# Risico gebaseerd testen

- **ZORGEN DAT: AL VINDT DE KLANT EEN FOUT ...**
  - Ok, shit happens.
- **... DAT DIT NOOIT HET BEDRIJFSPROCES LAAT FALEN.**
  - Big shit never happens.
- **MET ANDERE WOORDEN.**
  - We gaan bedrijfsrisico bepalen bij de functionaliteiten van het “onderwerp te testen”.
- **MET ANDERE WOORDEN.**
  - Onze test scenario's krijgen een prioritering.

# Wat verwacht men van een tester?

- **DAT ALLES GETEST WORDT.**
  - Dit is niet altijd realistisch.
- **EN ALTIJD WEER OPNIEUW.**
  - Het product wordt nooit kleiner, meestal wordt het groter.
- **DUS HET VOLUME VAN TESTEN.**
  - Het aantal test scenario's groeit constant.

# **En we leven niet in een ideale wereld**

- **ER ZIJN NIET GENOEG TESTERS.**
- **PROJECT IS COMPLEX.**
- **TE VEEL TESTSCENARIO'S.**
- **ONTWIKKELAARS ZIJN OOK TE LAAT.**
- **NIET GENOEG TIJD OM HET GRONDIG TE DOEN.**
- **DUS GAAN WE ONS MOETEN BEPERKEN IN HET UITVOEREN VAN TESTEN.**
  - Dus welke doen we wel?
  - En welke doen we niet?

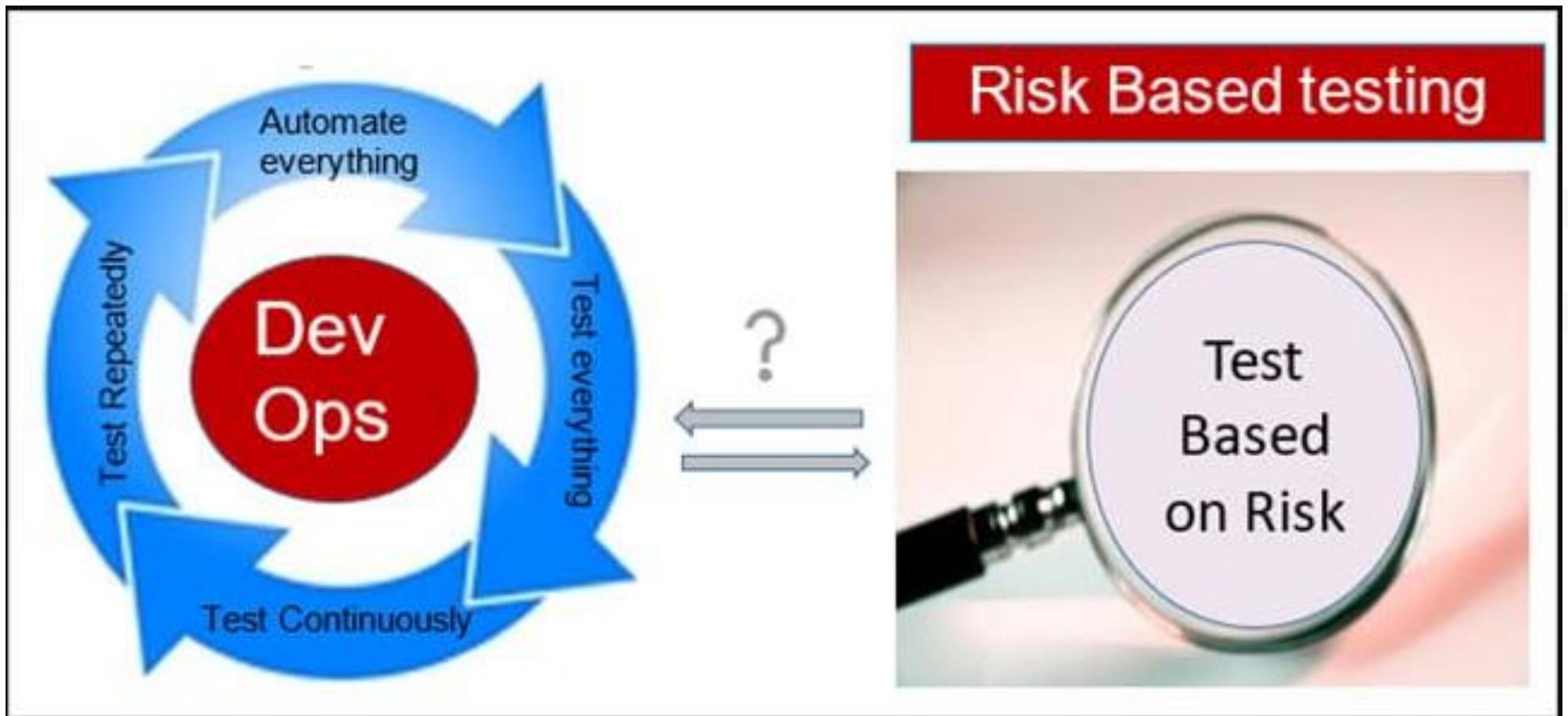
# En hoe beslissen we dit?

- **DUS GAAN WE ONS MOETEN BEPERKEN IN HET UITVOEREN VAN TESTEN.**
  - Dus welke doen we wel en welke doen we niet?
- **HOE BESLISSSEN WE WAT BELANGRIJKE TESTSCENARIO'S ZIJN?**
  - Welk perspectief?
  - De visie over de belangrijkheid van een functionaliteit kan in de ogen van de tester héél verschillend zijn dan die van de klant.
- **DUS WIE BESLIST EN HOE WORDT ER BESLIST?**
  - Risk Based Testing is een hulpmiddel om dit te bepalen.

# Risico gebaseerd testen

- **VERSUS AGILE.**
  - We kijken naar de Acceptatie Criteria van de User Stories.
  - En we voeren bij iedere release regressie testen.
- **DE VRAAG VERSCHUIFT.**
  - Welke testen worden er regressie testen?
- **VERSUS DEV-OPS.**
  - We hebben alles “geautomatiseerd” (of toch semi- automatisch).
- **DE VRAAG VERSCHUIFT.**
  - Welke testen worden er geautomatiseerd?

# Risico gebaseerd testen



# Wat deed een tester weer?

- **HET ONTDEKKEN VAN EEN FOUT?**
  - Software Quality Controle (SQC).
- **HET VOORKOMEN VAN EEN FOUT?**
  - Software Quality Assurance (SQA).
- **FOCUS LIGT DUS OOK OP**
  - Waar is de grootste kans dat er fouten gebeuren?
    - Your code as a Crime Scene investigation.
  - Probeer zo vroeg mogelijk de fout te vinden.
    - Kost minder om hem te verbeteren.
  - “Value” opleveren voor de klant.
    - End to End klant ervaring.

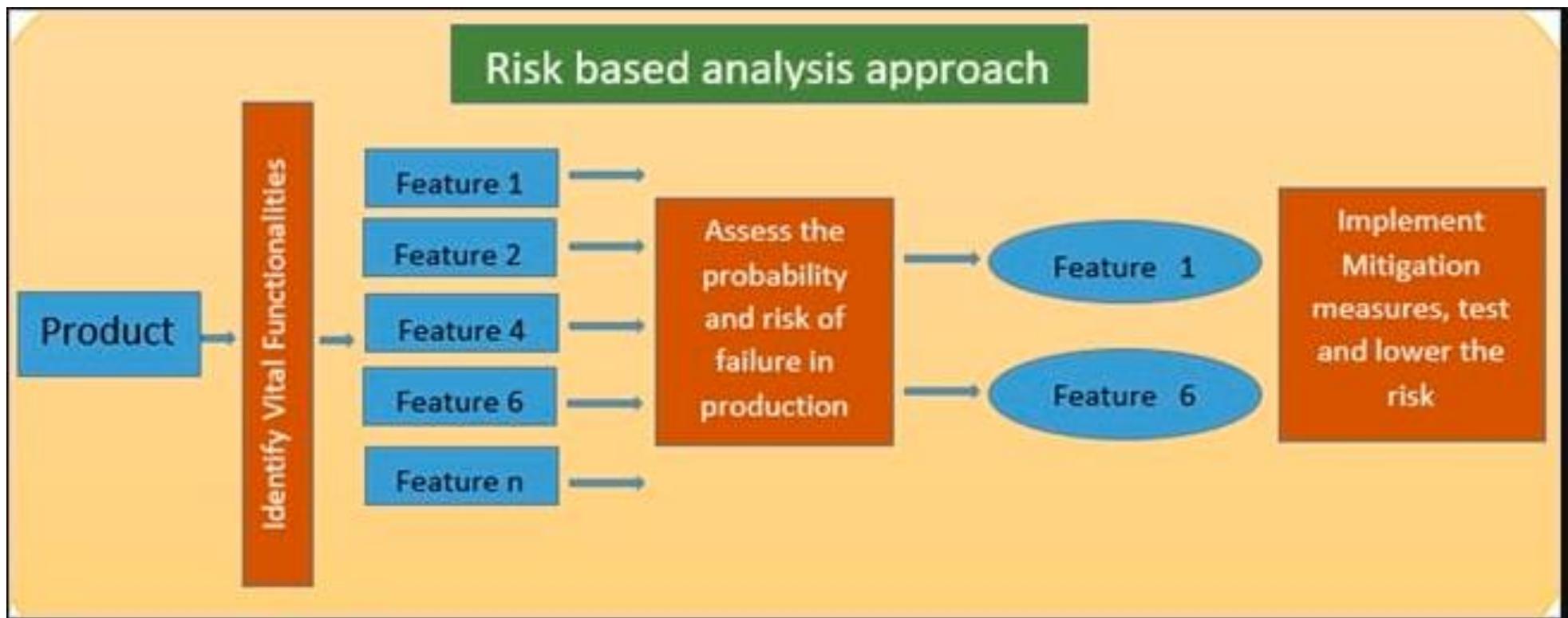
# En bugs / issues zelf hebben ook een prio

- **ZE MOETEN NIET DIRECT VERBETERD WORDEN.**
  - Het weten dat de fout er is, is soms genoeg.
  - Het verschil tussen Severity (Ernstigheid) en Priority (Prioriteit).

# Wanneer is Risico gebaseerd testen zinvol?

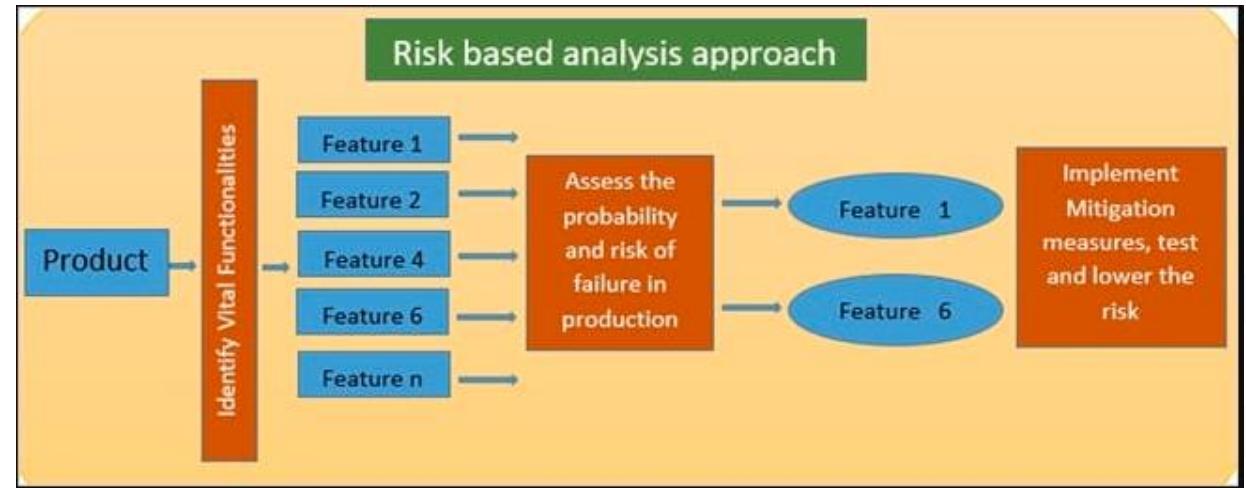
- **ALS WE KRAP IN TIJD EN RESOURCES ZITTEN.**
- **MAAR OOK:**
  - Als de applicatie hypercomplex wordt.
  - Als er nieuwe technologie gebruikt wordt.
  - Als het een Research en Development project is.
  - Als je kunt bepalen wat het “meest” gebruikt wordt.

# Wat doe je dus in de praktijk?



# Op welke items doe je het?

- EPICS.
- FEATURES.
- FUNCTIONALITEITEN.
- REQUIREMENTS.
- USER STORIES.
- USE CASES.
- TEST SCENARIO'S.
- TEST CASES.



# **Wat zijn de meest voorkomende oorzaken?**

- **SLECHT BEGRIJPEN VAN FUNCTIONALITEITEN.**
  - Historisch gegroeide functionaliteiten.
    - De “kennis” is weg.
  - Half afgewerkte functionaliteiten.
- **SLECHTE “ARCHITECTUUR” EN “ONTWERP”.**
- **TE WEINIG TIJD OM EEN GOED “ONTWERP” TE HEBBEN.**
- **INCOMPETENT TEAM.**
- **TE WEINIG MENSEN IN JE TEAM.**

# Methode

- HET CRITERIA HIER IS “RISICO”.
- HET GAAT DUS OVER DE SEVERITY (ERNSTIGHEID) VAN HET ONDERWERP, NIET OVER PRIORITY (PRIORITEIT)
- STANDPUNT VAN HET TECHNISCHE TEAM
- STANDPUNT VAN DE BUSINESS KANT

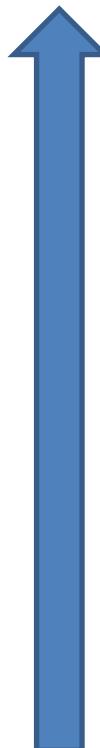
# Methode: Gaat het voorkomen?

- EEN RASTER VAN 3 OP 3


# Methode: Gaat het voorkomen?

- **STANDPUNT TECHNISCH**

- Zeer zeker
- Waarschijnlijk
- Bijna nooit




# Methode: Is er impact?

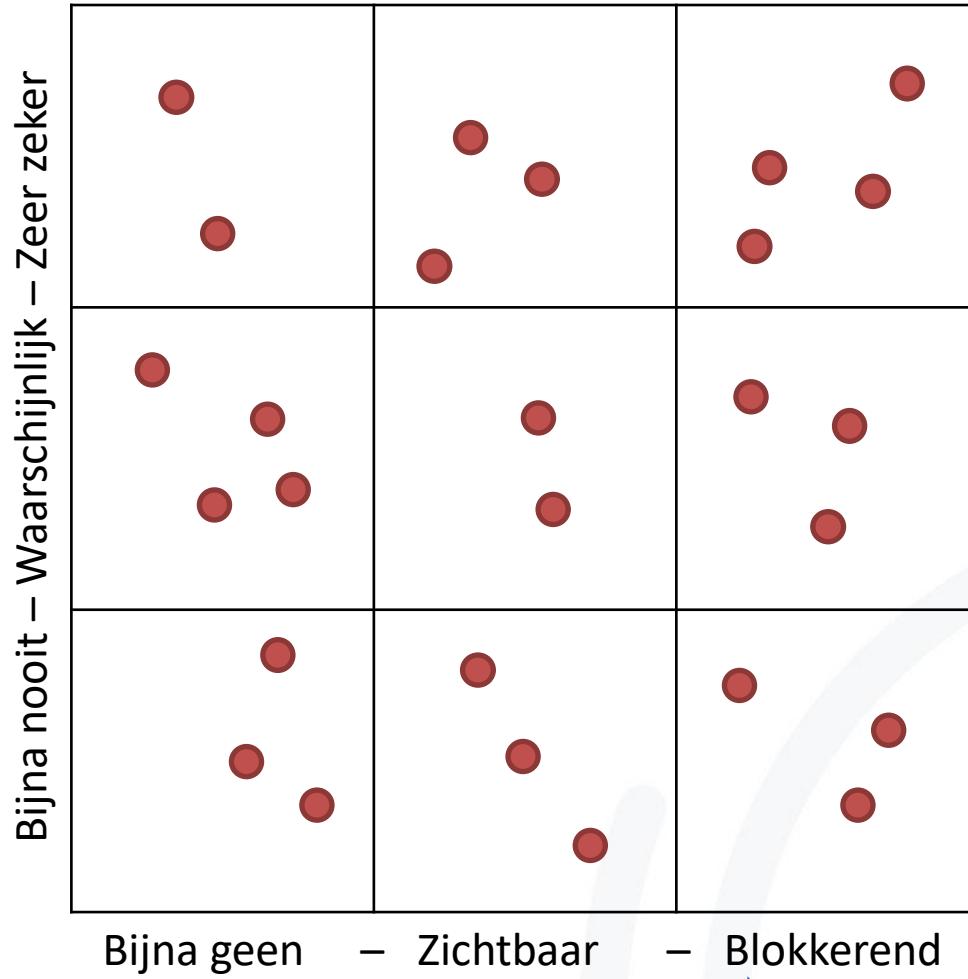
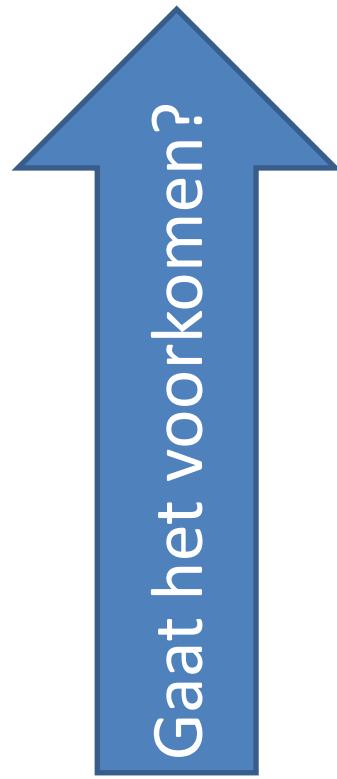
- **STANDPUNT BUSINESS**

- Bijna geen
- Zichtbaar
- Blokkerend



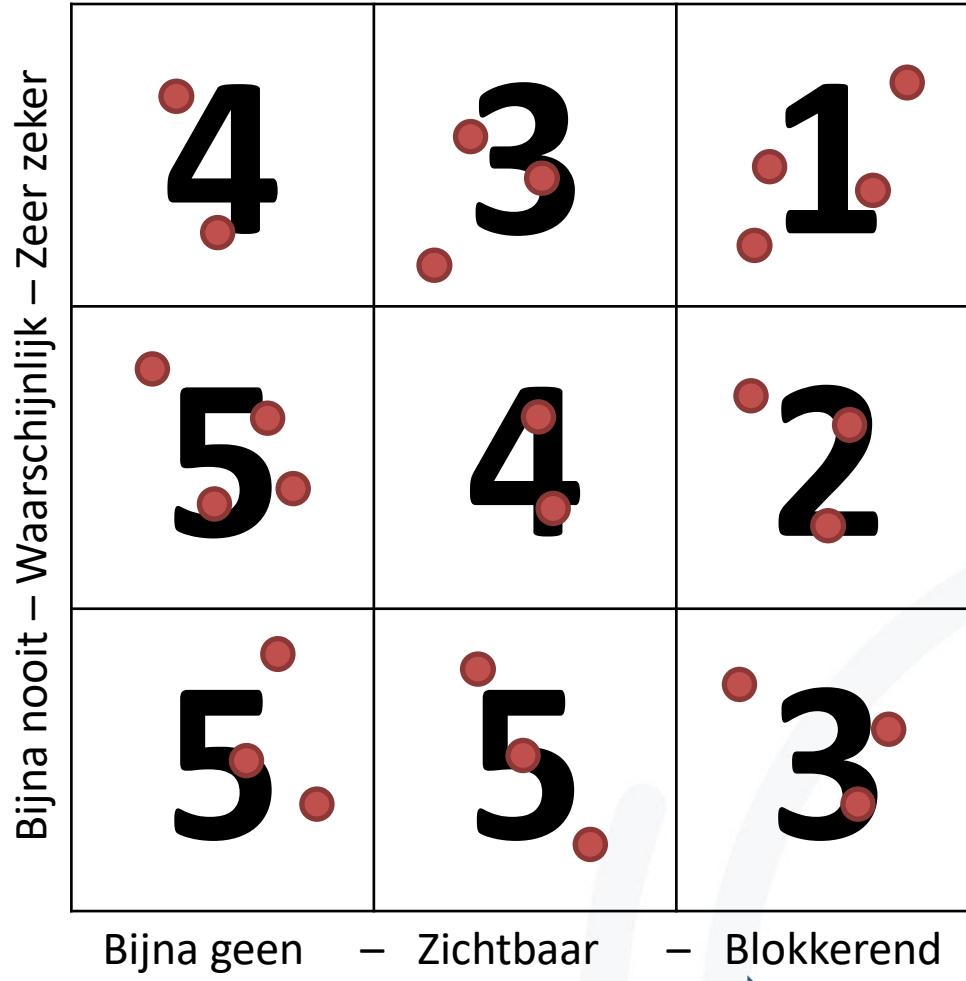
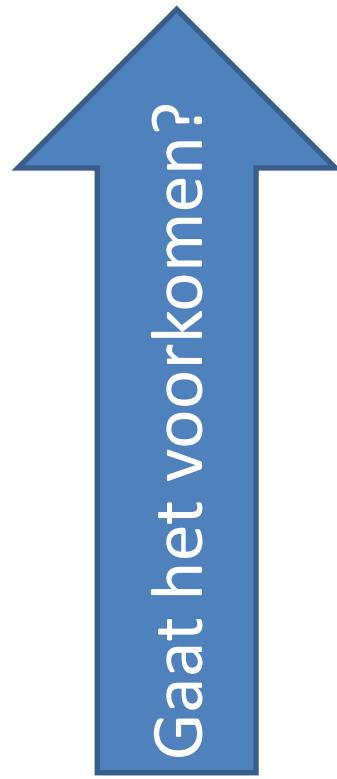
Bijna geen	Zichtbaar	Blokkerend

# Methode: Plaats de items in het raster?



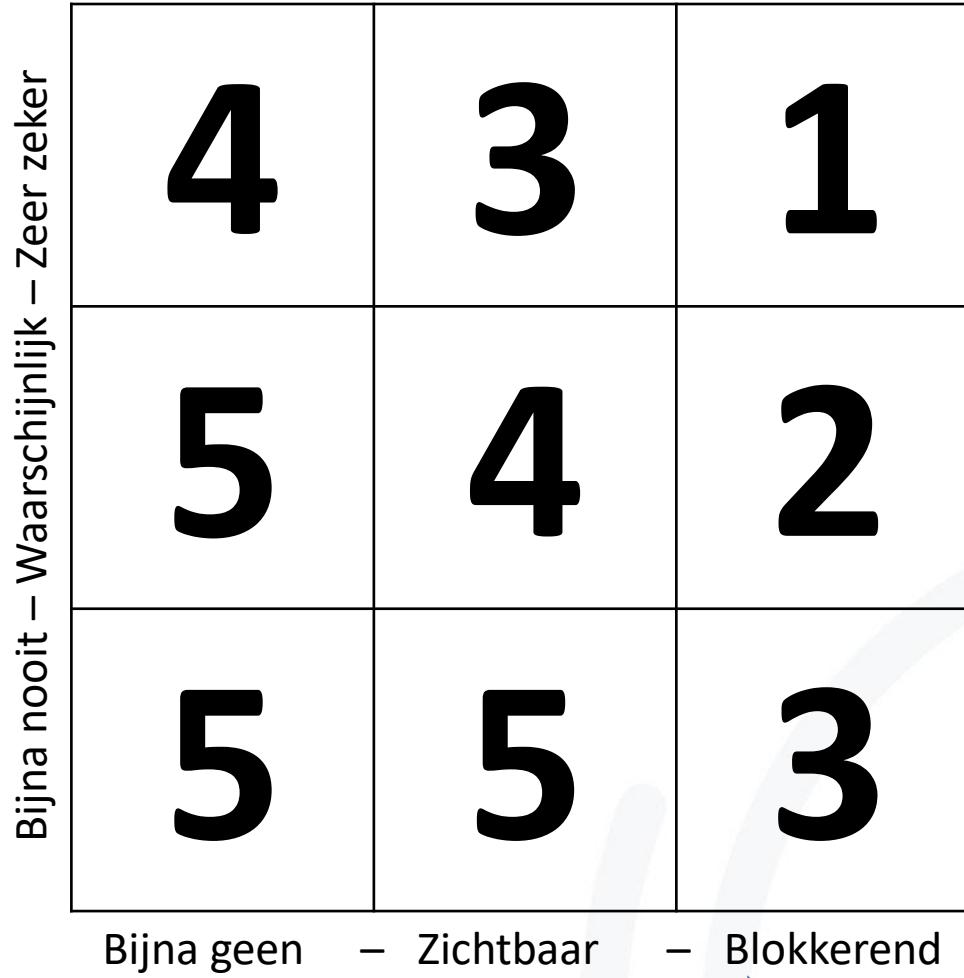
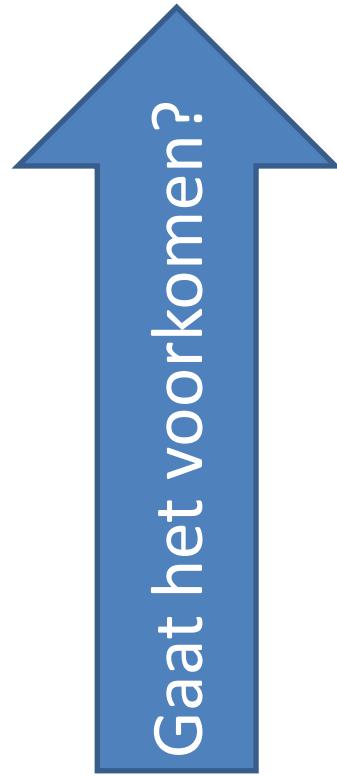
Wat is de impact?

# Methode: Geef de vakjes een waarde.



Wat is de impact?

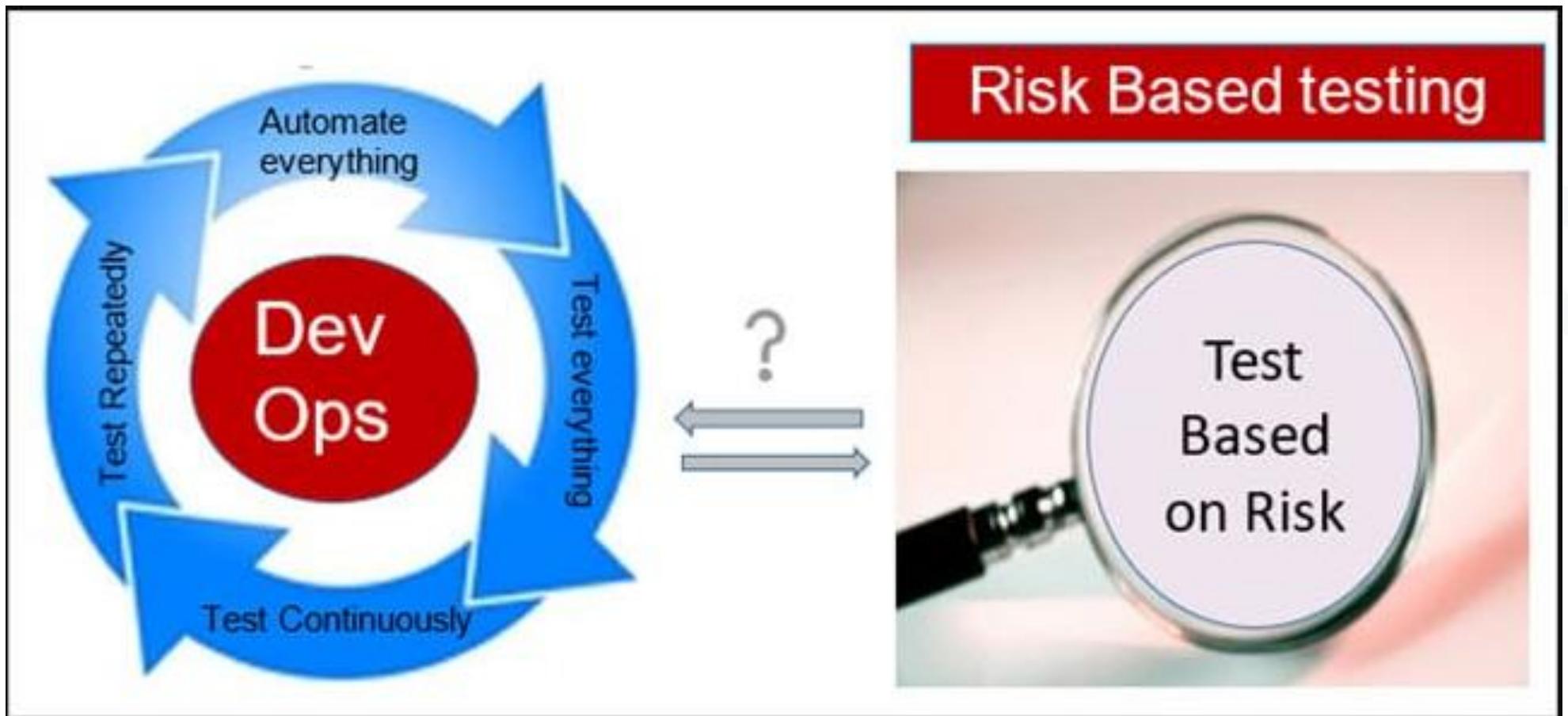
# Methode: Geef de vakjes een waarde.



- 5 • Laat vallen
- 4 • Identificeer
- 3 • Grondig
- 2 • Grondiger
- 1 • Zeer Grondig

Wat is de impact?

# Risico gebaseerd testen



# Risico gebaseerd testen in Agile & Dev-Ops

- NIET IEDEREEN AUTOMATISEERT ALLES.
- PROBEER “HOOG RISICO” AL IN ANALYSE FASE TE BEPALEN.
- ONTWERP / PLAN TEST SCENARIO’S GEBASEERD OP HET RISICO.

# Risico gebaseerd testen in Agile & Dev-Ops

- **UITVOERING VAN TESTEN BEPALEN OP BASIS VAN HET RISICO.**
- **IDENTIFICEER DEFECTEN IN EEN VROEG STADIUM**
  - Fail Fast (snel de fout vinden)
  - Fail First (eerst de fout vinden)
- **LEG DE FOCUS MEER OP QA (QUALITY ASSURANCE) DAN OP QC (QUALITY CONTROL).**

# Risico gebaseerd testen: Wanneer?

- **TIJDENS HET PLANNEN.**
  - Test Planning.
  - QA.
- **TIJDENS HET UITVOEREN.**
  - Test Uitvoering.
  - QC.

# Risico gebaseerd testen: Wanneer?

- **TIJDENS HET PLANNEN.**

- Identificatie.
  - Wat zijn onze risico's?
- Impact?
  - Hoe erg zouden we het vinden?
- We proberen "het" zoveel mogelijk te vermijden.
  - Door "voorzienig" te zijn.
  - Gebruik de "Retrospective" om herhaling te vermijden.



# Samenvatting

Zo vroeg mogelijk problemen voorzien

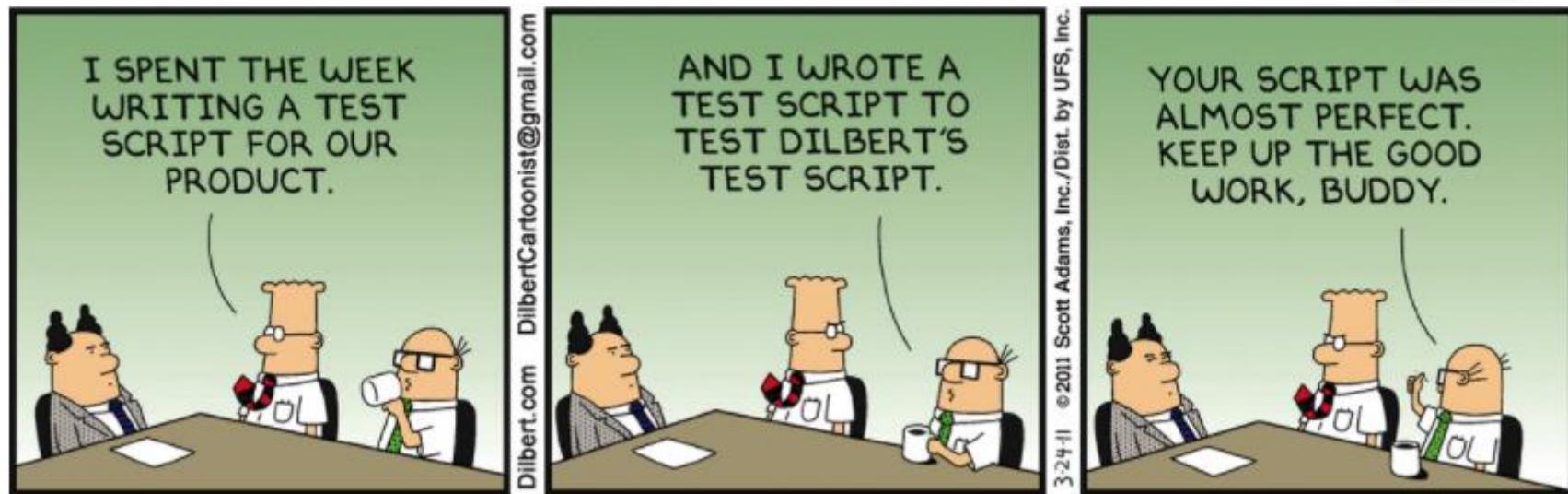
Schatten van het mogelijke “verlies”

Beslissen wat je ermee gaat doen

Vermijdt hetzelfde in de toekomst

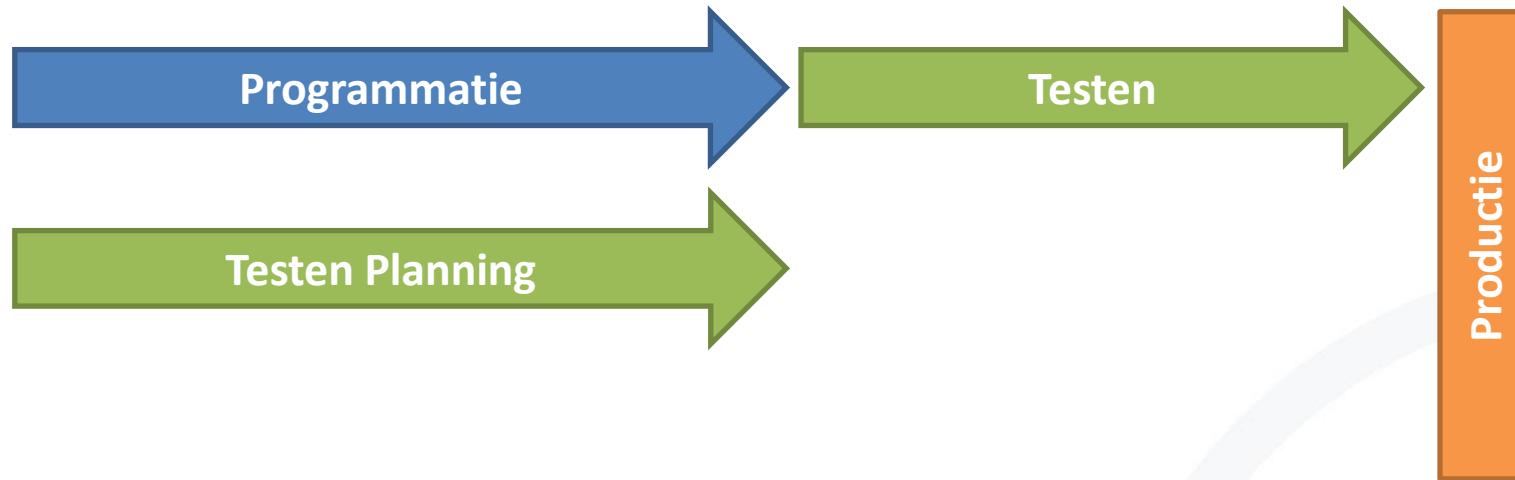
# Risico gebaseerd testen: Wanneer?

- **TIJDENS HET UITVOEREN.**
  - Test Uitvoering.
    - Hangt af van het plannen.
    - Wat niet gepland is, zal niet uitgevoerd worden.
    - Wat niet geprogrammeerd is, zal niet uitgevoerd worden.



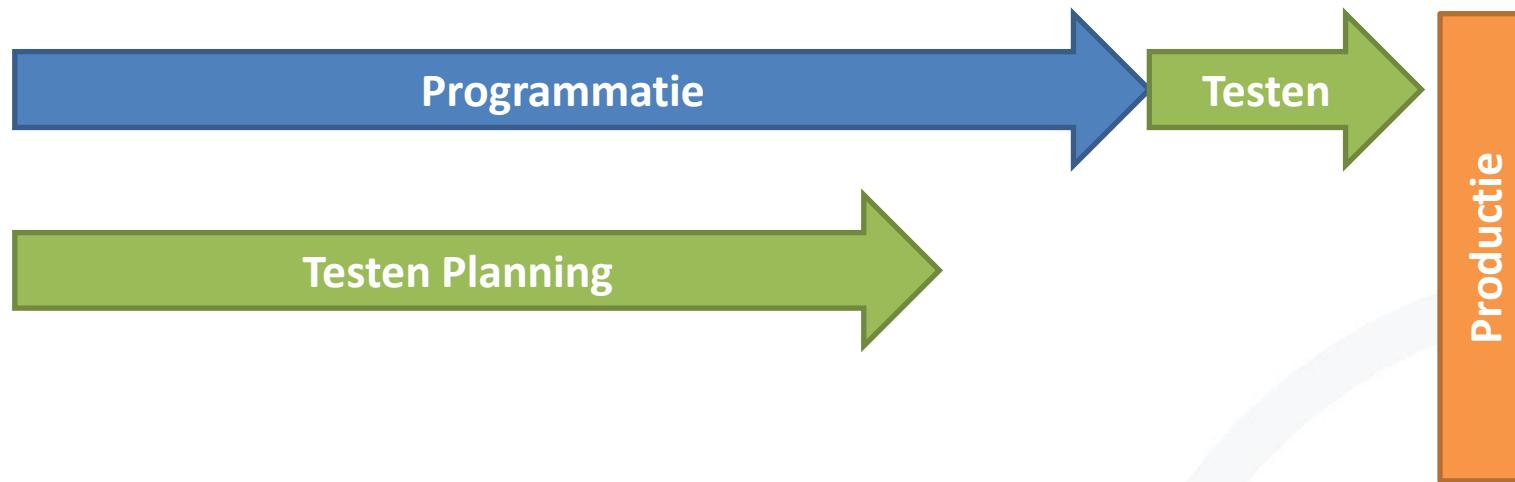
# Risico gebaseerd testen: Wanneer?

- **WAT IS ER GEPLAND?**



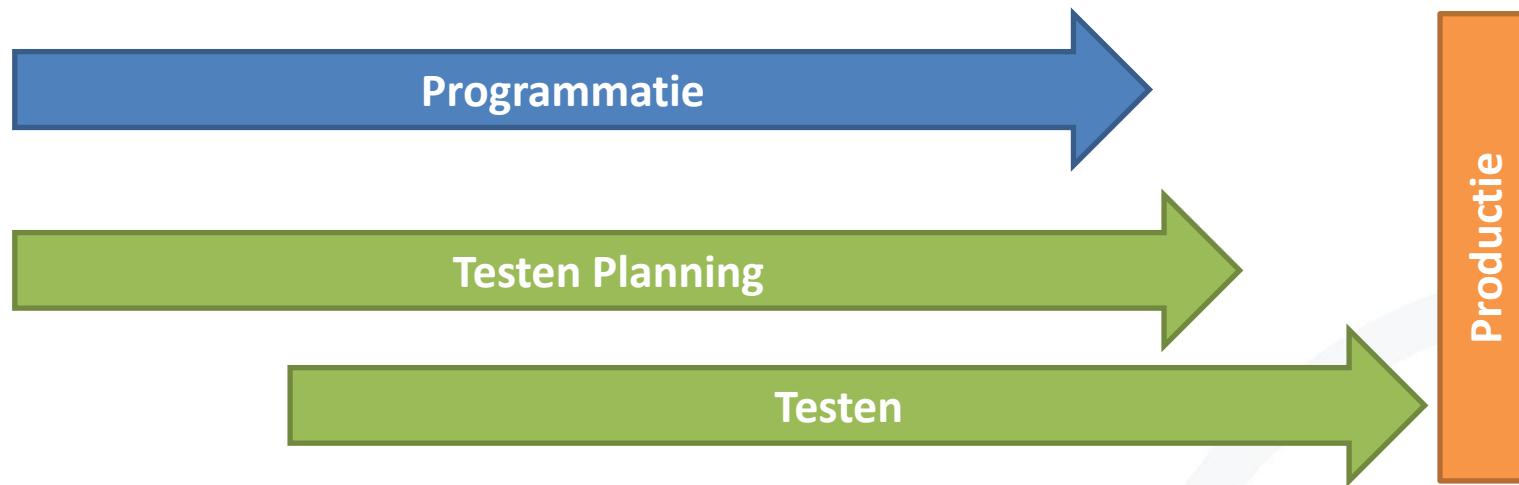
# Risico gebaseerd testen: Wanneer?

- **WAT IS REALITEIT?**



# Risico gebaseerd testen: Wanneer?

- HOE LOS JE HET OP?



# Extra onderverdeling

- **ER IS EEN RISICO IN HET PROJECT.**

- Een niet voorspelde gebeurtenis of activiteit dat de voortgang in het project in de weg staat.
- Risico in de Organisatie.
- Risico op Technisch vlak.
- Risico op Business vlak.

- **ER IS EEN RISICO IN HET PRODUCT.**

- De mogelijkheid dat een systeem of een software faalt of niet voldoet aan de vereisten van de klant, gebruiker, ....
- Risico in de functionaliteit.
- Het product doet niet wat er verwacht wordt.

# Voorzie problemen

Probeer het risico te vermijden

Probeer het risico tot een minimum te beperken

Verdeel het risico onder elkaar  
(Gedeelde smart is halve smart)

Accepteer het, en betaal de kosten